AN INVESTIGATION OF THE GAME OF POKER BY COMPUTER BASED ANALYSIS

by

A. Risticz B.Sc. Hons. (Adelaide)

A thesis submitted for the degree of
Doctor of Philosophy
in the University of Adelaide
December, 1973.

SUMMARY

The research reported in this thesis has been undertaken with the objective of finding optimal strategies for poker using both simulation and game-theoretic methods. In the introduction the nature of the problem is presented and the main work already done in this area is reviewed.

Next the simulation of poker is considered and the computational difficulties of this approach are investigated in some detail. It is concluded in this thesis that an analytic/numerical approach offers better prospects of success than simulation. Accordingly the numerical approach to poker is theoretically formulated and new methods of solution are developed which are significantly faster than existing methods found in the literature.

These methods are then used to solve 2, 3 and 4-person poker-like games. In the course of this work a new integral relevant to the solution of poker-like games is evaluated. Simulation methods are used to check the work wherever possible.

The results obtained from these solutions are unique for two reasons. First, even though simplifying assumptions were made in many phases of this work, the games solved are sufficiently realistic to be compared with a commonly played variety of poker and the solutions are shown to agree closely, in most details, with the strategies used by experienced players, even though some of the results are in less than complete agreement. Secondly, this appears to be the

first time that a solution has been obtained for a 4-person poker-like game.

The study then reports the results of the application of this work to two practical problems not directly connected with poker. The first of these relates to networks and formulates a new criterion of optimality of flow which is currently the subject of further research by another worker. The second is a problem in business management.

The thesis ends with a discussion in which general conclusions are drawn from the whole of the work and in which new areas of research are identified. In particular more research on numerical methods, on the methodology of simulation, and in applied games theory is recommended.

SIGNED STATEMENT

This thesis contains no material which has been accepted for the award of any other degree or diploma in any University. To the best of my knowledge and belief, this thesis contains no material previously published or written by any other person, except when due reference is made in the text of this thesis.

(A. Risticz)

ACKNOWLEDGEMENTS

I would like to express my appreciation of the guidance, encouragement and help given by Dr. J.L.C. Macaskill during the course of my research, and to Professor J. Ovenstone for his advice during the preliminary stages of my work. I also thank Professor F. Hirst in his capacity as Permanent Head of the Department of Computing Science for his encouragement and support, and Dr. I.N. Capon, Director of the University of Adelaide Computing Centre, and his staff, for their help and tolerance. The writer is indebted to Professor R.B. Potts, Dr. C.E.M. Pearce and Mr. R.J. Harris for valuable discussions and to Miss J. Potter for meticulous typing.

TABLE OF CONTENTS

I give consent to this copy of my thesis, when deposited in the rsity Library, being available for loan and photocopying.

....26/9/74.................. Signed

CHAPTER 1

INTRODUCTION

1.1 Statement of problem

Before making a general statement of the problem to be dealt with in this thesis the following 2 points should be noted.

(a) The mathematical treatment of poker can have practical applications in areas not directly related to poker. The following quote from Karlin, (19), supports this assertion (also see Pruitt (29)):

"The many respects in which business, politics and war resemble poker should be evident. Hence, any progress in our mathematical understanding of poker games can have its counterpart interpretation in many relevant circumstances of life."

(b) Poker is possibly the most complex of card games. Indeed Karlin, (19), states that:

"It is the considered opinion of many expert card players that poker requires the most skill and depth of any card game."

The intricacies of poker may be appreciated by reading one of the many books written about the subject, for example "Poker: Game of Skill", (30). As a consequence of this complexity the mathematical treatment of poker must be restricted to simplified poker-like games (see the literature survey given later in this chapter).

In this study the game of poker has been approached in such a way that, although certain essential characteristics of the game are retained, simplifying assumptions make the problem amenable to mathematical solution. Numeric methods, suitable for use on the computer are used to obtain solutions which are then directly applicable to a commonly played variety of poker.

Various applications of this work to other, not directly related areas, are then considered.

1.2 Main aim of the work of this thesis

The main difficulty in the mathematical treatment of poker-like games arises from computational difficulties (see later in this chapter and chapter 5). This work hypothesises that, by using numeric methods (suitable for implementation on a computer), further progress may be made in solving more realistic models than have been hereto possible (see chapter 5).

To find these solutions a new mathematical technique for handling the particular type of games treated here has been developed (see chapter 3). Although the applicability of poker to business has been suggested by Pruitt, (29), and Karlin, (19), no concrete examples could be found in the literature. Thus it was an object of this study to provide a specific example of the application of the above work to a business situation (see chapter 6). Other applications to poker simulation and the theory of networks are also given in chapters 2 and 6 respectively.

REVIEW OF RESEARCH INTO POKER

1.3 Methods of classifying poker-like games

Until 1928 the mathematical analysis of poker was limited to a probabilistic and combinatorial treatment (see Borel and Ville (3)). In 1928 von Neumann, (26), published the first paper on game theory and showed how it was possible to obtain exact solutions to simple poker-like games using his newly developed theory. Von Neumann formulated and solved a simple poker-like game, (27), and found that the solution contained the element of bluffing (see 1.4.1), which had previously been considered to include psychological factors, and was therefore considered not to be a suitable case for a simple mathematical treatment. (A further discussion of bluffing is given in chapter 3).

Since then many papers have been published dealing with increasingly sophisticated versions of poker-like games, (12,13,20,22,24,27,29). The recurring problem has been one of computational difficulty. Thus, although it can usually be proved that optimal strategies to any poker game exist, the mathematical techniques necessary to discover them are generally lacking.

When presenting the survey of the literature on poker-like games it is helpful if the games are classified by the following parameters.

(a) Number of players

The number of players largely determines the overall complexity of the game, as special difficulties are encountered when the number of players exceeds two.

(b) The hand structure in the game

The hand structure in the game is usually either discrete, finite and small or else continuous, and plays a large part in determining the method of solution. Real poker is an exception in that although the hands are discrete, they are so large in number, that the hand structure can be considered to be continuous.

(c) The rules governing betting

There is considerable variation in the rules governing betting. The more involved the betting, the greater are the computational difficulties. Some games have a great variety in the bidding and rebidding, while others are limited to one type of bet.

(d) The particular method used to solve the game

There is no single method which is applicable over the whole range of games. The techniques used to solve each individual poker-like game are usually different.

(e) Relationship to real poker

Games may be further classified by the insight that they provide into real poker.

1.4 Games of historical interest

A number of games that play an important part in past research but that are not directly related to this thesis are now presented.

1.4.1. The games of von Neumann and Morgenstern

Von Neumann, (27), was the first writer to formulate and solve a poker-like game. The rules of this game allow two players to obtain random hands $s_1 \in [0,1]$ and $s_2 \in [0,1]$ where s_1 and s_2 are uniformly distributed over the closed interval $[0,1]$. Both players bid simultaneously, either a or b units (where $a>b$ and a and b are fixed) not knowing the value of the other player's bid. The term "bid simultaneously" signifies that the players make their bids together, and each player has no knowledge of what his opponent will bid. If both bid the same amount, hands are compared with the higher hand winning the pot. If one bids high, the other low, the player bidding low has the option either of forfeiting his low bid, or of increasing the amount bid to equal the bid of the other player, in which case the higher hand wins the pot.

The method of solution given by von Neumann, (27), (although it is not able to be applied directly to the work of this thesis) gives heuristic insights into how real poker should be played and shows clearly the advantages of bluffing (see chapter 3).

Bluffing takes place when a player makes a high bid (a units) with a hand which is not likely to win if the other player should look. The bluffing strategy for this game will be described precisely in chapter 3. Bluffing has 2 purposes. First, the high bet may force a winning situation by causing the other player to back down. Second, if the other player does look, he will notice the bluff. Later, therefore, he is more likely to look at a high bet, since he may feel that it is another bluff. Thus, bluffing will tend to ensure that a higher profit is made from good hands as high bets will be more likely to be looked at.

A modified form of this game allows the first player to bid either a (high) or b (low). If he bids high the second player has the option either of dropping out and forfeiting b units, or of matching the first player's high bid, with the higher hand winning a units. This model is discussed in chapter 3.

1.4.2. The games of Gillis et al, and the game of Bellman and Blackwell

In their paper Gillis, Mayberry and von Neumann, (13), have solved a two person variant of a poker game, with simultaneous bidding and a continuous hand structure. It is similar to von Neumann's original game, (27), except that instead of allowing two fixed sizes of a bid, a and b, the bid can take any value between a and b. Two versions are considered, one where all bids in the closed interval [a,b] are allowed, and the other with only a finite possible number of bids in the same interval. But the rules differ from von Neumann's original game (1.4.1) in that no upgrading of bet by the lower bidder, is allowed and the higher bidder wins the amount of the low bid. If bids are equal, the higher hand wins the amount bet.

Bellman and Blackwell, (2), have also solved a variant of poker, which has two players, a continuous hand structure, and is very similar to the game of von Neumann, (27).

The main characteristics of the games that have been described are:-

(1) Only 2 participants are allowed (i.e. they are 2-person games).

(2) The hand structure is continuous.

(3) Betting is limited with no reraising.

(4) Although the methods of solution are different, none is applicable to games treated in this study.

(5) The solutions show the importance of bluffing.

(6) The solutions cannot be directly applied to any real game of poker.

1.4.3. The games of Karlin and Restreppo, and Pruitt

Karlin and Restreppo, (20), have developed a fixed point method which has application to various models as follows.

The first model relates to a 2-person game with a continuous hand structure and k rounds of betting. No other game has been noted that allows a large number of raises and reraises as does this model. The second model closely resembles the game solved by Gillis et.al. (see 1.4.2).

Pruitt, (29), has used the method developed by Karlin and Restreppo, and applied it to a continuous version of stud poker. Stud poker is a variety of poker that is fundamentally different from draw poker, to which this thesis relates. The major point of difference between the two games arises because certain cards belonging to each player are revealed in stud poker, whereas hands are completely concealed in draw poker.

1.5. Games of direct significance to this thesis

The games that are now presented are of direct significance to the development of the work of this thesis.

1.5.1. Kuhn's 2-person game

Kuhn, (22), has proposed a simplified 2-person poker-like game which introduces the concept of behaviour parameters. Behaviour parameters describe the probability with which actions are undertaken in any prescribed situation, and can be used to simplify the computation involved in solving a game. (This concept is described and used in chapter 4).

The game has a discrete hand structure (only 3 types of hands are allowed) and the solution exhibits both bluffing and underbidding. Bluffing is a feature which was present in the other models described previously, but underbidding has not occurred before. Underbidding occurs when a player bets the minimum possible while holding a strong hand. This manoeuvre is used by experienced poker players when holding a strong hand, in order to entice the opposing player into making a large bet, which is then raised. It has the added advantage that, the opposing player having been trapped in this way is later reluctant to raise a small bet, even when he has what is quite probably the winning hand.

1.5.2. Nash and Shapley's 3-person game

The 3-person game of Nash and Shapley, (24), is important to this study because it presents the concept of the equilibrium point (e.p.) Nash, (25), defined the e.p. in order to solve non-cooperative games with more than 2 players, a situation which von Neumann's original theory does not encompass. Nash and Shapley's game is an extension of Kuhn's 2-person game to the 3-person case although the number of possible hands is reduced from 3 to 2 . The solution obtained shows exactly the same features as Kuhn's game, i.e. bluffing and underbidding.

1.6. Friedman's game

Friedman, (12), was the first writer to consider a realistic bluffing situation, which even though very simplified, had a direct practical application to certain situations which arise in real poker. The general approach to obtain these solutions is based on concepts similar to those used in this thesis. The main features of these games are as follows.

Friedman first considers the 2-person game where the pot contains k units, and player A holds a 4-flush (1 card short of a flush) while player B holds 3 of a kind. Player A discards 1 card and has a chance $p \approx 0.2$ of making the flush. Player B discards 2 cards but his chance of improving is small by comparison (probability 0.085). It is clear to player B that since player A has bought 1 card, then if he completes his flush, he will beat player B's hand unless B improves (highly unlikely). Player A is first to bet, and he can either decline to bet or else bet 1 unit (the maximum bet allowed). Obviously, in this situation, he is in a strong position to bluff player B (by betting 1 unit).

Friedman shows that player A should bluff, when he misses his flush, with a probability of $\frac{p}{(1+k)(1-p)}$

[obviously A will always bet the maximum if he completes his flush]. Also player B should look at A's potential bluff with a probability of $\frac{k}{1+k}$. It is shown that with k=1, one third of A's bets should be bluffs, and that B should look at such possible bluffs one half of the time.

Friedman next considers the same situation except that this time player B is allowed to reraise 1 unit, after A's bet. Again the general conclusion reached is that one third of all raises should be bluffs, while one half of all potential bluffs should be called. Friedman conjectured that this generalised strategy may also apply to more complicated situations which defy analysis.

These solutions may have a direct application to real poker, and are simple to apply in practice.

1.7. Numerical techniques

All of the above mentioned games were solved by analytic methods, and none were solved numerically. By numerical methods it is meant that some sort of iterative technique is used which can approximate the exact solution. The reason for considering such methods is that they may be implemented on the computer and in this way allow solutions to be obtained which can not be obtained analytically (this is very similar to the situation encountered in the work on differential equations, where iterative numerical methods play an important role).

One of the major tasks of this thesis was to find e.p.'s for certain n-person games. However, in the course of this work it was found that the equations obtained were too complicated to be solved analytically.

Only 1 numeric method suitable for use with this problem was discovered in the literature. This was the method of Rosen, (31), which iterated to find the solution point by calculating certain derivatives. This method is discussed (chapter 4) and shown to be unsuitable for the games considered here.

A survey of the literature showed that very little work has been carried out in the area of numerical methods. However, the usefulness of this approach will be demonstrated later in this study (chapter 5) when results are obtained which could not be obtained analytically.

1.8. Computer simulation of poker

Simulation might be expected to provide an effective approach to the solution of poker-like games. However, the only worker whose publications develop this approach is Findler. Findler, (10), gave a flow-chart for a proposed poker playing program, and recently published papers (11,37,38) which indicate that further work is currently being carried out in this area. Some attention has been given in this thesis to the simulation of poker as noted in the following section.

NEW WORK PRESENTED IN THIS THESIS

1.9 Poker simulation

As a part of this study a poker simulator was programmed. The poker simulation is described in chapter 2 and is related to the main body of the thesis in the following ways.

First, even though initially it was hoped that simulation could be used to assist in the analysis of poker, it was established that this was not a suitable method for the purposes of this particular work. Secondly, certain analytic results which are obtained later may be employed in this program to enable it to play a better game of poker. Thirdly, the use of interactive programs in the analysis of poker are discussed. Furthermore, as a byproduct of this work, certain poker probabilities, which might be of interest to other workers in this field, and to the practical poker player, were evaluated.

1.10. A new solution method for n-person games

As has been mentioned in section 1.7, a suitable numerical method for solving the class of games considered here, could not be found in the literature. As a part of this study a new algorithm was formulated to meet this need, and it has been used extensively throughout this thesis.

1.11. Evaluation of a special integral

During the evaluation of the payoff functions for the poker-like games considered here, a particular multi-dimensional integral was found to occur repeatedly. This integral, is of central importance to this thesis and as it has not been evaluated elsewhere it is evaluated here (see Appendix A).

1.12. Formulation and solution of a new poker-like game

In chapter 5 a 4-person poker-like game is formulated and solved. It is considered that this game is a significant contribution to the results already achieved in this area of study, for the following reasons

(a) It is the first poker-like game solved which allows more than 3 players (a 4-person game is considered).

(b) It has the unique property that it can be related to a large subset of a commonly played variety of poker. It is noteworthy that the solutions found agree closely with strategies commonly used by experienced players.

1.13. Applications

An application of game theoretic methods to a network is presented in which a new criterion of optimality is defined. This allows the network to be optimized using methods developed in this study.

Even though the similarity of poker to business situations has been noted by Pruitt, (29), and Karlin, (19), no explicit examples could be found in the literature. Thus, as a sequel to this study, a business operation, which is directly related to poker, is defined and solved.

In this way it is shown that the main work of this thesis may be applied to other, not directly related problems.

LAYOUT OF THESIS

1.14. Layout of thesis

This thesis has been organized in the following way. Chapter 1 defines the problem considered, presents a literature survey, and describes new work carried out. Chapter 2 describes the computer simulation of a commonly played variety of poker, and its relevance to this work. In chapter 3 a mathematical statement of the problem of solving n-person games is given and then a new algorithm, specifically designed to solve the games treated in this thesis, is presented.

In chapters 4 and 5, 2,3 and 4-person versions of the game simulated (but unsolved) in chapter 2 are considered, and solutions found.

Chapter 6 presents examples of practical applications of the work carried out here. Chapter 7 summarizes the work done, draws conclusions, and indicates possible new areas of research and application.

The subjects associated with the evaluation of the integrals are treated in chapters 3,4 and 5 while the evaluation of the integral is presented in appendix A.

CHAPTER 2

SIMULATION OF POKER

INTRODUCTION

2.1. Introduction

Some work has been done in this study on the simulation of poker but, for reasons discussed in detail later in this chapter, the research was not completed. However, during the research, results were obtained as follows:-

(1) probabilities were defined and calculated of winning with specific hands, both before and after the draw, with given numbers of players

(2) values were obtained for computation times for investigating different poker strategies by simulation

(3) the strengths and weaknesses of investigating poker by interactive simulation were assessed by means of a pilot study.

These results are considered to be of sufficient value to justify a discussion of the work done on the simulation. This discussion is introduced by presenting briefly the principles used in simulating a game of poker.

2.2. Principles of poker simulation

The rules of the game simulated are given in some detail in section 2.5, but for general explanatory purposes a brief description of the game simulated is as follows. Five cards are dealt to each of up to 7 players. Players consider the value of their hands (defined by combinations of pairs, three of a kind etc. as described later) and may, by paying money to enter, take part in the game. Having entered, a player has the option of replacing up to 4 cards from his hand in order to improve his hand. When all players have exercised their option a given player (the player to the left of the dealer) may, if he wishes, bet an amount on his hand or drop out. If he bets the next player may raise the bet, or, by betting an equal amount stake a claim to "see" his predecessor's hand, or drop out. The opportunity to exercise these options passes from player to player until all players but one have dropped out or, until all players but one, have claimed to see the hand of the remaining player. In the latter instance the player seen must show his hand and the monies staked go to him unless some other player lays down a stronger hand. In the former instance the residual player takes the pot without showing his hand. Because a player may win without showing his hand there is the opportunity for a player to win by bluffing, that is by giving an impression of strength sufficient to frighten his opponents out of the game.

The simulation of the game may be undertaken in phases thus:

(1) Generation of hands

The first step in simulating a game of poker is to agree on a representation of each card by an integer from

1 to 52, thus:

Ace of Clubs $\equiv$ 1

Two of Clubs $\equiv$ 2

⋮

Queen of Spades $\equiv$ 51

King of Spades $\equiv$ 52

These 52 digits are stored in an array and a shuffling algorithm applied, which uses a random number generator, and which guarantees that every possible combination of 52 cards is equally likely. This algorithm is described in section 2.6. This representation of a shuffled deck of cards may now be used to deal each player a hand of 5 cards, and to deal replacement cards.

(2) Entering the game

In order to decide whether to enter, the probability of winning before the draw must be calculated. With this knowledge it is possible to define probability values which will be criteria for entering or dropping out. Thus, if at a given stage of the game a hand that has the option of entering has a probability of 0.80 (say) or more associated with it, then it could be asserted that that hand would continue. The probabilities for each hand would then be compared, in a similar way with these criteria to decide the fate of that particular hand at time of play.

(3) Improvement of hand

It is easy to define a deterministic algorithm by which, given any hand, the course to follow will be specified. For example, if a hand contains 2 pairs, then the rules might prescribe the discarding of the non-paired card. A deterministic algorithm of this type is described later in this chapter. Obviously the element of bluffing could be introduced into this stage of the game by specifying, according to the value of a random number generated, that a different algorithm might, or might not be used. This second algorithm might, for example, be designed to give the impression that a poor hand was good.

(4) Betting

The stage of the game concerned with the betting would be similar to (2) above. In the same way as in (2) it would be needed to know a probability, in this instance the probability of winning after improvement. Again, as in (2) a probabilistic system for betting would be required. In this instance, however, more sophisticated betting rules would be needed, based on the probabilities of winning after hand improvement. For example it would be needed to know how far betting should be taken on a given hand - i.e. how many rounds of betting should be entered into.

(5) Determination of the winner

Provided that more than one hand is left in the game at completion of betting, the method of winner identification is simple. Since each hand is known specifically, it is merely a matter of comparing hands and selecting the strongest.

2.3 Results achieved by other workers

A search of the literature revealed that Findler, (37,38,10,11), is the only writer who has considered the simulation of poker. Findler's first paper, (10), presented a flow diagram of a proposed poker playing program. The program presented here is, in broad detail, similar to the program suggested by Findler.

Findler's later work uses the simulation of poker as a means of studying decision making. A detailed examination of this work is not included here for two reasons. First, these papers were not available at the time of writing this study, and secondly, Findler's results do not affect the validity of the deductions made here.

2.4 Factors involved in setting up a simulation

The main activities involved in setting up a poker simulation are the following.

(1) Represent each of the 52 cards in the deck by a unique integer from 1 to 52.

(2) Develop algorithms which will shuffle this deck, and calculate various probabilities associated with a hand of cards that are crucial to the game of poker.

(3) Prepare an algorithm which will calculate which cards to discard during the hand improvement process.

(4) Now methods of describing betting strategies must be found which are intelligible to a computer. The method used here is based on the program being able to evaluate probabilities relating to a poker hand, then employing algorithms which calculate when and how much to bet in a given situation for a given strength of hand. These algorithms give numerical procedures which calculate the size of the bet, using the above mentioned criteria, and are based on observed patterns of play followed by experienced poker players, (30).

(5) Using (1) to (4) above it is a simple matter to construct a poker playing program, which must then be verified.

DETAILED METHOD OF SIMULATION

2.5 Rules of the game

Poker is a game played with a deck of 52 cards, and any combination of 5 cards constitutes a hand. There is a hierarchy of hands which is well known and is given in table 1. The particular game simulated here is called 5 card draw poker and is the most commonly played version of the game. It has the following rules.

TABLE 1

CLASSIFYING POKER HANDS

Poker hands may be classified into 9 different types as given below in decreasing order of strength. It should be noted that apart from the flush and the routine, suits are irrelevant to hand strength.

1. Straight flush or routine, which contains 5 cards in sequence, and in the same suit.
2. Four of a kind, and 1 odd card.
3. Full house which has 3 of one kind, and 2 of another.
4. Flush, which has 5 cards of 1 suit.
5. Straight, which has any 5 cards in sequence.
6. Three of a kind, and 2 idle cards. (Idle cards are ones which take no part in determining hand strength.)
7. Two pairs and 1 idle card.
8. One pair and 3 idle cards.
9. No pair and 5 idle cards.

The rules for determining the stronger of 2 hands when both are of the same type are well known and may be found in any book on poker (see (30)).

There are n players $(1 \le n \le 7)$ numbered consecutively by the integers 1 to n in a clockwise manner (looking on to the table from above). Player $n-1$ is called the dealer and player n the blind.

The dealer shuffles the deck of cards and deals 5 cards to each player. The blind now places a compulsory bet of 1 unit into the pot. Next all players look at their hands, and decide in turn, beginning with player 1 and ending with the dealer, whether or not to play. If they wish to play they must place 2 units in the pot. If no player decides to play, the blind retrieves his 1 unit bet and the game ends, otherwise the blind has 3 options open to him.

(1) He may drop out of the game and forfeit his 1 unit bet.

(2) He may elect to play on by making an additional bet of 1 unit.

(3) He may bet a further 3 units (thus making a total of 4 units).

This is called doubling, and in this case each player, in ascending numerical order, must either drop out and forfeit his bet or else place a further bet of 2 units into the pot. If, after a double, all other players drop out, then the game ends and the blind wins the pot.

Next, each player in turn, may discard up to 4 cards from his deck and receive an equivalent number of cards from the dealer. This will be referred to as the hand improvement stage and completes the first phase of the game.

In the second phase of the game all remaining players take part in a round of betting that determines the ultimate winner. Betting begins with player 1 and proceeds in a clockwise manner with each player in turn having 3 options.

(1) He may drop out of the hand.

(2) He may "look", that is make such a bet that his total bet becomes equal to the greatest total bet made by any other player still in the game.

(3) He may "raise", that is bet sufficient to look, and then increase this by any amount from 1 to 5 units.

Until some player makes an initial bet of 1 to 5 units, all players who "pass" (i.e. do not bet) must drop out of the game.

Betting is continued until either only 1 player remains in the game, in which case he is declared the winner, or else until, after a raise by one of the players, there has been no further raise by the time betting returns to that player. In the second case the winner is the player showing the best hand.

In the succeeding game the blind becomes the new dealer and the other players are appropriately renumbered.

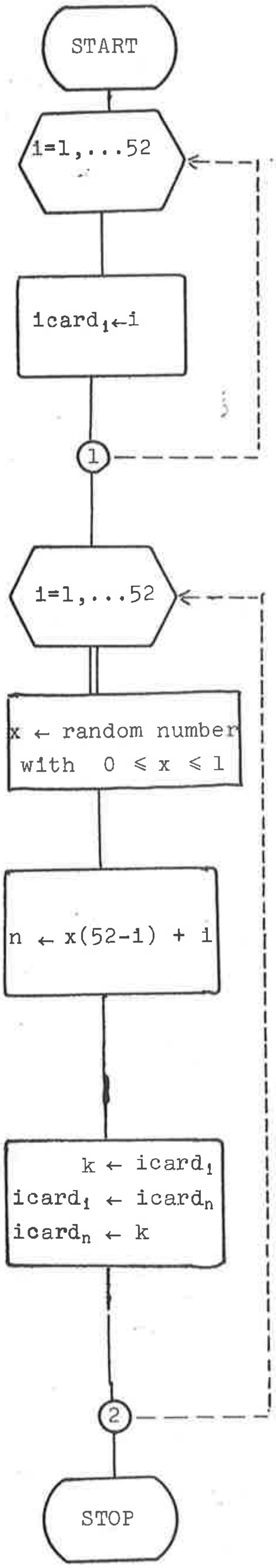
2.6 Simulation of shuffling

Each card is represented by a unique integer from 1 to 52. Initially, these integers may be stored in any order in an array, ICARD(1)....ICARD(52). An algorithm was written which used a random number generator to shuffle the digits randomly in the array ICARD. The algorithm was based on a method proposed by de Balbine, (1) and has the property that after the shuffling, all possible combinations of the 52 cards are equally likely. A flow diagram of this algorithm is given in figure 1 in the form applied in this study.

This algorithm works by exchanging the first element in the array ICARD(1),...ICARD(52) with itself or with any other element on the right, with equal probability for all exchanges (the particular element exchanged is determined by the random number x). This process is repeated with the second, third, fourth element etc. up to the second last element. de Balbine shows that, regardless of the initial ordering, this method will yield a random permutation of the original array, with all possible permutations being equally likely. Once the deck is shuffled the cards are dealt to the players in the conventional way, one card per player, until each player holds 5 cards.

FIGURE 1

DECK SHUFFLING ALGORITHM



Set up the integers

1-52 in the array $icard_1,...icard_{52}$.
This represents the unshuffled deck

Consider the ith card.

x is set to a random uniformly distributed number in the closed interval [0,1].

x is used to calculate a random integer n where, because of the method of calculation, n lies in the range from i to 52, with equal probability for all numbers.
For example, i=8, x=0.5 gives n=30.

Now the ith card is interchanged with the nth card.
This forms the basis of Balbine's algorithm, as he shows that at the completion of this procedure all possible orderings of the deck are equally likely.

At the completion of the algorithm the deck stored in $icard_1,...icard_{52}$ has been randomly shuffled.

2.7 Improvement of hand

An important matter in a hand of poker is choosing which cards to replace in the hand improvement stage. There is general agreement amongst card players concerning which cards to discard from any given hand, and an account of this is given in Reece and Watkins, (30).

Thus it is commonly accepted that, if a hand contains 2 paired cards, and 3 non-paired or idle cards, then it is best to discard the 3 idle cards. An aggregation of these accepted norms would make it possible to construct a deterministic algorithm to compute which cards should be discarded for any given hand. However, before using these arbitrary judgements, a Monte-Carlo method was programmed to determine optimal throwaway strategies for every type of hand. The method used to achieve this was based on an idea of Findler's, (10). The best throwaway combination for a particular hand was determined by considering all possible throwaway combinations, improving the hand randomly a large number of times ($\simeq$ 20,000) for each of these, and then choosing that throwaway which gave the best result. The results obtained agreed exactly with the throwaway strategies advocated by experienced players (30). However, the time taken to compute each case using this method was approximately 20 seconds of central processor (c.p.) time†, and hence a deterministic algorithm was tried.

† All computation was carried out using a CDC 6400 computer, and times taken apply to this machine (see (4)).

This new algorithm worked in the following way. First a hand was classified according to its type, and then the appropriate cards to discard were chosen on the basis of the results that had been obtained using the Monte-Carlo method. This algorithm was faster by 2 orders of magnitude than the Monte-Carlo method, and it is described in more detail in the flow diagram given in figure 2.

2.8 Calculation of probabilities

The calculation of the probability of winning with a certain hand, both before and after the draw, is crucial in poker. Findler, (10), suggests a Monte-Carlo method. This was tried (details will be given later), but found to be very slow. Findler in later work, (37,38), suggests alternative methods, but details of these could not be obtained in time for incorporation in this work. Alternatively, Epstein, (9), derived the table of probabilities given in table 2. However these are not sufficient in themselves, because although the table gives the probability of obtaining any given hand and the probability of improving to any better hand for any given number of cards drawn (discarded), it does not give the corresponding probabilities of winning. As a table giving the required probabilities could not be found, they were evaluated in the following way.

FIGURE 2

ALGORITHM TO DETERMINE WHICH CARDS TO DISCARD DURING HAND IMPROVEMENT

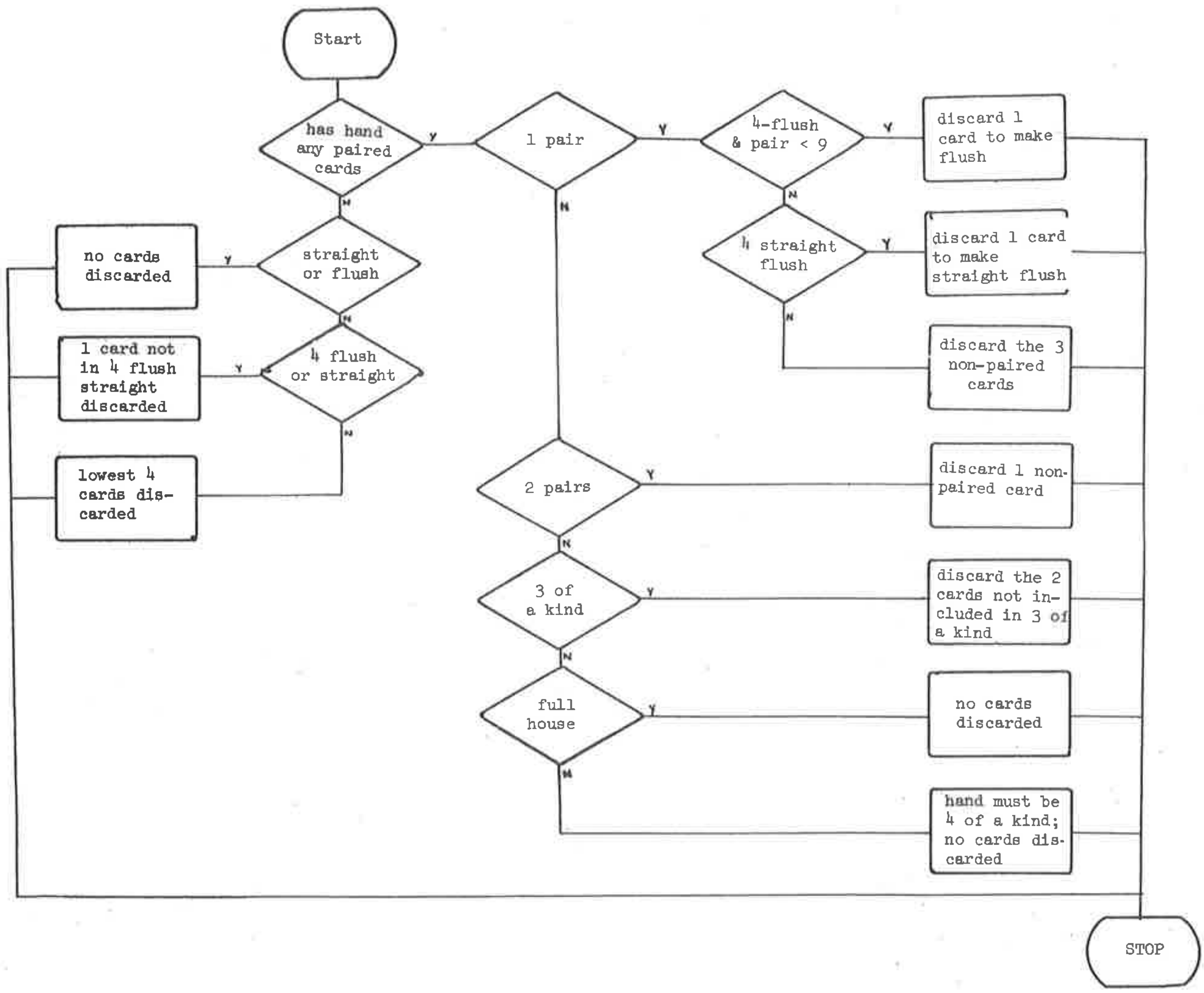


TABLE 2

EPSTEIN'S TABLE OF POKER PROBABILITIES

Original hand	Probability of receiving that hand	Cards drawn	Improved hand	Probability of receiving improved hand
One pair	0.4226	3	two pairs	0.16
		3	3 of a kind	0.114
		3	full house	0.0102
		3	4 of a kind	0.0028
		3	any improvement	0.287
		2	2 pairs	0.172
		2	3 of a kind	0.078
		2	full house	0.0083
		2	4 of a kind	0.0009
		2	any improvement	0.26
two pairs	0.0475	1	full house	0.085
3 of a kind	0.0211	2	full house	0.061
		2	4 of a kind	0.043
		2	any improvement	0.104
		1	full house	0.064
		1	4 of a kind	0.021
		1	any improvement	0.085
straight	0.0039	0	cannot be improved	0.0039
flush	0.0020	0	" " "	0.0020
full-house	0.0014	0	" " "	0.0014
4 of a kind	0.00024	0	" " "	0.0024
straight flush	0.000014	0	" " "	0.000014
royal flush	0.00000123	0	" " "	0.00000123
4 straight (open)	0.035	1	straight	0.17
4 straight (gap)	0.123	1	straight	0.085
4 flush	0.043	1	flush	0.191
4 straight flush	0.000123	1	straight flush	0.043
		1	any straight or flush	0.319

Let h be some unimproved poker hand, then $f_b(h)$ is the probability that the hand h has of beating any other random unimproved hand. Now if h is some already improved poker hand then define $f_a(h)$ as the probability that the hand h has of beating any other random improved hand. A quick method of evaluating $f_a(h)$ and $f_b(h)$ is required.

Findler, (10), proposed that a Monte-Carlo method be used. This method was programmed but found to be slow (with each calculation taking about 5 secs. of c.p. time).

INSET GOES HERE.

Findler in a later publication, (37), deals with the calculation of probabilities in a more sophisticated way. However, since this work was not available at the time of writing the simulation described here, use of these new methods could not be made. This does not affect the results of this work as the exact method of calculating probabilities is not of central importance. The approach to the calculation of probabilities adopted in this study is a two-phase one, and will now be described.

approximately determined thus. First, a large number, $N (N \simeq 30{,}000)$, of random hands are generated. Second, L, the number of times that the given hand h beats the randomly generated hands is noted, and $f_b(h)$ is approximated by L/N. Even though the implementation of this method is straightforward, the calculation must be carried out in such a way that the execution time is kept to a reasonable level. Because of this constraint various complications arise, and the technical details of this process are presented in appendix C.

The Monte-Carlo method used to evaluate $f_a(h)$ differs from the method given above only in that the hand h is tested against a large number of randomly improved hands (see appendix C).

The interpolative phase of this process will now be described.

Table 3 defines 9 intervals in which any unimproved hand h may lie. For each of these intervals the table gives h_i^1 which is the lowest hand of the i-th interval, and h_i^2 which is the highest hand of this interval (inclusive). The corresponding probabilities $f_b(h_i^1)$ and $f_b(h_i^2)$, as calculated by the Monte-Carlo method are also given.

Consider, for example, the second interval. This interval includes all hands that contain exactly 1 pair. By considering all hands of this type it can be seen that the lowest possible hand of this type is 2C 2D 3C 4C 5C while the highest hand is AC AS KC QC JC.

The probability of winning that any unimproved hand h has may be approximated from this table in the following way. First it is required to determine the interval i to which this hand belongs and corresponding to this interval there is a numbering function p_i (see table 4) that assigns an integer $p_i(h)$ to each hand h belonging to the interval. This function p_i is chosen in such a

TABLE 3

PROBABILITIES OF WINNING FOR UNIMPROVED HANDS CALCULATED BY A MONTE-CARLO METHOD

Interval No.	h_i^1: Low hand of interval	$f_b(h_i^1)$	h_i^2: high hand of interval	$f_b(h_i^2)$
1	7C 4S 3D 2D 5S	0.0000	AC KC QC JC 9D	0.4945
2	2C 2D 3C 4C 5C	0.5010	AC AS KC QC JC	0.9198
3	3C 3S 2C 2S 4C	0.9205	AC AS KC KS 2C	0.9711
4	2C 2S 2D 4C 5S	0.9711	AC AS AD 2C 3S	0.9920
5	5C 4D 3S 2C AS	0.9940	AC KD QS JC 10C	0.9944
6	7C 6C 5C 4C 2C	0.9964	AC KC JC 10C 9C	0.9975
7	2C 2S 2D 3S 3C	0.9982	AC AS AD KC KS	0.9993
8	2C 2S 2H 2D 3S	0.9993	AC AS AD AH KS	1.0000
9	5C 4C 3C 2C AC	1.0000	AC KC QC JC 10C	1.0000

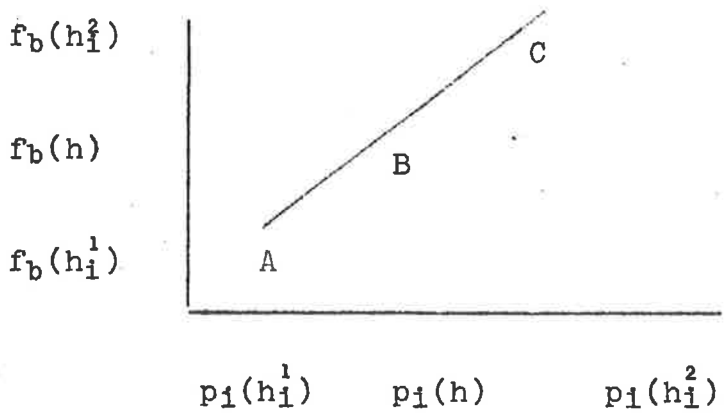
Note that even though suits do not influence the strength of a poker hand, they are included in the above table for greater clarity.

TABLE 4

HAND NUMBERING FUNCTION $p_i(h)$

Number, i	Hand type	variable description	$p_i(h)$ for h in group i
1	nothing	i1=numerical value of highest card	$p_1(h) = i1$
2	1 pair	i1=value of pair	$p_2(h) = i1$
3	2 pair	i1=value of higher pair i2=value of lower pair	$p_3(h) = \frac{(i1-2)(i1-3)}{2} + i2$
4	3 of a kind	i1=value of 3 of a kind	$p_4(h) = i1$
5	straight	i1=value of high card (the one exceptional case when Ace is counted as 1 rather than 14 occurs when the Ace forms the first card of a straight	$p_5(h) = i1$
6	flush	i1=value of high card	$p_6(h) = i1$
7	full-book	i1=value of 3 of a kind i2=value of 2 of kind	$p_7(h) = 12(i1-2)+i2$
8	routine	as for straight with i1=value of high card	$p_8(h) = i1$

way that the value of $f_b(h)$ may be approximated by a linear interpolation between $f_b(h_i^1)$ and $f_b(h_i^2)$, see diagram below, where the gradients AB and BC may differ slightly.



By assuming that the gradients AB and BC are equal it may be found that

$$f_b(h) = f_b(h_i^1) + \frac{p_i(h) - p_i(h_i^1)}{p_i(h_i^2) - p_i(h_i^1)} \cdot [f_b(h_i^2) - f_b(h_i^1)]$$

where $f_b(h)$ approximates the required probability.

For example, in the interval

$i=2$, where h_i^1 = 2C 2D 3C 4C 5C

and h_i^2 = AC AS KC QC JC

it is natural to choose $p_i(h)$ equal to the numerical value of the pair, where the ace is counted as 14 and it follows that $p_i(h_i^1) = 2$ and $p_i(h_i^2) = 14$.

Hence, given some hand h in this interval, say,

h = 6C 6D 7C 2D KS

$f_b(h)$ is approximated by

$$f_b(h_1^1) + \frac{p_1(h) - p_1(h_1^1)}{p_1(h_1^2) - p_1(h_1^1)} . [f_b(h_1^2) - f\ (h_1^1)]$$

$$= 0.5010 + \frac{6 - 2}{14 - 2} . [0.9198 - 0.5010]$$

$$= 0.6263.$$

Note that the failure of this approximation to distinguish, for example, between 6C 6D 7C 2D KS and 6H 6S AC QS KH, is unimportant because the 3 non-paired cards in each hand are later to be discarded, and thus may be safely ignored.

Winning probability for an already improved hand, $f_a(h)$.

The winning probability for an already improved hand $f_a(h)$, is found by the same interpolative method as was used to find $f_b(h)$ except, in this case table 5 defines the intervals used.

2.9 Betting strategies

The betting may be divided into 2 phases, betting before the draw, and betting after the draw.

In betting before the draw each player decides, in turn, whether to ante 2 units and play, or whether to drop out of the hand. This decision is determined by the cards that a player holds, the number of players who have already anted, the number of players still to decide, and the state of the game.

When this program was first written, a very simple

TABLE 5

PROBABILITIES OF WINNING FOR IMPROVED HANDS CALCULATED BY A MONTE-CARLO METHOD

Interval no.	h_i^1: low hand of interval	$f_a(h_i^1)$	h_i^2: high hand of interval	$f_a(h_i^2)$
1	7C 4S 3D 2D 5S	0.0000	AC KC QC JC 9D	0.5011
2	2C 2S 3C 4S 5C	0.5222	AC AS KS QC JC	0.7983
3	3C 3S 2C 2S 4C	0.7997	AC AS KC KS 2C	0.9097
4	2C 2S 2D 4C 5C	0.9100	AC AS AD 2C 3S	0.9716
5	5C 4D 3S 2C AS	0.9778	AC KD QS JC 10S	0.9820
6	7C 6C 5C 4C 2C	0.9820	AC KC QC JC 9C	0.9846
7	2C 2S 2D 3S 3C	0.9852	AC AS AD KC KS	0.9967
8	2C 2S 2H 2D 3S	0.9967	AC AS AH AD KS	0.9997
9	5C 4C 3C 2C AC	0.9999	AC KC QC JC 10C	1.0000

criterion for entering or dropping out was used. However, this was discarded, because later work in this thesis (see 5.5) showed that there was a better approach to this problem, and since little extra work was involved, the program was modified to operate on these improved principles.
The strategy, as used in the program, is most conveniently considered in two separate parts.

(1) <u>The minimum hand required to enter the game when no other player has yet entered.</u>

Suppose that a player holds hand h, no other player has yet entered the game, and remaining players have yet to decide. Then, (as shown in chapter 5) it is best to enter the game only if $f_b(h) \geqslant f(n)$ where

$$f(n) = 0.01n^2 + 0.13n + 0.5 \qquad (1)$$

where n is the number of players.

(2) <u>Minimum hand required to enter if at least one other player has already entered.</u>

The minimum hand required to enter if at least one other player has already entered is calculated in the following way. A player should only enter if his hand h is such that it is significantly better than the minimum hand, H, that the last player to enter is expected to have. More specifically, as shown in chapter 5, a hand h is good enough if $f_b(h) \geqslant g(\ell)$ where

$$\ell = f_b\,(H)$$

and $$g(\ell) = \ell + \tfrac{1}{3}(1-\ell). \qquad (2)$$

The practical application of this idea is illustrated by means of a simple example.

Consider a 7-player game where the players hold hands where probabilities of winning, $f_b(h)$, are as given below.

Player number	Number left to declare	$f_b(h)$
1	6	0.97
2	5	0.99
3	4	0.87
4	3	0.96
5	2	0.52
6	1	0.76
7	0	0.84

The decisions made by each player are summarised and explained in table 6.

Player n (n=7 in the example above), is called the "blind", and according to the rules of the game has 3 options available to him, to drop out, play on, or double. He chooses to select his strategy in the following way (see chapter 5).

<u>Drop</u> if $f_b(h) < \ell/2$

<u>Play on</u> if $\ell/2 \leqslant f_b(h) \leqslant \ell + \frac{1}{2}(1-\ell)$

<u>Double</u> if $\ell+\frac{1}{2}(1-\ell) < f_b(h)$

TABLE 6

AN EXAMPLE OF THE BETTING ALGORITHM

Player number	Decision made	Reason	ℓ, the probability of the minimum required hand held by last player to enter
1	enters	because $0.97 \geq f(6) = 0.92$	ℓ becomes 0.92
2	enters	because $0.99 \geq g(\ell) = g(0.92) = 0.96$	ℓ becomes 0.96
3	drops	because $0.87 < g(\ell) = g(0.96) = 0.97$	ℓ remains at 0.96
4	drops	because $0.96 < g(\ell) = g(0.96) = 0.97$	ℓ remains at 0.96
5	drops	because $0.52 < g(\ell) = g(0.96) = 0.97$	ℓ remains at 0.96
6	drops	because $0.76 < g(\ell) = g(0.96) = 0.97$	ℓ remains at 0.96
7	player 7, the "blind", selects his strategy in a different way, which is described on the previous page.		

It is also shown in chapter 5 that following a double it is best for the players still in the game to continue playing and this strategy is used here.

Second round of betting

Now the second round of betting will be considered. As this part of the game has not been solved analytically due to its complexity, the following strategy, partially based on ideas proposed by Findler, (10), and Coffin, (6), is used in this program.

Let $p_a^n(h)$ be the probability that a hand h has of winning, after the draw, against n other players, where $p_a^n(h) = [f_a(h)]^n$. Suppose that the size of the pot is p, q is the amount required to look, and $\varepsilon=1$ is some constant. Then the strategy followed is:

a player drops if $p_a^n(h).(p+q) < q-\varepsilon$

a player looks if $q-\varepsilon \leqslant p_a^n(h).(p+q) \leqslant q+\varepsilon$

a player raises by

$p_a^n(h).(p+q) - q$ units if $q+\varepsilon < p_a^n(h).(p+q)$ (3)

The strategy is based on a computation of a player's expectation for a given hand (expectation equals probability of winning × total size of pot), which is then compared with q, the amount required to look. As this expectation is below, approximately equal to, or above q, so a player drops, looks or raises. Obviously the value of ε can be adjusted so that a player will raise more or less frequently.

This very simple algorithm does not include bluffing although this was included in an earlier version of the program, but removed because the program was too slow (see next section).

Programming

The implementation (programming) of the simulation was a simple though tedious matter, and hence further details will not be given here. It was experimentally found that each game simulation took approximately $\frac{1}{10}$ sec. of central processor (c.p.) time.

DISCUSSION

2.10 Checking of results from computer runs

Before the results obtained were used, a check was carried out to confirm that the program was functioning correctly. This was done by simulating a number of games (about 60) and printing out all the relevant details of each game. It was found that the plays made in each game were consistent with the logic used in the simulation.

2.11 Interactive poker

The program was set up to play poker interactively, with a human player able to take the part of one or more of the simulated players. Communication between the players and the machine was via a screen and keyboard terminal. Figure 3 below illustrates the main operating details of this program.

FIGURE 3

INTERACTIVE POKER SIMULATOR

START

↓

Machine shuffles and deals the cards to all players, where human players have their hands displayed on the screen. Special arrangements are made so that each player's hand is not revealed to the other players.

↓

Players, in turn, nominate whether they wish to play, or not. The machine is informed, via the keyboard, which players decide to ante, while the actions of the simulated players are described via the screen. Doubling, if it arises, is handled in the same way.

↓

All players still in the game are allowed to improve their hands. The human players must type in which cards they wish to discard, and their replacement cards are shown on the screen.

↓

The final round of betting takes place, with all relevant bets displayed on the screen. The simulated player's bets are computed by the program, while the human players must key in their bets on cue from the screen.

↓

The machine calculates the winner, and computes amounts won and lost by each player. Then a new game is commenced.

Because of the following practical difficulties this work on interactive poker could not be carried to a satisfactory conclusion (a realistic poker game involving human players).

(1) Poker needs to be played with real money for the results to have any practical significance. This would not only have been difficult to arrange, but it was felt that this direction of research was outside the intended scope of the work, which deals primarily with methods of finding optimal strategies for poker-like games.

(2) Poker is always played in sessions ranging from several hours up to 12 hours or more. Indeed, good players often vary their style of play periodically in order to create uncertainty in the other players, and this tactic is of great importance in the game of poker.

However, even if it were possible to overcome objection (1), it was not possible to obtain the computer for the many long, uninterrupted sessions which would have been required in order to conduct such an experiment, as no interactive time sharing system was available.

Nevertheless, the results obtained with the interactive poker simulator suggest that it would be possible to continue work in this area, using the interactive poker program developed here as a starting point.

2.12 Investigation of poker strategies by simulation

It is theoretically possible to investigate poker strategies by simulation in the following way.

First note that the strategy of each player in the simulation model presented here depends on the functions $f(n)$ and $g(\ell)$ (see eqs.1 and 2), and the arbitrarily chosen parameter ε (see eq.3). Consider the parameter ε.

By keeping the parameters ($f(n)$, $g(\ell)$ and ε) of all players constant, one player could be left free to vary the parameter ε as he chose. If a sufficiently large number of games were simulated for each different choice of ε, he could determine that ε which gives him the best overall results under the given conditions.

However, this approach has several disadvantages. First, when a simulation of this type was tried on a simplified version of this game (see chapter 4) it was found that a large number of games (several thousand) needed to be simulated with any particular set of parameters to determine the expectation to any reasonable degree of accuracy (say to within 1% of the exact expectation which, in this case, could also be calculated analytically).

But since the game considered here is more complicated than this, and since each hand takes approximately $\frac{1}{10}$ sec. of c.p. time to simulate, the total amount of c.p. time required to compute each of these expectations would be high (several hundred seconds). Now, consider the following additional factors.

(i) This whole process would need to be repeated, in turn, for each player.

(ii) The strategies used by each player are inter-related and so (i) above would need to be repeated until stable strategies were obtained for all players concurrently.

From the above analysis it becomes clear that this method would not be practicable (too much computer time would be required), even if, as has been done here, the poker program was kept as simple as possible (i.e. bluffing was not introduced and only one variable, ε, was considered in the hypothetical experiment).

This failure does not imply that a more sophisticated approach to this problem might not drastically reduce the computation time. But such work would be sufficiently complicated to form a large, complete area of research, and it was decided that this lay outside the scope of this present work.

2.13 Concluding remarks

It was shown that investigation of poker by simulation is theoretically feasible, but, without further development, far too time consuming using the methods described here. The program developed here reduced the c.p. running time by working out probabilities and hand improvement in the fastest possible way. Evidently, further progress in simulation must follow these and other lines if these methods are to be at all practicable, (see Findler (37,38)).

It became clear that the amount of work required to achieve effective progress in simulation would be likely to be extensive. There was a choice of continuing with the research on simulation or of attacking the problem using game theoretic methods. After preliminary investigation it was realized that it would be impossible to do both, and thus the game theoretic approach was chosen as it seemed likely to yield useful results more quickly.

Findler's poker group is currently working in this area, (37,38), and are obtaining some interesting result.

...ed out on simulation

and could form a basis for further research.

Finally, the possibility of using the poker simulator to play interactive poker was demonstrated. Although this idea has no doubt been considered before details of experimental work in this area have not previously been published. Interactive poker is an important part of poker simulation since it may be used to validate experimentally the effectiveness of strategies calculated either by simulation or by other methods, by testing them under conditions of actual play.

CHAPTER 3

LOOKAHEAD (LAH) ALGORITHM

3.1 An Introduction

It was shown in chapter 2 that simulation could not be conveniently used to study the game of poker. An alternative method of approaching this problem is to construct mathematical poker models then solve these by game theoretic methods. When this approach was tried it was found that only the 2-person version of the particular game formulated could be solved by using standard methods (as found in the literature). It became evident that more realistic versions of the game, because of their greater complexity, could not be solved algebraically, by available methods, and thus numerical iterative methods were considered.

The only numerical method described in the literature which is relevant to this study, [Rosen, (31)], is iterative and requires derivatives which cannot be obtained for some of the games treated here. Thus a modified version of Rosen's method was suggested which does not require derivatives, but it was found to be slow for the purposes of this research.

As a result a new iterative method, called the Lookahead (LAH) algorithm, was formulated. Its speed of execution was shown to be at least 3 times as fast as

Rosen's method when applied to problems of the type that are treated in this study.

This chapter will be organised in the following way. A general statement of the problem will be given. Next, two numerical methods of solving games will be described. The first of these is Rosen's method mentioned above, and the second, the Lookahead (LAH) algorithm, which is the main theme of this chapter.

3.2 Mathematical Statement of Problem

The games that arise in this study fall into the category of n-person non-cooperative games, and may be defined in the following way. If, in a game Γ, with n players, each player acts purely in his own individual self interest, and coalitions between players are not allowed, then Γ is defined to be a n-person non-cooperative game. Consider such a game Γ when the players are denoted by the integers $i = 1,2,...n$. The following definitions relating to this game Γ will be used.

A strategy is the specification of the courses of action that a player will adopt in all given situations with which he can be presented. Let S_i, the strategy space for player i, be the set of all possible strategies available to player i.

Let $\alpha^i \in S_i$ be some particular strategy being used by player i. When the n players are using strategies $\alpha^1,\ldots\alpha^n$ let $\alpha = [\alpha^1,\ldots\alpha^n]^T$ be the current strategy vector. If $S = S_1 \otimes \ldots \otimes S_n$ is the cartesian product of $S_1,\ldots S_n$ then it follows that $\alpha \in S$. Each player i has a real valued payoff function $f_i(\alpha)$ which assigns a unique payoff for every $\alpha \in S$. Hence, $f_i(\alpha)$, the payoff to player i, is not only a function of his own strategy α^i, but also depends on the strategies $\alpha^1,\ldots\alpha^n$ followed by the other players.

The variable α may be constrained by p constraint relationships of the form

$$g_j(\alpha) \geqslant 0 \quad \text{for} \quad j=1,\ldots p$$

where the functions $g_1,\ldots g_p$ are real valued functions defined over the set S. Furthermore, each strategy $\alpha^i \in S_i$ is defined to be a vector with m_i real components $\alpha^i = [\alpha^i_1 \ldots \alpha^i_{m_i}]^T$.

Equilibrium points and optimal strategies

Nash, (25), introduced the concept of the equilibrium point (e.p.), which is commonly used to characterize optimality in the general n-person non-cooperative game. Nash's definition of optimal strategy is applicable to the games considered in this study, and will therefore be used, and is as follows:-

Let α^i, $(i=1,\ldots n)$ be an optimal strategy for player i in the game Γ, where $\alpha = [\alpha^1,\ldots\alpha^n]^T$ is the optimal joint strategy. Then Nash states that player i may not change his strategy α^i without suffering a decrease in his payoff function $f_i(\alpha)$ and describes this situation as occurring at an equilibrium point (e.p.). For example, if player i changes his strategy from α^i to α^{i*}, the new joint strategy is

$$\alpha^* = [\alpha^1,\alpha^2,\ldots\alpha^{i*},\ldots\alpha^n]^T,$$

and Nash's definition of the e.p. requires that $f_i(\alpha) \geqslant f_i(\alpha^*)$.

If, at the end of the game Γ, total gains equal total losses, then Γ is said to be a zero sum game.

i.e. $\sum_{i=1}^{n} f_i(\alpha) = 0$ for all $\alpha \in S$.

If $\left.\frac{\partial f_i}{\partial \alpha^i_j}\right|_\alpha = 0$ for $i=1,\ldots n$ and $j=1,\ldots m_i$ then Nash, (24), states that the joint strategy α is an e.p.

<u>Solving n-person games</u>

For simple games, exact analytic solutions are obtainable (see chapter 4). However, for more complicated games, numerical iterative methods are required, as discussed in the next section.

3.3 Rosen's method of numerical solution

Rosen's method approaches the e.p. by stepping in the direction of increasing gradient (computed from the derivatives of each player's payoff function). In this study it is not possible to calculate derivatives and consequently the approximation

$$f'(x) \simeq \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}, \quad \Delta x \text{ small}$$

is used. Since this is the only significant change to this algorithm further details will not be given, but may be found in (23).

This algorithm was able to solve the 2-person poker-like game formulated in chapter 4, but it was slow (see 3.9). Since, later, 4-person versions of the 2-person game were to be treated, speed of execution was crucial.

3.4 The Lookahead algorithm

The Lookahead (LAH) algorithm was formulated as an alternative to Rosen's method. It is an iterative method that solves a game by approaching the e.p. in a series of steps. When implemented on the computer it was found to be 3 times faster than the modified version of Rosen's method as used here. The algorithm is based on looking ahead in the same way that a chess player considers the replies open to his opponent, before making his own move.

The algorithm is first applied to a 2-person game and shown to give an e.p. consistent with Nash's definition.

This algorithm is then generalized to the n-person case. Finally, two problems with known solutions are solved correctly.

3.5 The LAH algorithm in a 2-person game

Consider a 2-person non-cooperative game where players 1 and 2 control respectively the variables α^1 and α^2. The joint strategy is $\alpha = [\alpha^1, \alpha^2]^T$ and the payoffs $f_1(\alpha)$ and $f_2(\alpha)$ depend for each player both on his strategy and on that of his opponent. Suppose that player 2 has fixed α^2 and player 1 is considering a change in α^1. Player 1 has no control over player 2. Accordingly he considers the effect of his opponent's response first to a small increment in α^1, and then a small decrement. With this knowledge he chooses the course most likely to maximize his return. Player 2 responds to the choice made by player 1, in a similar way. The players continue to alter their variables in turn until a stable situation is reached from which neither player is prepared to deviate.

A simple numerical example will now be given to show how this algorithm works in practice. Suppose in the game described above the rules are such that $0 < \alpha^1 \leqslant 1$, $0 < \alpha^2 \leqslant 1$ and the payoff functions are given by:-

$$f_1(\alpha) = 5\alpha^1 + 7\alpha^2 - 12\alpha^1\alpha^2$$

$$f_2(\alpha) = -f_1(\alpha).$$

Assume that each player can only increment or decrement his variable by $\pm\Delta$ where $\Delta = 0.1$, and hence α^1, α^2 can only assume the discrete values 0.1,... 0.9, 1.0. Table 7 below gives all possible payoffs $f_1(\alpha)$ in these circumstances.

TABLE 7

AN EXAMPLE OF THE LAH ALGORITHM

PLAYER 2

value of α^2

Player 1 value of α^1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.1	1.08	1.66	2.24	2.82	3.40	3.98	4.56	5.14	5.72	6.30
0.2	1.46	1.92	2.38	2.84	3.30	3.76	4.22	4.68	5.14	5.60
0.3	1.84	2.18	2.52	2.86	3.20	3.54	3.88	4.22	4.56	4.90
0.4	2.22	2.44	2.66	2.88	3.10	3.32	3.54	3.76	3.98	4.20
0.5	2.60	2.70	2.80	2.90	3.00	3.10	3.20	3.30	3.40	3.50
0.6	2.98	2.96	2.94	2.92	2.90	2.88	2.86	2.84	2.82	2.80
0.7	3.36	3.22	3.08	2.94	2.80	2.66	2.52	2.38	2.24	2.10
0.8	3.74	3.48	3.22	2.96	2.70	2.44	2.18	1.92	1.66	1.40
0.9	4.12	3.74	3.36	2.98	2.60	2.22	1.84	1.46	1.08	.70
1.0	4.50	4.00	3.50	3.00	2.50	2.00	1.50	1.00	.50	.00

Suppose that initially $\alpha^1 = 0.4$ and $\alpha^2 = 0.3$, and that each player is aware of what the other is doing.

Player 1 considers the following 3 possibilities.

(1) Player 1 chooses $\alpha^1=0.4$

In this case player 2, knowing that $\alpha^1=0.4$ will attempt to maximize his payoff $f_2(\alpha)$ by changing α^2 to 0.2, 0.3 or 0.4. Since player 2's payoffs for these 3 choices are -2.44, -2.66, and -2.88 respectively, [as $f_2(\alpha) = -f_1(\alpha)$], he will naturally choose $\alpha^2=0.2$, as this maximizes his payoff.

Hence if player 1 lets $\alpha^1=0.4$, he may expect player 2 to choose $\alpha^2=0.2$. Thus the expected payoff of player 1 is 2.44. The algorithm employs only 1 degree of look ahead because it was experimentally shown that this is sufficient for the algorithm to converge when dealing with the class of games encountered in this study.

(2) Player 1 chooses $\alpha^1=0.3$

Using similar reasoning to that above player 1 calculates that if $\alpha^1=0.3$, he may expect player 2 to choose $\alpha^2=0.2$. Thus his expected payoff in this situation is 2.18.

(3) Player 1 chooses $\alpha^1=0.5$

Player 1 computes his expected payoff to be 2.7 in this case.

Hence player 1 realizes that his best strategy is to choose $\alpha^1=0.5$, as then his expected payoff is maximized at 2.7.

Now it is player 2's turn to repeat this same process. The starting point this time is $\alpha^1=0.5$ and $\alpha^2=0.3$. Player 2 considers the following 3 possibilities.

(1) Player 2 chooses $\alpha^2=0.3$

In this case player 2 calculates that he may expect player 1 to choose $\alpha^1=0.6$ (as this maximizes player 1's payoff). Thus the expected payoff for player 2 will become -2.94.

(2) Player 2 chooses $\alpha^2=0.2$

Now player 1 may be expected to choose $\alpha^1=0.6$, making player 2's expected payoff -2.96.

(3) Player 2 chooses $\alpha^2=0.4$

Again player 1 may be expected to choose $\alpha^1=0.6$, making player 2's expected payoff -2.92.

Hence player 2's best move is to choose $\alpha^2=0.4$, as $-2.92 \geq \max\{-2.94, -2.96, -2.92\}$. The solution point now becomes $\alpha^1=0.5$ and $\alpha^2=0.4$.

In this way the solution point will keep moving through the game matrix until it reaches $\alpha^1=0.6$ and $\alpha^2=0.4$, from which point neither player is willing to deviate. Now the grid size Δ is diminished by some fraction $\sigma(0 < \sigma < 1)$, and the process repeated until another stable point is reached. The grid size is again diminished, and this process continues until Δ becomes smaller than some predetermined value e. It will be shown in 3.6 that in certain cases when the payoff functions are sufficiently smooth, the final point reached will be consistent with the equilibrium point of Nash.

Often problems in game theory contain constraint relationships, and these are handled as follows.
If at any time a variable is altered in such a way that it no longer satisfies the constraint relationships then it is moved back towards its original value until the constraint relationship is again satisfied.

3.6 Equivalence between LAH solution and e.p.

In this section it will be shown that for a 2 person game, where each player controls 1 variable, and where the payoff functions are sufficiently smooth, there is an equivalence between the e.p. of Nash, and the solution found using the LAH algorithm.
Consider the 2 person game defined in 3.5. Let the grid size Δ, used in the LAH algorithm be very small, and assume that the payoff functions, in the vicinity of some point $\alpha = [\alpha^1, \alpha^2]^T$ can be approximated by the planes

$$f_1(\alpha) \simeq a_{11}\alpha^1 + a_{12}\alpha^2 + a_{13} \quad (1)$$

$$f_2(\alpha) \simeq a_{21}\alpha^1 + a_{22}\alpha^2 + a_{23} \quad (2)$$

which pass through the four grid intersection points adjacent to α. Then if $f_1(\alpha)$ and $f_2(\alpha)$ are sufficiently smooth

these approximations and their derivatives will converge uniformly to the functions $f_1(\alpha)$ and $f_2(\alpha)$ and their corresponding derivatives, as $\Delta \to 0$ (see D.C. Handscomb, (15)).

Hence the derivatives of the functions $f_1(\alpha)$ and $f_2(\alpha)$ may be closely approximated by the derivatives of these planes.

Differentiating eq.1 and 2, and taking the limit as $\Delta \to 0$

$$\frac{\partial f_1}{\partial \alpha^1} = a_{11} \quad \text{and} \quad \frac{\partial f_2}{\partial \alpha^2} = a_{22} \tag{3}$$

In this section some new notation will be used.

Let

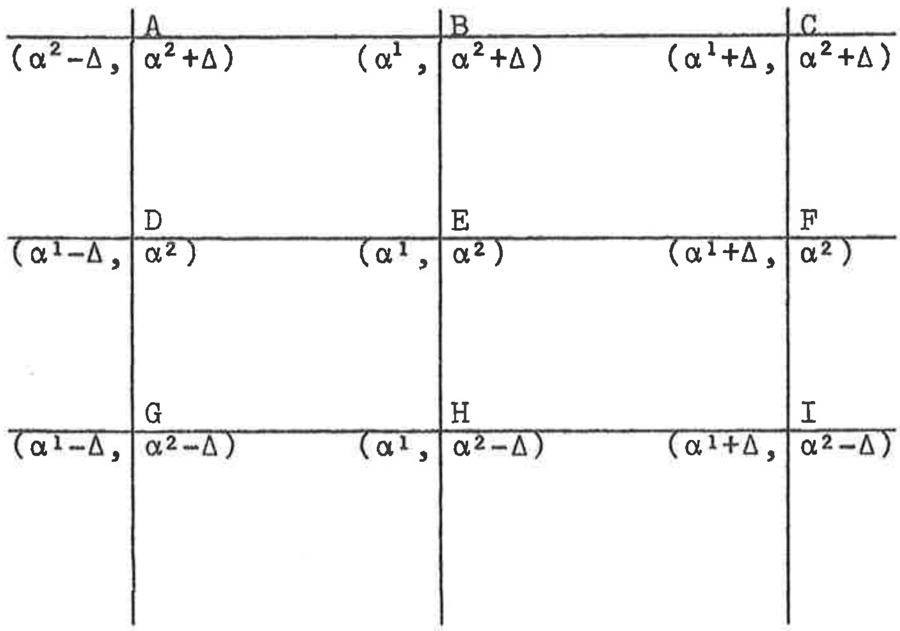
$$S_1(\alpha^1,\alpha^2) = \begin{cases} 1 \ ; & \dfrac{\partial f_1}{\partial \alpha^1} > 0 \\ 0 \ ; & \dfrac{\partial f_1}{\partial \alpha^1} = 0 \\ -1 \ ; & \dfrac{\partial f_1}{\partial \alpha^1} < 0 \end{cases}$$

Therefore from eq.3

$$S_1(\alpha^1,\alpha^2) = \begin{cases} \dfrac{a_{11}}{|a_{11}|} \ ; & a_{11} \neq 0 \\ 0 \ ; & a_{11} = 0 \end{cases} \tag{4}$$

Hence $S_1(\alpha^1,\alpha^2)$ is a constant, independent of α^1 and α^2.

Suppose that the LAH algorithm is being used on this problem, and it is the turn of player 1 to move. The current joint strategy point is at E in the diagram given below. Player 1 must decide whether to remain where he is at E, or move right to F, or move left to D.



Thus he will consider the 3 situations (a), (b) and (c) given below.

(a) Player 1 remains at E

Player 1 assumes that player 2 has 3 choices, each having the payoff as given below.

(i) Player 2 remains at E then his payoff is

$$f_2(\alpha^1,\alpha^2) = a_{21}\alpha^1+a_{22}\alpha^2+a_{23}$$

(ii) Player 2 moves to B then his payoff is

$$f_2(\alpha^1,\alpha^2+\Delta) = a_{21}\alpha^1+a_{22}(\alpha^2+\Delta)+a_{23}$$

(iii) Player 2 moves to H then his payoff is

$$f_2(\alpha^1,\alpha^2-\Delta) = a_{21}\alpha^1+a_{22}(\alpha^2-\Delta)+a_{23}$$

From this it can be seen that if $a_{22} > 0$ then player 2 moves to B, if $a_{22} < 0$ then player 2 moves to H, and if $a_{22} = 0$ then player 2 remains at F.

But as from eq.4

$a_{22} = 0$ implies that $S_2 = 0$

and $a_{22} > 0$ implies that $S_2 = 1$

and $a_{22} < 0$ implies that $S_2 = -1$,

the above situation can be expressed thus.

Player 1 will expect player 2 to

(i) remain at E if $S_2 = 0$
(ii) move to B if $S_2 = 1$
(iii) move to H if $S_2 = -1$

That is player 2 will move to $\alpha^2 + \Delta.S_2$

(b) Player 1 moves to F

Using the same argument as in (a) it can be shown that

player 2 will

(i) remain at F if $S_2 = 0$
(ii) move to C if $S_2 = 1$
(iii) move to I if $S_2 = -1$

That is player 2 moves to $\alpha^2 + \Delta.S_2$

(c) Player 1 moves to D

player 2 will

(i) remain at D if $S_2 = 0$
(ii) move to A if $S_2 = 1$
(iii) move to G if $S_2 = -1$

That is player 2 moves to $\alpha^2 + \Delta . S_2$

Hence, regardless of whether player 1 chooses E,F or D it may be seen that player 2 will move from α^2 to $\alpha^2 + \Delta . S_2$.

Thus, player 1 may now calculate his expected payoffs in the 3 situations

(a) <u>player 1 remains at E</u>

expected payoff $f_1(\alpha^1, \alpha^2+\Delta . S_2) = a_{11}\alpha^1 + a_{12}[\alpha^2+\Delta . S_2] + a_{13}$

(b) <u>player 1 moves to F</u>

expected payoff $f_1(\alpha^1+\Delta, \alpha^2+\Delta S_2) = a_{11}(\alpha^1+\Delta) + a_{12}[\alpha^2+\Delta S_2] + a_{13}$

(c) <u>player 1 moves to D</u>

expected payoff $f_1(\alpha^1-\Delta, \alpha^2+\Delta S_2) = a_{11}(\alpha^1-\Delta) + a_{12}[\alpha^2+\Delta S_2] + a_{13}$.

It can be seen from the above equations that player 1 will maximise his payoff $f_1(\alpha)$ if he

remains at E if $a_{11} = 0$

moves to F if $a_{11} > 0$

moves to D if $a_{11} < 0$

It follows from eq. 4 that as $S_1 = \begin{cases} 1 \; ; \; a_{11} > 0 \\ 0 \; ; \; a_{11} = 0 \\ -1 \; ; \; a_{11} < 0 \end{cases}$ then, to maximize his return player 1 moves from α^1 to $\alpha^1 + \Delta . S_1$.

Hence, it has been shown that when Δ is sufficiently small player 1 will move in the direction of increasing gradient S_1, and will only cease to move when $S_1 = 0$, that is when the derivative $\frac{\partial f_1}{\partial \alpha^1} = 0$.

By a similar means it can be shown that when Δ is sufficiently small, player 2 will move in the direction of his increasing gradient, S_2, and will only cease to move when $S_2 = 0$, that is $\frac{\partial f_2}{\partial \alpha^2} = 0$. Thus, in the case of a 2 person game, with each player controlling 1 variable, and the payoff functions satisfying certain conditions, the solution found using the LAH algorithm will be such that the derivatives $\frac{\partial f_1}{\partial \alpha^1}$ and $\frac{\partial f_2}{\partial \alpha^2}$ equal zero at that point. Hence this point will also be an e.p. in the sense of Nash.

By similar means it may be proved that there is a similar equivalence between the 2 solutions for the case of an n-person game, with each player controlling any finite number of variables, provided that the payoff functions $f_i(\alpha)$ are sufficiently smooth. This will not be proved here as the proof used is similar to the one given above.

3.7 Generalisation of LAH algorithm to the n-person case

The LAH algorithm can be generalised to the situation described in 3.2, involving n players, where player i controls m_i variables.

Suppose that the solution point is currently at some point $\alpha = [\alpha^1, \ldots \alpha^n]^T$ and player i, who controls the variables $\alpha^i_1, \ldots \alpha^i_{m_i}$ must determine whether or not to alter any of his variables. Player i will treat each variable $\alpha_1^i,\ldots\alpha_{m_i}^i$ in the following way. First he will consider α_1^i and he will compute his expected payoff for the 3 possibilities,

(i) α_1^i altered to $\alpha_1^i+\Delta$

(ii) α_1^i altered to $\alpha_1^i-\Delta$

(iii) α_1^i remains unchanged.

He then chooses that value of α_1^i which promises him the highest expected payoff, which is calculated in the manner described below.

The expected payoff to player i for each of the above 3 possible choices of α_1^i is calculated as follows. Let each of the other n-1 players be considered in turn, and allow each one the option of temporarily changing any of the variables under his control by an amount $\pm\Delta$, in such a way that his own individual payoff is maximized. When this process has been completed for all players the joint strategy α is changed to some α^*, and the expected payoff of player i is given, for that particular initial choice of α_1^i, by $f_i(\alpha^*)$.

Player i repeats this process for all his other variables $\alpha_2^i,\ldots\alpha_{m_i}^i$.

In this way, starting with player 1, and finishing with player n, each player is given the opportunity of changing his variables, and this procedure constitutes one cycle of the algorithm.

These cycles are continued until the joint strategy α remains unchanged for 1 cycle. At this point the grid size Δ is made smaller by some factor σ, and the complete process is repeated with the new Δ. The algorithm stops when Δ becomes smaller than some pre-determined minimum grid size e.

A flow chart of this algorithm is given in figure 4.

3.8 Applications of the LAH algorithm

Three applications of the LAH algorithm to problems with known solutions were considered. In each case the correct result was obtained. Two of these problems were linear programs reformulated as games, (23), and will not be discussed here.

von Neumann's poker-like game

The rules of this game have already been given in chapter 1.

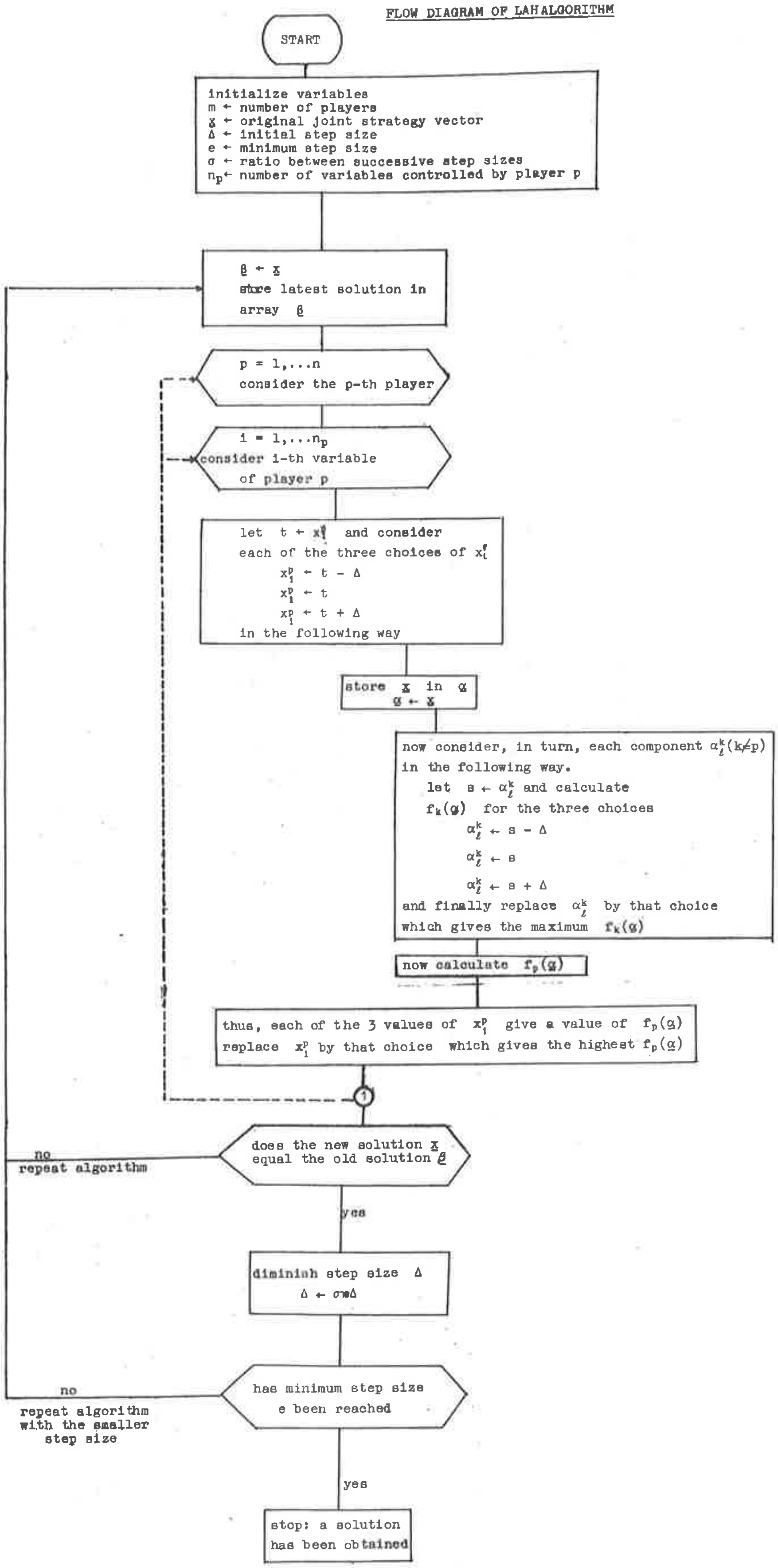
The game is solved in the following way. This solution will be given in some detail as the methods used play an important role in the games solved in chapters 4 and 5.

It is assumed that player 1 follows a strategy governed by the function $\phi_1(x)$, where $\phi_1(x)$ is the probability of making the larger bet, that is A units, for a given hand x.

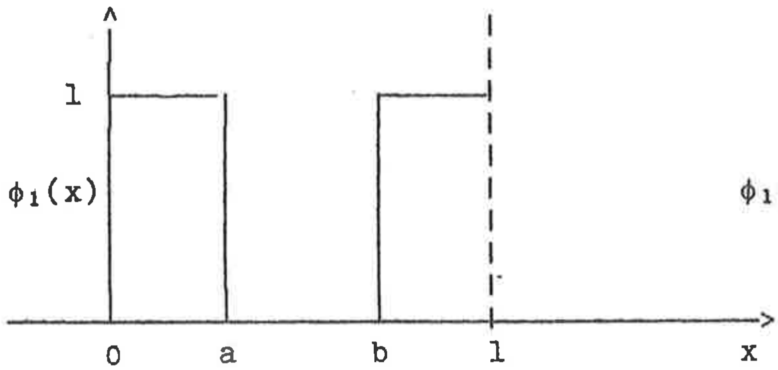
von Neumann shows that the rules for this game lead to the function $\phi_1(x)$ having the general form given below,

FIGURE 4

FLOW DIAGRAM OF LAH ALGORITHM



and this form of the function is used here.

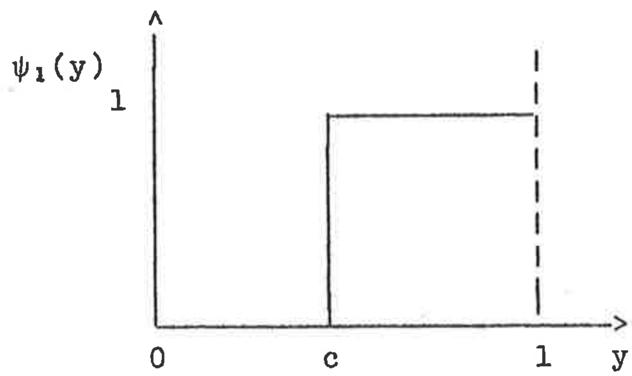


$$\phi_1(x)=\begin{cases}0; & a\le x\le b\\ 1; & x<a \text{ or } x>b\end{cases}$$

The above function, $\phi_1(x)$, describes a strategy in which player 1 bets A units if his hand is either very good, $(x>b)$ or very bad $(x<a)$. The latter bet on a low hand corresponds to the bluff. If the hand x is such that $a\le x\le b$ then player 1 bets B units.

The strategy for player 2 is given by the function $\psi_1(y)$, where $\psi_1(y)$ is the probability that player 2 looks at a bet of A units by player 1, if he (player 2) holds hand y. For similar reasons to that given above it is assumed that $\psi_1(y)$ is of the form

$$\psi_1(y)=\begin{cases}0; & y<c\\ 1; & y\ge c\end{cases}$$



This means that player 2 only looks at a bet of A units if his hand is reasonably good. Note that there is no opportunity for player 2 to bluff in this situation.

If both players follow the above strategies it is possible to calculate the expected payoffs to players 1 and 2 as functions $f_1(a,b,c)$ and $f_2(a,b,c)$ respectively, and as this is a zero sum game, (i.e. the losses of one player constitute the gains of the other)

$$f_2(a,b,c) = -f_1(a,b,c).$$

Before calculating the payoff functions a new term must be defined.

$$\chi_2^{w,\ell}(x,y) = \begin{cases} w; \text{ if } & x \geqslant y \\ \ell; \text{ if } & x < y \end{cases}$$

Now it is possible to list all the possible plays, compute their probability of occurring, and calculate the corresponding payoff to player 1 in the following way. Consider the play BB (i.e. player 1 bets B, and player 2 bets B).

The probability of player 1 betting B is $[1-\phi_1(x)]$, and the conditional probability of player 2 betting B is 1 (as player 2 is forced to bet B when player 1 bets B).

Therefore the probability of this play is $[1-\phi_1(x)]*1$, and the payoff to player 1 is $B\chi_2^{1,-1}(x,y)$ because if $x>y$, then player 1 wins B units

($\chi_2^{1,-1}(x,y) = +1$ for $x>y$) otherwise if 2's hand wins, i.e. $y>x$, then player 1 loses B units ($\chi_2^{1,-1}(x,y) = -1$ for $x<y$). The event $x=y$ of a draw has probability zero and so does not affect these calculations.

In this manner the 3 different possible plays with their probabilities and their payoffs may be calculated and are summarised in table 8.

TABLE 8 : SUMMARY OF PLAYS FOR VON NEUMANN'S GAME

Play I	Play II	Probability	Payoff to I
B	B	$1-\phi_1(x)$	B $\chi_2^{1,-1}(x,y)$
A	B	$\phi_1(x)[1-\psi_1(y)]$	B
A	A	$\phi_1(x)\psi_1(y)$	A $\chi_2^{1,-1}(x,y)$

Hence for any given hands x and y, dE, the expected payoff to player 1 is given by multiplying the payoffs for each different play by the probability of that play occurring and summing over all possible plays, i.e.

$$dE=B\chi_2^{1,-1}(x,y).[1-\phi_1(x)]+B\phi_1(x)[1-\psi_1(y)]+A\chi_2^{1,-1}(x,y)\phi_1(x)\psi_1(y)$$

The total expected payoff, $f_1(a,b,c)$, is obtained by integrating dE over the range $0\leq x\leq 1$ and $0\leq y\leq 1$, (for a similar calculation see Pruitt, (29).)

Thus

$$f_1(a,b,c) = \int_{x=0}^{1}\int_{y=0}^{1} \{B(1-\phi_1(x)\chi_2^{1,-1}(x,y)$$

$$+B\phi_1(x)(1-\psi_1(y))+A\phi_1(x)\psi_1(y)\chi_2^{1,-1}(x,y)\}dx\ dy$$

$$= \int_{x=0}^{1}\int_{y=0}^{1} B(1-\phi_1(x))\chi_2^{1,-1}(x,y)dx\ dy$$

$$+ \int_{x=0}^{1}\int_{y=0}^{1} B\phi_1(x)(1-\psi_1(y))dx\ dy$$

$$+ \int_{x=0}^{1}\int_{y=0}^{1} A\phi_1(x)\psi_1(y)\chi_2^{1,-1}(x,y)dx\ dy.$$

Now by using the definitions of $\phi_1(x)$ and $\psi_1(y)$ given earlier, it is possible to expand the above integrals as follows.

$$f_1(a,b,c) = \int_{x=a}^{b}\int_{y=0}^{1} B\chi_2^{1,-1}(x,y)dx\ dy$$

$$+ \int_{x=0}^{a}\int_{y=0}^{c} B\ dxdy + \int_{x=b}^{1}\int_{y=0}^{c} B\ dx\ dy$$

$$+ \int_{x=0}^{a}\int_{y=c}^{1} A\ \chi_2^{1,-1}(x,y)dx\ dy + \int_{x=b}^{1}\int_{y=c}^{1} A\ \chi_2^{1,-1}(x,y)dxdy$$

Thus $f_1(a,b,c) = BI_1+Bac+B(1-b)c+AI_2+AI_3$ where

$$I_1 = \int_{x=a}^{b} \int_{y=0}^{1} \chi_{\frac{1}{2}}^{1,-1}(x,y)\,dx\,dy$$

$$I_2 = \int_{x=0}^{a} \int_{y=c}^{1} \chi_{\frac{1}{2}}^{1,-1}(x,y)\,dx\,dy$$

$$I_3 = \int_{x=b}^{1} \int_{y=c}^{1} (x,y)\,dx\,dy$$

This integral presents some difficulty in evaluation, and occurs again in chapters 5 and 6 in a more complicated form. Because of this a general way was found of evaluating this type of integral, and is given in Appendix A. The results obtained in appendix A permit I_1, I_2 and I_3 to be evaluated, and it is found that:

$$I_1 = (b-a)(a+b-1)$$

$$I_2 = \begin{cases} -a(1-c) & ;\ a < c \\ -a(1-c)+(a-c)^2 & ;\ a \geqslant c \end{cases}$$

$$I_3 = \begin{cases} (1-c)(b-c); & b < c \\ (1-b)(b-c); & b \geqslant c \end{cases}$$

The constraint relationships governing the variables a,b and c are:

$$a \leqslant b, \quad \text{and} \quad a,b,c \in [0,1].$$

With the above information the problem was solved by means of the LAH algorithm, which gave

$$a = 0.1, \quad b = 0.7, \quad c = 0.4.$$

These results were consistent with those given by von Neumann in (27). By evaluating certain derivatives at the e.p. it may be shown that this optimal strategy also satisfies Nash's condition for an e.p.

3.9 Comparison between Rosen's modified method and the LAH algorithm

A 2-person poker-like game is formulated in chapter 4 which is typical of the class of games to be treated in this study. This 2-person game has a known exact solution, given in terms of variables a,b,d and e. The meanings of the variables are unimportant in the present discussion and are therefore not defined. Several comparisons of the modified Rosen's method and the LAH algorithm, as applied to this game, were made. Table 9 given below shows the example that was most favourable to the modified Rosen's method.

TABLE 9: COMPARISON BETWEEN ROSEN'S MODIFIED METHOD AND LAH ALGORITHM

	a	b	c	d	time (c.p.seconds)
exact soln.	.02980	.64177	.61267	.81056	-
LAH alg.	.02980	.64179	.61269	.81056	30
Rosen's modified method	.02803	.65770	.61448	.81973	94

It can be clearly seen from the above table that the LAH algorithm, even in this instance in which its performance was least good, achieves a more accurate solution than Rosen's modified method in only $\frac{1}{3}$ of the time.

For this reason the LAH algorithm is to be preferred for solving the class of games dealt with in this study.

CHAPTER 4

FORMULATION AND SOLUTION OF A 2-PERSON POKER-LIKE GAME

4.1 Introduction

Chapter 4 is concerned with solving a 2-person poker-like game (2-PG). The more difficult task of solving the 3 and 4 person versions of this game is left to chapter 5. This work has some significant features which are mentioned below.

(a) A search of the literature suggests that this is the first time that solutions to a poker-like game have been sufficiently realistic to predict strategies commonly used by experienced players.

(b) The solution found to this poker-like game is applicable not only to the analysis of poker, but also to certain types of business situations. Furthermore a network problem is solved by the methods developed in this study to obtain a new and interesting result. These two topics will be discussed in chapter 6.

Before proceeding to define and solve the 2-person poker-like game (2-PG), the original poker game on which the 2-PG is based will be discussed. From the rules given in chapter 3 it may be seen that this poker game may be conveniently divided into the 2 phases given below (familiarity with the poker game described in chapter 2 will be assumed in the remainder of this chapter).

Phase 1 of the poker game

Phase 1 of the poker game extends from the deal up to the improvement of hands, which is achieved by discarding unwanted cards and replacing them with new ones from the deck.

Phase 2 of the poker game

Phase 2 of the poker game consists of a round of betting, after which the winner is determined.

A relationship between phase 1 of the game and the entire game will now be established.

Consider the above poker game under the simplifying assumption that no further betting is to be allowed in phase 2, and call this the phase 1 game. Experienced players assume that a sound optimal strategy for the entire game must be based on a sound optimal strategy for the phase 1 game (see (30)). This is justified in the following way.

Observations of games played by experienced players have shown that most poker hands have no significant betting in phase 2. Thus, in these cases, the game reduces essentially to the phase 1 game and hence the phase 1 optimal strategy can be used. Now consider what will happen when a player uses the phase 1 strategy [redacted] where there is significant betting in phase 2.

Books on the subject (8,30), indicate that in this case the strategy followed in phase 1 is overshadowed by the strategy followed in phase 2. That is the elements of bluffing and poker psychology are paramount, and, to a large extent, the exact strategy followed in phase 1 loses its importance. The use of the phase 1 optimal strategy therefore, carries with it no significant penalty when applied to this case. For this reason the phase 1 optimal strategy can be used in the entire game (which consists of phase 1 followed by phase 2), otherwise losses will be sustained in the instances mentioned above, where there is no significant betting in phase 2.

The complexity of the game is such that its entire solution was considered infeasible, partly because of the difficulty of describing the payoff functions mathematically, and partly because of the difficulty of finding an e.p.

Therefore, the phase 1 game only is solved in this study, and this solution is then used to formulate a strategy for the entire game in accord with the assumption made earlier.

Presentation of material

In this chapter a 2-person version of the phase 1 game will be defined and solved by means of the LAH algorithm (see chapter 3).

The 2-PG is based on the game described in chapter 2 Section 5 (of which the relevant parts, for convenience, are repeated here in shortened form), but has significant detailed differences as follows.

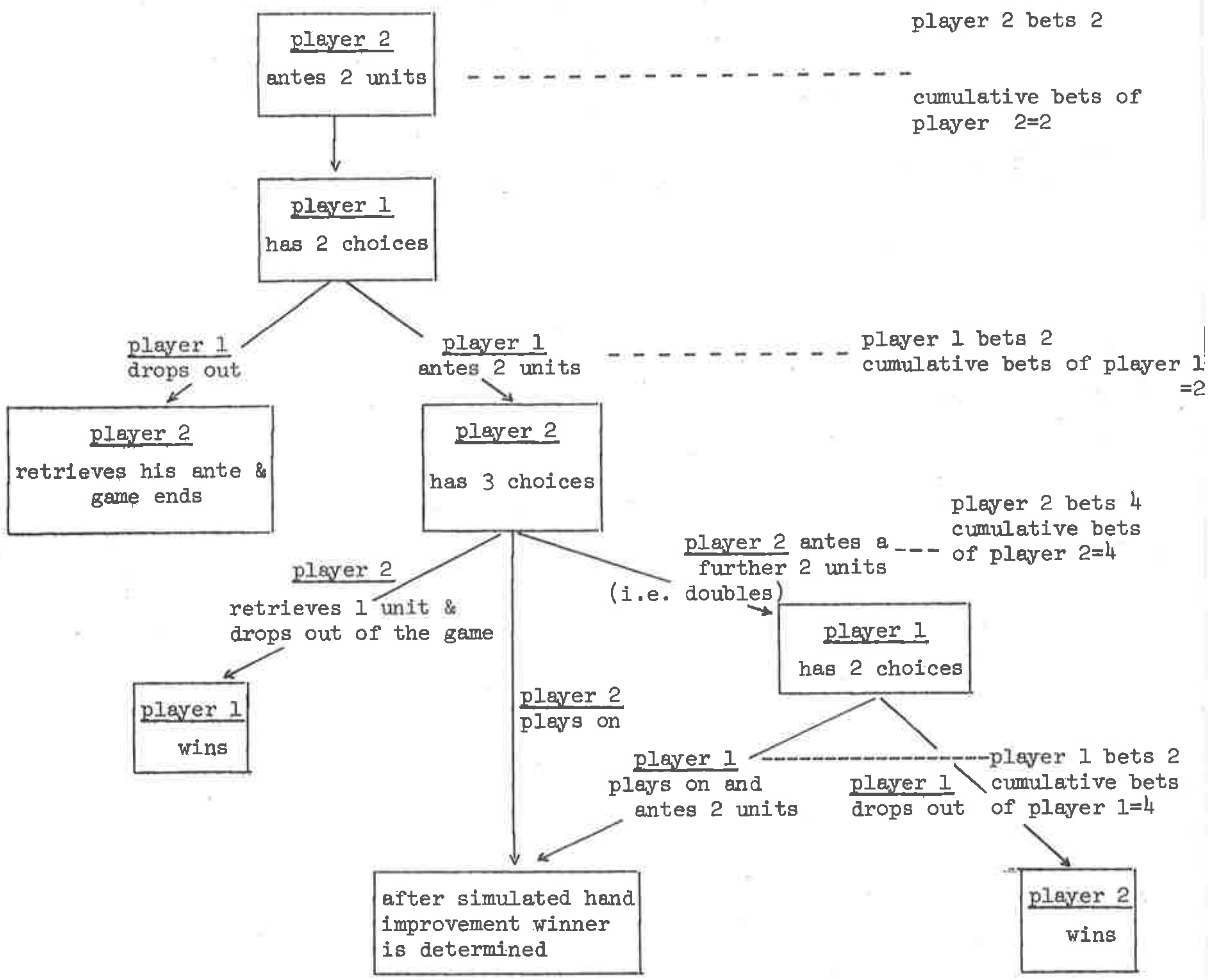
The game is played by two persons called player 1 and player 2. Both players receive a hand of cards, where the hands are respectively represented by random numbers x and y, that are uniformly distributed over the closed interval [0,1] (see 4.5). The play before hand improvement is summarized in the flow diagram given in figure 5. The hand improvement in this game is based on certain simplifying assumptions which are made by expert players in order to analyze the hand improvement process (8), and are given below.

(a) <u>The first assumption is that each player initially holds exactly 1 pair</u>

The justification for this assmption is that players seldom, if ever, play on less than 1 pair,(30). It may be established from table 2 in chapter 2, that if hands weaker than 1 pair are ignored, then 7 out of 8 hands are 1 pair. Thus assumption (a) correctly represents real poker 87% of the time. But since the probability of improving a pair is 0.287 (see table 2, chapter 2), if (a) above is assumed, it follows that the probability of improving any hand is a game constant equal to 0.287. Also, since from (a) above all hands are initially pairs, it follows that if one hand improves while the other does not, then the improved hand must win.

FIGURE 5

FLOW DIAGRAM OF 2-PG



(b) <u>The second assumption is that if 2 hands improve then the higher original hand will win</u>

From (a) above the probability that 2 hands improve is $(0.287)^2 \approx 0.08$. Since, it is true that in this case the higher original hand will win at least half the time, it follows that the probability that assumption (b) is incorrect (i.e. the lower original hand wins), is less than $\frac{1}{2}(0.287)^2 \approx 0.04$ (4%). As a consequence of this it follows that (b) is correct in better than 96% of cases.

The probability of all the above assumptions being correct concurrently (obtained by multiplying conditional probabilities) is $0.87 \times 0.96 \approx 0.84$ (84%). That is the assumptions break down in 16% of instances.

Hence, it would be reasonable to follow a strategy based on these assumptions provided that heavy losses are not sustained in the 16% of instances where at least one of these assumptions was wrong. But it has already been mentioned that in at least 85% of all hands there is no significant betting in phase 2, and in this case no more can be lost on the 16% of occasions when the assumptions do not hold, than in the 84% of instances when they do. Furthermore, the probability of assumptions (a) and (b) being wrong, and there being significant betting in phase 2 is less than $.16 \times 0.15 = 0.024$, and as noted in 4.1, the actual strategy and assumptions used in phase 1 in this instance are unimportant in any case.

As a consequence of this, the experienced player finds it reasonable to use the simplifying assumptions (a) and (b) (see (30)) which are correct in 84% of cases, since he knows that in the 16% of instances where he is wrong, he will not sustain any heavy losses as a result of being incorrect.

Mathematical definition of approximate hand improvement

In accordance with assumptions (a) and (b) made earlier, the mathematical definition of hand improvement is made in the following way.

Define the game constant q, where $q = 0.287$. Then T_q is a random transformation defined as

$$T_q(z) = \begin{cases} z & : \text{with probability } 1-q \\ 2+z & : \text{with probability } q \end{cases} \qquad (1)$$

At the end of the 2-PG, x is replaced by $X = T_q(x)$ and y by $Y = T_q(y)$. Player 1 wins the game if $X \geq Y$ and player 2 if $Y > X$.

4.2 Strategies of players

The strategies of the 2 players are described by the functions $\phi_1(x)$, $\phi_2(x)$, $\psi_1(y)$ and $\psi_2(y)$. The meanings assigned to these functions are displayed in graphical form in figure 6 below.

FIGURE 6

STRATEGIES FOR THE 2-PG

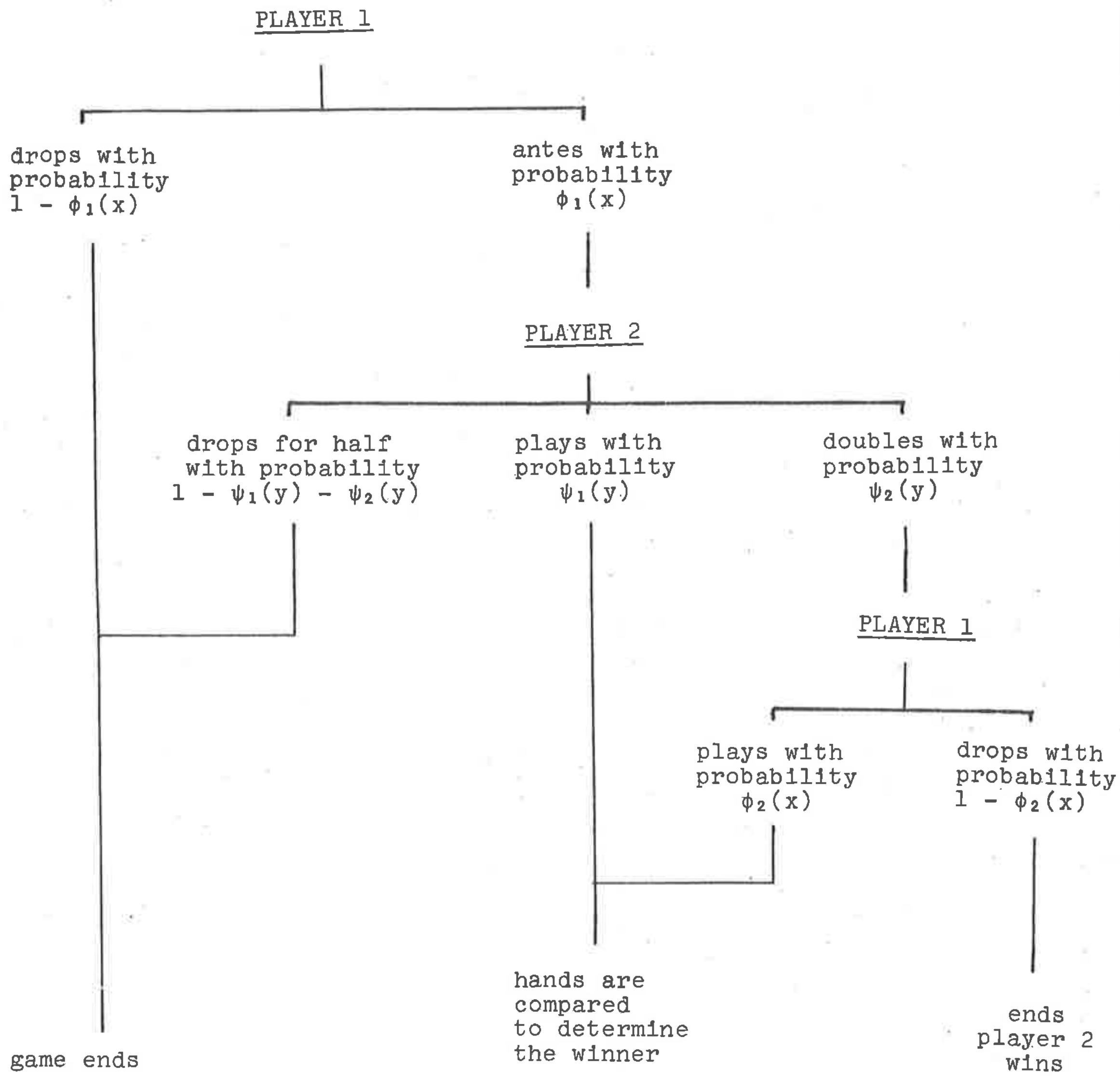


Figure 7 gives the form of the strategy functions and is to be interpreted in the following way. Consider, for example, the function $\phi_1(x)$. This means that player 1 only ever makes an ante if his hand x is less than a (a bluff), or greater than b. He will never play on a hand between a and b.

Observation of experienced players as recorded in (30), and solutions to other poker-like games, (27), suggest that (in addition to the matters noted above) the strategy functions have the following properties:-

(i) Player 1 looks at a double only if his hand x is such that $x \geqslant c$.

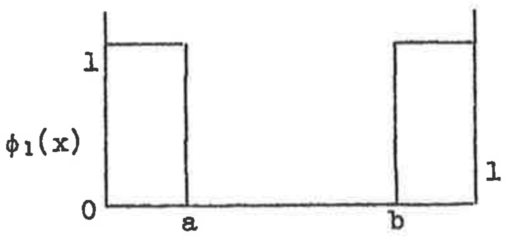
(ii) Player 2 always plays on if his hand is better than d. He doubles on hands better than e, or less than f (the latter bet being a bluff). Otherwise he drops out.

The form of the functions chosen implies that players never follow a variable strategy, i.e. one in which the probabilities of certain actions lie between 0 and 1. This is substantiated in practice, and is explained by Bellman and Blackwell (2), in the following way.

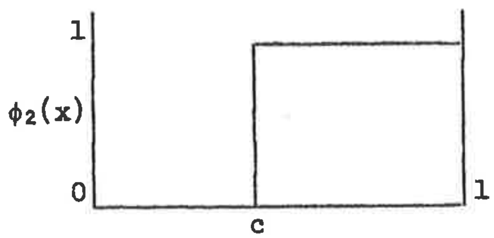
FIGURE 7

2-PG STRATEGY FUNCTIONS

PLAYER I: has a hand x

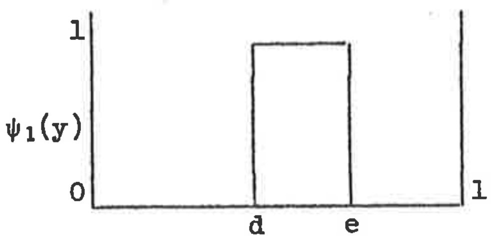


probability of player 1 making an ante for a given hand x, where $0 \le a \le 1$, $0 \le b \le 1$ and $a \le b$.

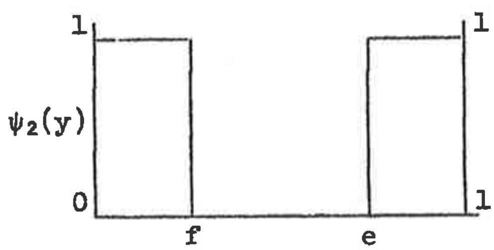


probability of player 1 looking at a double for a given hand x where $0 \le c \le 1$.

PLAYER II: has a hand y



probability of player 2 playing a given hand y, where $0 \le d \le 1$, $0 \le e \le 1$ and $d \le e$.



probability of player 2 doubling for a given hand y, where $0 \le f \le 1$, $0 \le e \le 1$ and $f \le d$.

"In a continuous game, one allows the game, which furnishes a random card, to do the bluffing. It turns out that if the cards are dealt at random, any further randomization furnished by mixed strategies on the part of the players is superfluous."

Thus, this approach uses a very broad general observation, made by experienced players, based on sound practical assumptions, to find a solution in much finer detail. The results finally obtained show that this approach yields satisfactory results.

4.3 Evaluation of payoff functions and solution

The payoff functions for the 2-PG are evaluated in the same way as the payoff functions for von Neumann's game.

First define the function

$\chi_{q_2}^{w,\ell}(x,y)$ = the expected winnings of player 1 holding hand x, competing in the 2-PG against player 2 holding hand y, where player 1 either stands to gain w units if $T_q(x) \geq T_q(y)$ or ℓ units if $T_q(x) < T_q(y)$, (2)

and, as in chapter 3, define

$$\chi_2^{w,\ell}(x,y) = \begin{cases} w \ ; \ x \geq y \\ \ell \ ; \ x < y \end{cases} \tag{3}$$

then from eqs. 1 and 2 it follows that

$$\chi^{w,\ell}_{q2}(x,y) = \begin{Bmatrix}\text{prob. that both players improve}\\ \text{+ prob. that neither player improves}\end{Bmatrix}\begin{bmatrix}\begin{Bmatrix}w;\text{if } x>y\\ \ell;\text{if } x<y\end{Bmatrix}\end{bmatrix}$$

$$+ \begin{Bmatrix}\text{prob. that player 1 improves while}\\ \text{player 2 does not}\end{Bmatrix}.w$$

$$+ \begin{Bmatrix}\text{prob. that player 2 improves while}\\ \text{player 1 does not}\end{Bmatrix}.\ell \qquad (4)$$

Now, since from eq. 1

prob. that a player improves = q (5_

and hence

prob. that a player fails to improve = 1-q (6)

it follows that by using eqs. 3, 5 and 6 , eq. 4 may be written

$$\chi^{w,\ell}_{q2}(x,y) = \{q^2 + (1-q)^2\}\chi^{w,\ell}_{2}(x,y)$$

$$+ q(1-q)w$$

$$+ (1-q)q\ell.$$

Hence, on simplifying

$$\chi^{w,\ell}_{q2}(x,y) = \{q^2+(1-q)^2\}\chi^{w,\ell}_{2}(x,y)+(w+\ell)q(q-q) \qquad (7)$$

Now it is possible to proceed as in chapter 3.

Table 10 below gives the following information.

(a) All possible outcomes which may occur in the game.

(b) The probability of any particular outcome occurring for

TABLE 10

SUMMARY OF PLAY FOR THE 2-PG

Game outcome (the superscripts below refer to the player making a particular move)	Probability of a particular outcome	Payoff to player 1 for a particular outcome
$\text{Play}^1.\text{Drop}^2$	$\phi_1(x).\{1-\psi_1(y)-\psi_2(y)\}$	+1
$\text{Play}^1.\text{Play}^2$	$\phi_1(x)\ \psi_1(y)$	$\chi_{q2}^{2,-2}(x,y)$
$\text{Play}^1.\text{Double}^2.\text{Drop}^1$	$\phi_1(x)\ \psi_2(y).\{1-\phi_2(x)\}$	-2
$\text{Play}^1.\text{Double}^2.\text{Play}^1$	$\phi_1(x)\ \psi_2(y)\ \phi_2(x)$	$\chi_{q2}^{4,-4}(x,y)$

a given set of hands x and y. (These probabilities may be easily found by referring to the rules of the game and the function definitions given in figure 7).

(c) The payoff occurring to player 1 for a particular game outcome (as defined by the rules).

Now, as in chapter 3, P_1, the total expected payoff to player 1, is given by

$$P_1 = \int_{x=0}^{1} \int_{y=0}^{1} [\phi_1(x)\{1-\psi_1(y) - \psi_2(y)\} + \phi_1(x)\psi_1(y)\chi_{q2}^{2,-2}(x,y) - 2\phi_1(x)\psi_2(y)\{1-\phi_2(x)\} + \phi_1(x)\psi_2(y)\phi_2(x)\chi_{q2}^{4,-4}(x,y)]dxdy \tag{8}$$

As this is a zero sum game, P_2, the expected payoff of player 2 is $-P_1$.

Hence

$$P_2 = -P_1. \tag{9}$$

In order to evaluate eq. 8 define

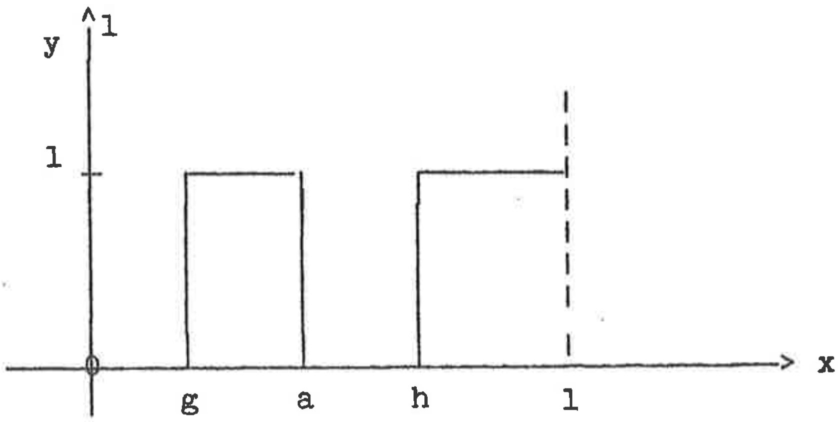
$$\phi_3(x) = \phi_1(x)\phi_2(x) \tag{10}$$

and

$$\phi_4(x) = \phi_1(x)\{1 - \phi_2(x)\} . \tag{11}$$

Consider the function $\phi_3(x)$ given by eq. 10 above. Figure 7 defines the functions $\phi_1(x)$ and $\phi_2(x)$ in terms of variables a,b and c, where $a \leqslant b$.

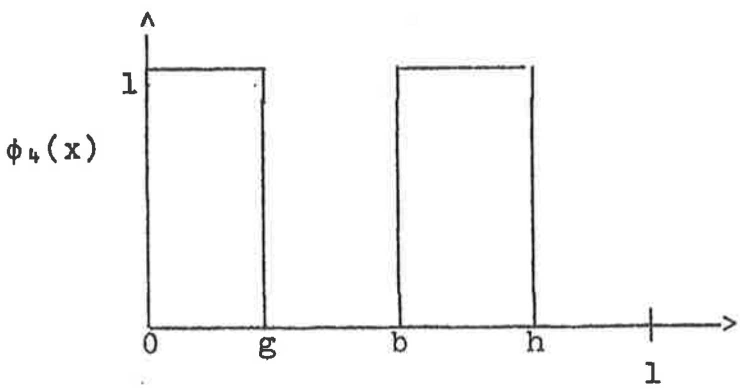
Then it is simple, though tedious, to show that the function $\phi_3(x)$ has the form



where $g = \min(a,c)$ (12)

and $h = \max(b,c)$ (13)

By similar means it can be shown that $\phi_4(x)$ has the form



where g and h are as defined by eqs. 12 and 13. This result may be stated algebraically as

$$\phi_3(x) = \begin{cases} 0 \; ; \; 0 \leq x \leq g & \text{or} \quad a \leq x \leq h \\ 1 \; ; \; g < x \leq a & \text{or} \quad h < x \leq 1 \end{cases}$$

and

$$\phi_4(x) = \begin{cases} 0 \; ; \; g < x \leq b & \text{or} \quad h < x \leq 1 \\ 1 \; ; \; 0 \leq x \leq g & \text{or} \quad b < x \leq h \end{cases}$$

Now eq. 8 may be written

$$P_1 = \int_{x=0}^{1}\int_{y=0}^{1} \phi_1(x)\{1-\psi_1(y)-\psi_2(y)\}dxdy$$

$$+\int_{x=0}^{1}\int_{y=0}^{1}\phi_1(x)\psi_1(y)\chi_{q2}^{2,-2}(x,y)dxdy$$

$$-2\int_{x=0}^{1}\int_{y=0}^{1}\phi_4(x)\psi_2(y)dxdy$$

$$+\int_{x=0}^{1}\int_{y=0}^{1}\phi_3(x)\psi_2(y)\chi_{q2}^{4,-4}(x,y)dxdy$$

and by using the definitions of the functions $\phi_1(x)$, $\phi_2(x)$, $\phi_3(x)$, $\phi_4(x)$, $\psi_1(y)$, $\psi_2(y)$, and expanding the integral, and evaluating some of the component sub-integrals, it can be shown that

$$P_1 = (a + 1 - b)(d - f)$$

$$+\int_{x=0}^{a}\int_{y=d}^{e}\chi^{2,-2}(x,y)dxdy + \int_{x=b}^{1}\int_{y=d}^{e}\chi_{q2}^{2,-2}(x,y)dxdy$$

$$-2(g + h - b)(1 - e + f)$$

$$+\int_{x=g}^{a}\int_{y=e}^{1}\chi^{4,-4}(x,y)dxdy + \int_{x=h}^{1}\int_{y=e}^{1}\chi_{q2}^{4,-4}(x,y)dxdy$$

$$+\int_{x=g}^{a}\int_{y=0}^{f}\chi^{4,-4}(x,y)dxdy + \int_{x=h}^{1}\int_{y=0}^{f}\chi_{q2}^{4,-4}(x,y)dxdy \qquad (16)$$

where $g = \min(a,c)$

and $h = \max(b,c)$.

Now define

$$XQ_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} = \int_{x=a_1}^{b_1} \int_{y=a_2}^{b_2} \chi_{q2}^{w,\ell}(x,y)\,dxdy$$

and using eq. 7

$$XQ_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} = \int_{x=a_1}^{b_1} \int_{y=a_2}^{b_2} (w+\ell)\,q(1-q)\,dxdy$$

$$+ \int_{x=a_1}^{b_1} \int_{y=a_2}^{b_2} \{q^2+(1-q)^2\}\chi_2^{w,\ell}(x,y)\,dxdy \tag{17}$$

and as in chapter 3 define

$$X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} = \int_{x=a_1}^{b_1} \int_{y=a_2}^{b_2} \chi_2^{w,\ell}(x,y)\,dxdy \tag{18}$$

where it has been shown in appendix A that

$$X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} = \ell(u-a_1)\left(\frac{u+a_1}{2} - a_2\right) + w(b_1-u)\left(\frac{b_1+u}{2} - a_2\right)$$

$$- \ell(v-a_1)\left(\frac{v+a_1}{2} - a_2\right) - w(b_1-v)\left(\frac{b_1+v}{2} - a_2\right) \tag{19}$$

with

$$u = \max(a_1, \min(b_1, a_2))$$

and

$$v = \max(a_1, \min(b_1, b_2)).$$

Hence by combining eqs. 17, 18 and 19

$$XQ_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} = q(1-q)(w+\ell)(b_1-a_1)(b_2-a_2) + \{q^2+(1-q)^2\}X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} \qquad (20)$$

Now eq.16 may be written

$$P_1 = (a+1-b)(d-f) - 2(g+h-b)(1-e+f) \qquad (21)$$

$$+ XQ_2^{2,-2}\begin{pmatrix} a & e \\ 0 & d \end{pmatrix} + XQ_2^{2,-2}\begin{pmatrix} 1 & f \\ b & d \end{pmatrix}$$

$$+ XQ_2^{4,-4}\begin{pmatrix} a & 1 \\ g & e \end{pmatrix} + XQ_2^{4,-4}\begin{pmatrix} 1 & 1 \\ h & e \end{pmatrix}$$

$$+ XQ_2^{4,-4}\begin{pmatrix} a & f \\ g & 0 \end{pmatrix} + XQ_2^{4,-4}\begin{pmatrix} 1 & f \\ h & 0 \end{pmatrix}$$

where $g = \min(a,c)$, $h = \max(b,c)$

and $XQ_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix}$ may be found from eq. 20.

Before the problem may be solved by the LAH algorithm, the constraints on the variables a,b,c,d,e and f must be specified. Consider the definitions of $\phi_1(x)$, $\phi_2(x)$, $\psi_1(y)$, $\psi_2(y)$ as given in figure 7. It may be noted from this that the following constraints hold,

$$a \leqslant b$$

$$d \leqslant e$$

and a,b,c,d,e,f all lie in the closed interval [0,1].

(22)

Using the above data it is now possible to solve the 2-PG using the LAH algorithm, for differing values of q. In each case the starting point of the iteration is zero, and the minimum step size is 10^{-6}. A list of the e.p.'s obtained is given in table 11 below.

From the definitions of the strategy functions (see figures 6 and 7) it may be seen that:

(1) c is the minimum hand required by player 1 to play on (look) after a double by player 2;

(2) player 1 never plays on a hand between a and b.

Thus, from (1) and (2) above, it follows that the exact value of c is arbitrary if $a \leq c \leq b$, even though in the solutions given above $c=b$.

4.4 Analytic solution of the game

4.4.1 Initial assumptions

The method of solving the game analytically is a 2 stage process. In the first stage, to be given in this section, certain assumptions are made, and consequently a solution is obtained. The second stage of this process, given in 4.4.2, validates these initial assumptions. Section 4.4.3 will then show that the 2-PG could not be solved analytically if an approximate numerical solution was not already known.

In the previous section it has been shown that the solution to this game, when $q = 0.287$ is:

TABLE 11

SOLUTION TO THE 2-PG

q	a	b	c	d	e	f	payoff to player 1
0.0	0.0889	0.7333	0.7333	0.6667	0.8667	0.0000	0.1778
0.1	0.0719	0.7035	0.7035	0.6425	0.8438	0.0000	0.1842
0.2	0.0502	0.6708	0.6708	0.6241	0.8236	0.0000	0.1897
0.287	0.0298	0.6418	0.6418	0.6127	0.8106	0.0000	0.1940
0.4	0.0076	0.6107	0.6107	0.6000	0.8000	0.0000	0.2000

$a = .0298, \quad b = .6418, \quad c = .6418, \quad d = .6127, \quad e = .8106,$
$f = 0.0000.$ (23)

Suppose that, motivated by the above approximate numerical solution, the following inequality relationships are assumed to hold between the variables at the exact e.p.

$$a \leqslant d, \quad a \leqslant e, \quad d \leqslant b \leqslant e \tag{24}$$

The correctness of this assumption will be shown in 4.4.2.

Now from eq. 24 it may be established that

$$\begin{aligned} a &= \max(0,\min(a,d)) \\ b &= \max(b,\min(1,d)) \\ a &= \max(a,\min(a,e)) \\ e &= \max(a,\min(1,e)) \\ e &= \max(b,\min(1,e)) \\ a &= \max(a,\min(a,0)) \\ b &= \max(b,\min(1,0)) \end{aligned} \tag{25}$$

and by using eqs. 19 ,20 ,21 and 25 , after some lengthy algebra it may be shown that P_1 reduces to

$$\begin{aligned} P_1 &= (d-f)(1+a-b)-2a(1+f-e) \\ &+2Q\{(e-d)(1-e-a)+(e-b)(b-d+2e-2)+2f(1-b)\} \end{aligned} \tag{26}$$

where $$Q = q^2 + (1-q)^2 \tag{27}$$

and thus from eq. 9

$$\begin{aligned} P_2 &= -(d-f)(1+a-b)+2a(1+f-e) \\ &-2Q\{(e-d)(1-e-a)+(e-b)(b-d+2e-2)+2f(1-b)\} \end{aligned} \tag{28}$$

It has been explained earlier in this chapter that if $a \leqslant c \leqslant b$ then the exact value of c is irrelevant to the payoff obtained. This is confirmed by the fact that c does not appear in eqs. 26 and 28.

It is now possible to solve this game analytically in the following way. First it is assumed that

$$f = 0 \qquad (29).$$

The correctness and consistency of this assumption will be shown in 4.4.2.

Thus, putting $f=0$ into eqs. 26 and 28, gives

$$P_1 = d(1+a-b) - 2a(1-e) + 2Q\{(e-d)(1-e-a)+(e-b)(b-d+2e-2)\} \qquad (30)$$

$$P_2 = -d(1+a-b)+2a(1-e) - 2Q\{(e-d)(1-e-a)+(e-b)(b-d+2e-2)\} \qquad (31)$$

Evaluate the following partial derivatives using eqs. 30 and 31

$$\frac{\partial P_1}{\partial a} = d-2(1-e)-2Q(e-d)$$

$$\frac{\partial P_1}{\partial b} = -d+2Q(-2b+d-e+2)$$

$$\frac{\partial P_2}{\partial d} = -1-a+b-2Q(a+b-1)$$

$$\frac{\partial P_2}{\partial e} = -2a-2Q(2e-a-b-1)$$

and solving the equations

$$\frac{\partial P_1}{\partial a} = 0 \quad , \quad \frac{\partial P_1}{\partial b} = 0$$

$$\frac{\partial P_2}{\partial d} = 0 \quad , \quad \frac{\partial P_2}{\partial e} = 0$$

using Gaussian elimination (assuming that q is such that division by zero does not occur) it is found that

$$a = \frac{c_5}{c_4} \quad , \quad b = \frac{c_1 \; c_7 - c_2}{4Q}$$

$$d = \frac{2 - c_7(2-2Q)}{1+2Q} \quad , \quad e = c_7 \; .$$

where

$$Q = q^2 + (1-q)^2$$

$$c_1 = \frac{2(1-4Q)}{1+2Q}$$

$$c_2 = -4Q - \frac{2(2Q-1)}{2Q+1}$$

$$c_3 = 4Q - \frac{c_1}{2}$$

$$c_4 = 2-2Q - \frac{c_3(4Q-1)}{2Q(2Q-1)}$$

$$c_5 = \frac{c_1 - c_2}{2} - 2Q$$

$$c_6 = 4Q-2 - \frac{c_5}{c_4}\left\{\frac{4Q-1}{Q}\right\}$$

$$c_7 = \frac{c_6}{2(2Q-1)}$$

and for $q = .287$ it can be shown that

$$a = 0.0298 \qquad b = 0.6418$$
$$d = 0.6127 \qquad e = 0.8106. \tag{32}$$

Since, as explained earlier, the exact value of c is immaterial if $a \leqslant c \leqslant b$,

let $$c = b = 0.6418.$$

Also, by eq.29, f=0, thus the complete solution is

a = 0.0298	b = 0.6418	c = 0.6418
d = 0.6127	e = 0.8106	f = 0.0000

and this is the same as the numerical solution given in table 11 for $q = 0.287$. The numerical solutions given in table 11 for other values of q may be checked in the same way.

4.4.2 Showing that the solution found is an e.p.

In this section the assumptions made in eqs.24 and 29 will be validated, and then the solution will be shown to satisfy Nash's criterion.

First, eq.24 follows from eq.32.

It will now be shown that eq.29 (f=0) satisfies the Nash condition and is therefore correct.

First differentiating eq.28 partially with respect to f,

$$\left.\frac{\partial P_2}{\partial f}\right|_{\substack{\text{e.p of} \\ \text{eq.32}}} = -0.41 < 0 \tag{33}$$

Eq.33 shows that the assumption that f=0 is correct, for the following reason. From eq.33 player 2 can only increase his payoff P_2 by making f smaller (if the other variables remain constant) as $\frac{\partial P_2}{\partial f} < 0$. But since $f=0$ and f is constrained to $0 \leqslant f \leqslant 1$, player 2 is forced to leave f at 0, and hence eq.29 is justified. The remaining derivatives satisfy the Nash conditions, since they are zero.

Thus the e.p. found analytically satisfies the conditions for a Nash e.p.

4.4.3 The dependence of the analytic solution on the approximate solution

The above method of analytic solution is based upon certain assumptions (later proved correct) made in eqs.24 and 29. However, it can be shown that there can be approximately 2^8 different possible initial assumptions. As the algebra required to solve each individual case is lengthy it would be clearly impractical to attempt to solve this problem analytically unless the number of such possible cases was first reduced to manageable proportions. One way of achieving this is to have an approximate numerical solution initially.

All the solutions given in table 11 have been checked analytically in the manner described in this section.

4.5 Validation of hand improvement simulation

In this section it will be shown that the approximate simulation of hand improvement in the 2-PG is, for the purposes of this study, indistinguishable from real hand improvement.

Define the 2-PG* to be a game, identical to the 2-PG in every respect, except that exact hand improvement (as will be defined below), rather than approximate hand improvement, will be used, and thus the 2-PG* is equivalent to phase 1 of real poker. This section will show that strategies optimal for the 2-PG are also optimal for the 2-PG*.

The function $f_b(h)$ was defined in chapter 2, and determines the winning probability, $x = f_b(h)$, associated with any poker hand h. Define $f_b^{-1}(x)$ to be the inverse of the function $f_b(h)$, where $h = f_b^{-1}(x)$ determines the poker hand h associated with any given winning probability x. Consequently, $h = f_b^{-1}(x)$ relates any given x, $0 \leqslant x \leqslant 1$, to some poker hand h.

Suppose a player holds hand x in the 2-PG*. Then hand improvement in the 2-PG* is defined in the following way. First calculate $h_x = f_b^{-1}(x)$. Next improve h_x by discarding the appropriate number of cards (see chapter 2) and replacing them with new cards (taking into account cards already held) to give an improved hand h_x'.

Now it will be demonstrated that the payoff functions for the 2-PG* are indistinguishable from the corresponding payoff functions for the 2-PG.

Proceeding as in the 2-PG, define

$\chi_{q2}^{w,\ell^*}(x,y)$ = the expected winnings of player 1 holding hand x, competing against player 2 holding hand y in 2-PG* where both players improve their hands in the manner described above, and player 1 stands to either gain w units or ℓ units depending on whether he holds the winning hand or not, (34)

and

$$XQ_2^{w,\ell^*}\begin{pmatrix} b & d \\ a & c \end{pmatrix} = \int_{x=a}^{b} \int_{y=c}^{d} \chi_{q2}^{w,\ell^*}(x,y)\,dxdy \qquad (35)$$

It follows from eq.34 and 35 that a monte carlo method may be employed to define XQ_2^{w,ℓ^*} in the following way.

$XQ_2^{w,\ell^*}\begin{pmatrix} b & d \\ a & c \end{pmatrix}$ = the expected winnings of player 1 over a period of many games of the 2-PG* where he stands to either gain w units, or ℓ units during any particular game (depending on whether he wins or loses the hand), and where x, the hand of player 1, is such that $a \leqslant x \leqslant b$, and y, the hand of player 2 is such that $c \leqslant y \leqslant d$. (36)

Now by using eq.36 it may be seen that it is possible to estimate $XQ_2^{w,\ell *}$ by simulating the results of a large number of games played under these particular conditions.

Thus $XQ_2^{w,\ell *}$ may be calculated to an arbitrary degree of accuracy, which is determined by the number of games simulated (max). A Fortran program was written to do this, and the results are given in table 12 below.

Max was limited to approximately 10,000 for reasons of economy.

It may be seen from the above results that:

(a) Lines (13) - (15) in table 12 show that when the exact value of $XQ_2^{w,\ell *}$ is known to be zero, the approximate value of $XQ_2^{w,\ell *}$ approaches the correct value for increasing values of max.

(b) The expected order of inaccuracy which appears to be associated with the $XQ_2^{w,\ell *}$ approximation for max = 10,000 is approximately 0.005 (see table 12, lines 2 and 6). Furthermore it may be noted that the maximum differences associated with $XQ_2^{w,\ell}$ and the $XQ_2^{w,\ell *}$ approximation are of the same order.

If it is hypothesised that the function $XQ_2^{w,\ell *}$ is a normally distributed approximation to $XQ_2^{w,\ell}$, which becomes more accurate as max is increased, then a standard statistical test (t-test) shows that the data supports this hypothesis to better than the 5% confidence level.†

---

†For the values considered t = -0.395 where $t_6(5\%) = 2.45$.

TABLE 12

SIMULATION OF 2-PG PAYOFF FUNCTIONS

	w	ℓ	a	b	c	d	max. no.of games used in simulation	$XQ_2^{w,\ell}$	approx. $XQ_2^{w,\ell*}$	exact $XQ_2^{w,\ell*}$ (if known)
(1)	4	-2	0.1	1.0	0.1	1.0	10,000	0.810	0.815	-
(2)	2	-2	0.8	1.0	0.8	1.0	10,000	0.000	0.004	0.000
(3)	2	-2	0.0	0.03	0.5	0.9	10,000	-0.014	-0.016	-
(4)	2	-2	0.5	1.0	0.5	0.8	10,000	0.071	0.069	-
(5)	2	-2	0.8	1.0	0.5	0.8	10,000	0.071	0.068	-
(6)	2	-2	0.5	1.0	0.5	1.0	10,000	0.000	0.006	0.000
(7)	5	-2	0.1	0.9	0.2	0.7	10,000	0.703	0.702	-
(8)	5	-2	0.1	0.6	0.3	0.8	10,000	0.044	0.039	-
(9)	3	-1	0.2	0.4	0.3	0.6	10,000	0.001	0.002	-
(10)	2	-1	0.3	0.7	0.5	0.9	10,000	-0.026	-0.023	-
(11)	1	-1	0.6	0.9	0.1	0.7	10,000	0.100	0.100	-
(12)	1	-1	0.8	1.0	0.8	1.0	10,000	0.002	0.001	-
(13)	2	-2	0.5	1.0	0.5	1.0	10,000	0.000	0.006	0.000
(14)	2	-2	0.5	1.0	0.5	1.0	40,000	0.000	0.003	0.000
(15)	2	-2	0.5	1.0	0.5	1.0	80,000	0.000	0.002	0.000

Hence from the above and eq.21 it follows that the payoff functions calculated for the 2-PG and 2-PG* can be for the purposes of this study, considered identical. Thus the optimal strategy for the 2-PG will also be optimal for the 2-PG*.

4.6 Non-optimal solutions

Situations often arise in real games where one of the players is known to be playing non-optimally. It is then possible to use the LAH algorithm to find strategies for the other player, which will yield him an even better return. Consider the example of a 2-PG with q=0.287, with the solution (see table 11):-

a	b	c	d	e	f	payoff to I	payoff to II
0.03	0.642	0.642	0.613	0.811	0.000	0.1940	-0.1940

Note that player II has a payoff of -0.1940 which means that, on the average, he expects to lose 0.1940 units per game.

Suppose that player 1 now plays b=0.500 (instead of b = 0.642). It is then possible to solve the game as before, except now the variable b is treated as a constant, equal to 0.5. The new solution is -:

a	b	c	d	e	f	payoff to I	payoff to II
0.020	0.500	0.500	0.538	0.750	0.000	0.174	-0.174

Note the changes made in the variables d and e by player 2 in order to take full advantage of player 1's bad play and thereby increase his own expected payoff from -0.194 to -0.174, that is, instead of losing 0.194 he now loses 0.174, which is an improvement of 0.02.

4.7 Discussion of results

A general discussion of this analytic solution of poker is given in chapter 5 in the context of the 4-person version of this game (4-PG). However, particular aspects of the 2-PG which lead to simplifying assumptions used in solving the 4-PG, are discussed below.

(a) Bluffing by player 1

The 2-PG, with $q = 0.287$, has the solution (given in table 11):-

$a = 0.03$, $b = 0.64$, $c = 0.64$, $d = 0.61$, $e = 0.82$, $f = 0.00$

Figures 6 and 7 show that player 1 will play with a hand x if $0 \leq x \leq a$ or $b \leq x \leq 1$. If x is such that $0 \leq x \leq a$, and since $a = 0.03$, this means that x is a weak hand. Hence, to bet with such a hand is to bluff, and the value of a shows that this bluff should only be tried, on the average, 1 hand in every 33.

This agrees with the opinions expressed in poker books that this bluff should be used extremely sparingly, if at all.(30).

In the complete poker game bluffing in phase 1 becomes even less important, since bluffing can be done far more efficiently in phase 2, when more information has become available and unlimited betting is possible.

(b) Playing on after a double by the last player

The solution to the 2-PG (see (a) above) shows b = c = 0.64. Figure 7 may be used to interpret these values, and it is found that player 1 will always play on after a double unless his original bet was a bluff.

This may be intuitively justified because player 1 only ever plays with a good hand (assuming he rarely, if ever, bluffs, as discussed in (a) above). Thus it is always worth playing on after a double as it only costs a further 2 units from player 1 to have a reasonable chance of winning what will now have become a large pot. This same strategy is recommended in poker books, (32).

(c) Bluffing by the last player (player 2)

The solution to the 2-PG (see (a) above) shows that e=0.0. From figure 7 and the fact that e=0.00 it follows that player 2 never bluffs by doubling with a weak hand.

It will become apparent in chapter 5 that this game becomes too complicated to solve (by the methods used here) if the rules are extended to allow 3 and 4 players. In order to obtain a solution for the 4-person case the rules must be simplified.

It has been shown here that in the 2-PG, players very rarely bluff (see (a) and (c) above), and a player will always continue after a double (see (b) above). In addition poker books suggest that these same precepts are used by experienced poker players (see discussion above). Hence, the following simplifying assumptions were incorporated in the rules of the 3-PG and 4-PG.

First, players do not bluff, and second, all players continue after a double.

CHAPTER 5

SOLUTION OF 3 AND 4 PERSON GAMES (3-PG AND 4-PG)

5.1 Introduction

In this chapter a simplified 3 and 4 person version (3-PG and 4-PG) of the game treated in chapter 4, will be solved. There are, however, several points which should first be noted.

(1) The need for simplification of the rules (see 4.7) will become apparent when the large amount of central processor (c.p.) time required to solve the 4-PG is noted (see 5.3.3). Without this simplification the amount of c.p. time would be many times greater, and as a consequence, it would not be practicable to solve this problem on available computers.

(2) Since much of this work is a straightforward extension of work done in chapter 4, unnecessary details will be omitted. Also the arguments used to justify approximate hand improvement and the applicability of the phase-1 solution to the entire game will not be repeated as they remain unchanged from chapter 4.

5.2 Solution of the 3-PG

5.2.1 Rules of play

The rules of the 3-PG and 4-PG are the same as for the 2-PG with the following 2 simplifications, which have been discussed in 4.7, and will be stated here without further explanation.

(a) The form of the strategy functions is so chosen that direct bluffing is not allowed. Note, however, that a subtle form of bluffing is still possible (see 5.5.2).

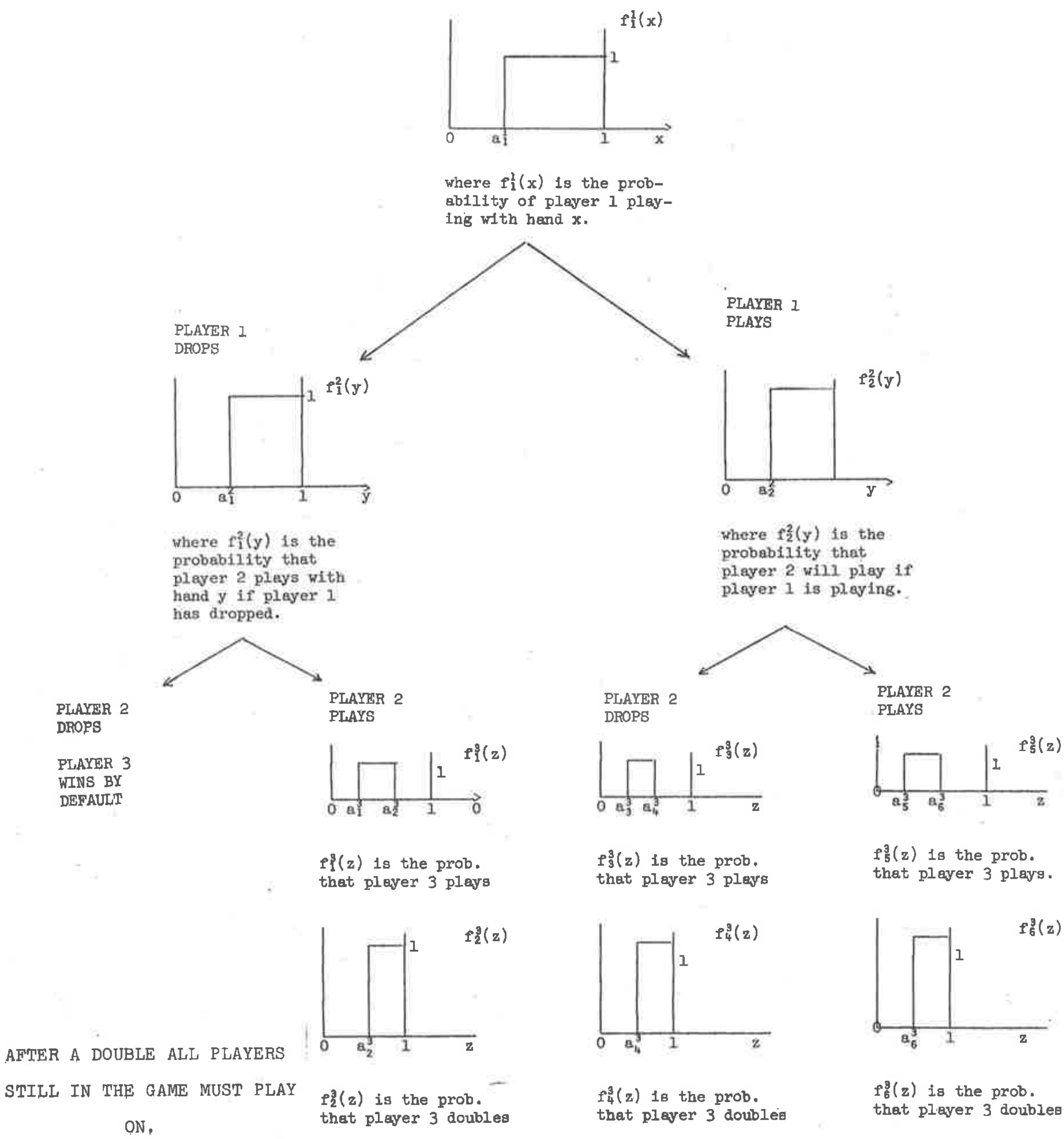
(b) When the last player doubles all other players must look.

5.2.2 Strategies

In the 3-PG players 1,2 and 3 receive hands x,y and z respectively, and then must make decisions, according to the rules of the game, on the basis of their hands and the actions taken by other players. This is most conveniently represented by figure 8 below, which gives a flow diagram for the 3-PG. It is assumed that the 3 players are dealt hands x,y and z. The diagram lists all possible game outcomes and specifies the probability function which each player will use to decide which action to take at any particular stage of the game. Each probability function $f_j^i(x)$ is defined in terms of variables a_j^i as indicated in figure 8.

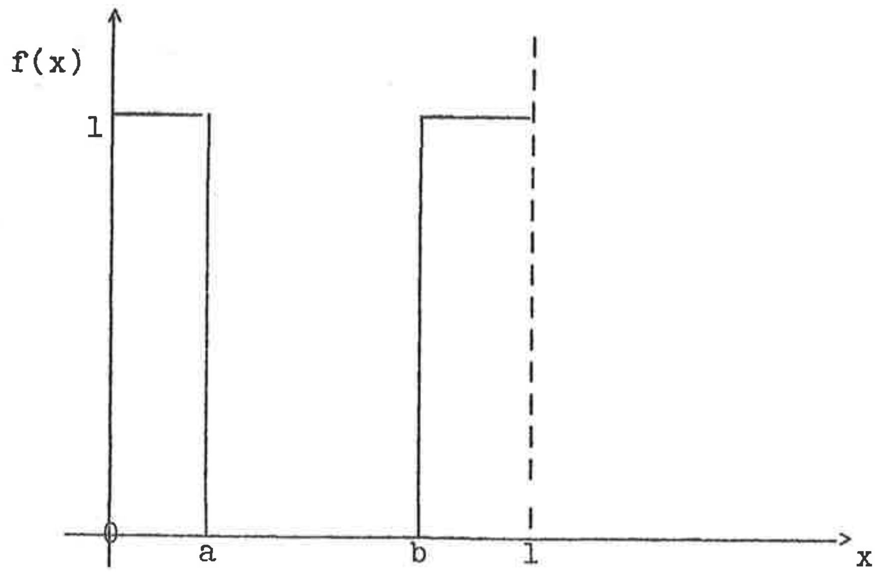
FIGURE 8

Summary of strategy function definitions and game flow diagram for 3-PG



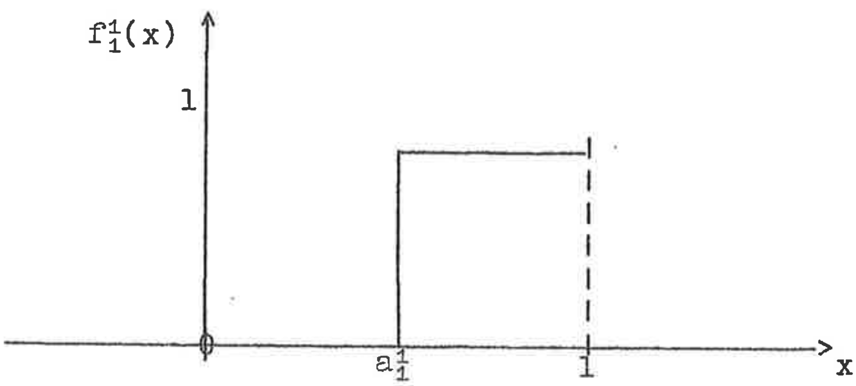
The following points relating to the form of the probability functions should be noted.

The strategy function for the first player in the 2-PG was assumed to be of the form $f(x)$, as described in the figure below.



(1)

As has been noted in 4.7 a non-zero value of a indicates bluffing. As bluffing has been excluded from the 3-PG, a is set to zero, and thus the strategy function for the first player in the 3-PG, $f_1^1(x)$, takes the form shown in the figure below.



(2)

5.2.3 Evaluation of payoff functions

Before the payoff function is evaluated (in the same manner as for the 2-PG) it is necessary to define the following function.

$\chi_{q3}^{w,\ell}(x,y,z)$ = the expected winnings of player 1, holding hand x, competing against players 2 and 3 holding hands y and z respectively, in the 3-PG, where player 1 wins w if he beats his opponents, or ℓ if he loses. (3)

Now, proceeding as before, it is possible to express this function in terms of the functions

$\chi_2^{w,\ell}(x,y)$ and $\chi_3^{w,\ell}(x,y,z)$ where

$$\chi_2^{w,\ell}(x,y) = \begin{cases} w \; ; \; x \geq y \\ \ell \; ; \; x < y \end{cases} \tag{4}$$

and

$$\chi_3^{w,\ell}(x,y,z) = \begin{cases} w \; ; \text{ if } \; x \geq y \; \text{ and } \; x \geq z \\ \ell \; ; \text{ if } \; x < y \; \text{ or } \; x < z \end{cases} \tag{5}$$

By combining eqs. 3,4 and 5 it may be seen that

$$\chi_{q3}^{w,\ell}(x,y,z) = \left\{\begin{matrix}\text{probability that hand } x \\ \text{improves while hands } y \text{ and } z \text{ do not}\end{matrix}\right\}\cdot\{w\} \tag{6}$$

$$+\left\{\begin{matrix}\text{probability that at least 1 of hands} \\ y \text{ or } z \text{ improve while hand } x \text{ does not}\end{matrix}\right\}\cdot\{\ell\}$$

$$+\left\{\begin{matrix}\text{probability that hands } x \text{ and } y \text{ im-} \\ \text{prove while hand } z \text{ does not.}\end{matrix}\right\}\cdot\left\{\chi_2^{w,\ell}(x,y)\right\}$$

$$+\left\{\begin{array}{l}\text{probability that hands x and z}\\ \text{improve while hand y does not.}\end{array}\right\}\cdot \chi_2^{w,\ell}(x,z)$$

$$+\left\{\begin{array}{l}\text{probability that either all hands}\\ \text{improve, or all fail to improve}\end{array}\right\}\cdot \chi_3^{w,\ell}(x,y,z)$$

Now from the definitions given by eq.1, chapter 4, it follows that:-

probability of improving hand = q (7)

probability of failing to improve a hand = 1-q (8)

and combining eqs. 6, 7 and 8 it may be seen that

$$\chi_{q3}^{w,\ell}(x,y,z) = w\{q(1-q)^2\} + \ell\{(1-q)[1-(1-q)^2]\}$$
$$+ \chi_2^{w,\ell}(x,y)\ \{q^2(1-q)\} + \chi_2^{w,\ell}(x,z)\ \{q^2(1-q)\}$$
$$+ \chi_3^{w,\ell}(x,y,z)\ \{q^3+(1-q)^3\}$$

which simplifies to

$$\chi_{q3}^{w,\ell}(x,y,z) = wq(1-q)^2 + \ell(1-q)[1-(1-q)^2] \tag{9}$$
$$+ \{q^2(1-q)\}\ \{\chi_2^{w,\ell}(x,y) + \chi_2^{w,\ell}(x,z)\}$$
$$+ \{q^3 + (1-q)^3\}.\chi_3^{w,\ell}(x,y,z).$$

Since from this point, the triple integrals of $\chi_{q3}^{w,\ell}(x,y,z)$ and $\chi_3^{w,\ell}(x,y,z)$ arise repeatedly in the text, it is convenient to define, (as in the 2-PG),

$$XQ_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix} = \int_{x=a_1}^{b_1}\int_{y=a_2}^{b_2}\int_{z=a_3}^{b_3} \chi_{q3}^{w,\ell}(x,y,z)dxdydz \tag{10}$$

and

$$X_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix} = \int_{x=a_1}^{b_1} \int_{y=a_2}^{b_2} \int_{z=a_3}^{b_3} \chi_3^{w,\ell}(x,y,z)\,dxdydz$$

then by substituting eq.9 into eq.10, and using the above equation, it can be shown that

$$XQ_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix} = \left\{ \prod_{i=1}^{3} (b_i - a_i) \right\} . \left\{ wq(1-q)^2 + \ell(1-q).[1-(1-q)^2] \right\}$$

$$+ \; q^2(1-q). \left\{ (b_3-a_3) X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix} + (b_2-a_2) X_2^{w,\ell}\begin{pmatrix} b_1 & b_3 \\ a_1 & a_3 \end{pmatrix} \right\}$$

$$+ \left\{ q^3 + (1-q)^3 \right\} . X_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix} \qquad (11)$$

where the functions $X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix}$ and $X_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix}$

have been evaluated and are given in appendix A.

From this point the calculation follows the pattern established in chapter 4. First, all possible plays, their probabilities of occurring, and corresponding payoffs, are calculated, and these are given in table 13.

Next, proceeding as in the 2-PG, and making use of table 13 and eq.11, the payoff functions for each player may be evaluated. Since the method of calculation is the same as for the 2-PG, only the final result will be given here.

TABLE 13

SUMMARY OF PLAY FOR THE 3-PG

	Play	Probability	Payoffs I	Payoffs II	Payoffs -III
1.	$D^1P^2D^3$	$[1-f_1^1(x)]f_1^2(y)[1-f_1^3(z)-f_2^3(z)]$	0	1	-1
2.	$D^1P^2P^3$	$[1-f_1^1(x)]f_1^2(y)f_1^3(z)$	0	$\chi_{q2}^{2,-2}(y,z)$	$\chi_{q2}^{2,-2}(z,y)$
3.	$D^1P^2D_b^3$	$[1-f_1^1(x)]f_1^2(y)f_2^3(z)$	0	$\chi_{q2}^{4,-4}(y,z)$	$\chi_{q2}^{4,-4}(z,y)$
4.	$P^1D^2D^3$	$f_1^1(x)[1-f_2^2(y)][1-f_3^3(z)-f_4^3(z)]$	+1	0	-1
5.	$P^1D^2P^3$	$f_1^1(x)[1-f_2^2(y)]f_3^3(z)$	$\chi_{q2}^{2,-2}(x,z)$	0	$\chi_{q2}^{2,-2}(z,x)$
6.	$P^1D^2D_b^3$	$f_1^1(x)[1-f_2^2(y)]f_4^3(z)$	$\chi_{q2}^{4,-4}(x,z)$	0	$\chi_{q2}^{4,-4}(z,x)$
7.	$P^1P^2D^3$	$f_1^1(x)f_2^2(y)[1-f_5^3(z)-f_6^3(z)]$	$\chi_{q2}^{3,-2}(x,y)$	$\chi_{q2}^{3,-2}(y,x)$	-1
8.	$P^1P^2P^3$	$f_1^1(x)f_2^2(y)f_5^3(z)$	$\chi_{q3}^{4,-2}(x,y,z)$	$\chi_{q3}^{4,-2}(y,x,z)$	$\chi_{q3}^{4,-2}(z,x,y)$
9.	$P^1P^2D_b^3$	$f_1^1(x)f_2^2(y)f_6^3(z)$	$\chi_{q3}^{8,-4}(x,y,z)$	$\chi_{q3}^{8,-4}(y,x,z)$	$\chi_{q3}^{8,-4}(z,x,y)$

Define the joint strategy vector

$$\underset{\sim}{a} = [a_1^1, a_1^2, a_2^2, a_1^3, a_2^3, a_3^3, a_4^3, a_5^3, a_6^3]^\dagger \tag{12}$$

and let $P_3^i(\underset{\sim}{a})$ be the corresponding payoff to player i, for $i=1,2$ and 3.

Then, it may be shown that:

$$P_3^1(\underset{\sim}{a}) = a_2^2 a_3^3 (1-a_1^1) + a_2^2 XQ_2^{2,-2}\begin{pmatrix}1 & a_4^3\\ a_1^1 & a_3^3\end{pmatrix}$$

$$+ a_2^2 XQ_2^{2,-2}\begin{pmatrix}1 & 1\\ a_1^1 & a_4^3\end{pmatrix} + a_5^3 XQ_2^{3,-2}\begin{pmatrix}1 & 1\\ a_1^1 & a_2^2\end{pmatrix}$$

$$+ XQ_3^{4,-2}\begin{pmatrix}1 & 1 & a_6^3\\ a_1^1 & a_2^2 & a_5^3\end{pmatrix} + XQ_3^{8,-4}\begin{pmatrix}1 & 1 & 1\\ a_1^1 & a_2^2 & a_6^3\end{pmatrix}$$

$$P_3^2(\underset{\sim}{a}) = a_1^1(1-a_1^2)a_1^3 + a_1^1 XQ_2^{2,-2}\begin{pmatrix}1 & a_2^3\\ a_1^2 & a_1^3\end{pmatrix} + a_1^1 XQ_2^{4,-4}\begin{pmatrix}1 & 1\\ a_1^2 & a_2^3\end{pmatrix}$$

$$+ a_5^3 XQ_2^{3,-2}\begin{pmatrix}1 & 1\\ a_2^2 & a_1^1\end{pmatrix} + XQ_3^{4,-2}\begin{pmatrix}1 & 1 & a_6^3\\ a_2^2 & a_1^1 & a_5^3\end{pmatrix} + XQ_3^{8,-4}\begin{pmatrix}1 & 1 & 1\\ a_2^2 & a_1^1 & a_6^3\end{pmatrix}$$

$$P_3^3(\underset{\sim}{a}) = a_1^1(1-a_1^2)a_1^3 + a_1^1 XQ_2^{2,-2}\begin{pmatrix}a_2^3 & 1\\ a_1^3 & a_1^2\end{pmatrix}$$

$$+ a_1^1 XQ_2^{4,-4}\begin{pmatrix}1 & 1\\ a_2^3 & a_1^2\end{pmatrix} - (1-a_1^1)a_2^2 a_3^3 + a_2^2 XQ_2^{2,-2}\begin{pmatrix}a_4^3 & 1\\ a_3^3 & a_1^1\end{pmatrix}$$

$$+ a_2^2 XQ_2^{4,-4}\begin{pmatrix}1 & 1\\ a_4^3 & a_1^1\end{pmatrix} - (1-a_1^1)(1-a_2^2)a_5^3$$

$$+ XQ_3^{4-2}\begin{pmatrix}a_6^3 & 1 & 1\\ a_5^3 & a_1^1 & a_2^2\end{pmatrix} + XQ_3^{8,-4}\begin{pmatrix}1 & 1 & 1\\ a_6^3 & a_1^1 & a_2^2\end{pmatrix}$$

Constraints

By virtue of the function definitions (see figure 8) it follows that:

$$0 \leqslant a^i_j \leqslant 1 \quad \text{for all} \quad a^i_j$$

and

$$a^3_1 \leqslant a^3_2, \quad a^3_3 \leqslant a^3_4, \quad a^3_5 \leqslant a^3_6$$

Knowledge of these data permits solution of the 3-PG by the LAH algorithm.

5.2.4 Solution of the 3-PG

The solution of the 3-PG, for $q = 0.287$, is presented below.

Player	Expected Payoff	Variables		
1	0.1386	$a^1_1 = 0.715$		
2	0.1839	$a^2_1 = 0.615$	$a^2_2 = 0.800$	
3	-0.3225	$a^3_1 = 0.620$	$a^3_2 = 0.815$	$a^3_3 = 0.730$
		$a^3_4 = 0.830$	$a^3_5 = 0.825$	$a^3_6 = 0.885$

This solution will be discussed in 5.5.4. The LAH algorithm took approximately 44 seconds of Central Processor (C.P.) time to compute this solution.

5.3 Solution of the 4-PG

5.3.1 Evaluation of payoff functions

The rules of the 4-PG, apart from the addition of 1 extra player, are exactly the same as for the 3-PG.

Also the method of evaluating the payoff functions remains unchanged, except that the following preliminary result is first required.

Before evaluating the payoff functions for the 2-PG, the function $XQ_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix}$ was defined and evaluated. Similarly the function $XQ_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix}$ was required before the 3-PG payoff functions could be calculated. Now, following this pattern, the function

$$XQ_4^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix}$$

will be defined and evaluated as a prerequisite to calculating the 4-PG payoff functions.

First, define

$\chi_{q4}^{w,\ell}(x,y,z,w)$ = the expected winnings of player 1 in the 4-PG holding hand x, competing against players 2,3 and 4 who are holding hands y,z and w respectively, where all players are given an opportunity of improving their hand with probability q. (13)

Now, as for the 3-PG, $\chi_{q4}^{w,\ell}(x,y,z,w)$ may be expressed in terms of the functions

$$\chi_2^{w,\ell}(x,y) = \begin{cases} w & ; \; x \geq y \\ \ell & ; \; x < y \end{cases} \tag{14}$$

$$\chi_3^{w,\ell}(x,y,z) = \begin{cases} w & ; \; x \geq y \text{ and } x \geq z \\ \ell & ; \; x < y \text{ or } x < z \end{cases} \tag{15}$$

$$\chi_4^{w,\ell}(x,y,z,w) = \begin{cases} w & ; \; x \geq y, \; x \geq z, \; x \geq w \\ \ell & ; \; x < y \text{ or } x < z \text{ or } x < w \end{cases} \tag{16}$$

By combining eqs. 13 to 16,

$$\chi_{q4}^{w,\ell}(x,y,z,w) = \left\{\begin{matrix}\text{probability that hand x improves} \\ \text{while hands y,z,w do not}\end{matrix}\right\} \cdot \{w\} \tag{17}$$

$$+\left\{\begin{matrix}\text{probability that hand x does not improve} \\ \text{and at least one of the hands y,z,w does}\end{matrix}\right\} \cdot \{\ell\}$$

$$+\left\{\text{all hands improve except hand y}\right\} \cdot \left\{\chi_3^{w,\ell}(x,z,w)\right\}$$

$$+\left\{\text{all hands improve except hand z}\right\} \cdot \left\{\chi_3^{w,\ell}(x,y,w)\right\}$$

$$+\left\{\text{all hands improve except hand w}\right\} \cdot \left\{\chi^{w,\ell}(x,y,z)\right\}$$

$$+\left\{\text{only hands x and y improve}\right\} \cdot \left\{\chi_2^{w,\ell}(x,y)\right\}$$

$$+\left\{\text{only hands x and z improve}\right\} \cdot \left\{\chi_2^{w,\ell}(x,z)\right\}$$

$$+\left\{\text{only hands x and w improve}\right\} \cdot \left\{\chi_2^{w,\ell}(x,w)\right\}$$

$$+\left\{\begin{matrix}\text{all hands improve or all fail} \\ \text{to improve concurrently}\end{matrix}\right\} \cdot \left\{\chi_4^{w,\ell}(x,y,z,w)\right\}$$

From eqs. 7 and 8, eq.17 reduces to

$$\chi_{q4}^{w,\ell}(x,y,z,w) = wq(1-q)^3 + \ell\{(1-q)[1-(1-q)^3]\} \quad (18)$$

$$+ q^3(1-q).\{\chi_3^{w,\ell}(x,z,w) + \chi_3^{w,\ell}(x,y,w) + \chi_3^{w,\ell}(x,y,z)\}$$

$$+ q^2(1-q)^2\{\chi_2^{w,\ell}(x,y) + \chi_2^{w,\ell}(x,z) + \chi_2^{w,\ell}(x,w)\}$$

$$+ \{q^4 + (1-q)^4\}\chi_4^{w,\ell}(x,y,z,w).$$

Now, as before, define

$$XQ_4^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix} = \int_{x=a_1}^{b_1}\int_{y=a_2}^{b_2}\int_{z=a_3}^{b_3}\int_{w=a_4}^{b_4} \chi_{q4}^{w,\ell}(x,y,z,w)\,dx\,dy\,dx\,dw \quad (19)$$

and

$$X_4^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix} = \int_{x=a_1}^{b_1}\int_{y=a_2}^{b_2}\int_{z=a_3}^{b_3}\int_{w=a_4}^{b_4} \chi_4^{w,\ell}(x,y,z,w)\,dx\,dy\,dz\,dw \quad (20)$$

Thus using eqs.11, 18, 19 and 20, and integrating, it can be shown that

$$XQ_4^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix} = \left\{\prod_{i=1}^{4}(b_i-a_i)\right\}.\left\{wq(1-q)^3+\ell(1-q)[1-(1-q)^3]\right\}$$

$$+ q^3(1-q)\left\{(b_4-a_4)X_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 \\ a_1 & a_2 & a_3 \end{pmatrix} + (b_3-a_3)X_3^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_4 \\ a_1 & a_2 & a_4 \end{pmatrix}\right.$$

$$\left. + (b_2-a_2)X_3^{w,\ell}\begin{pmatrix} b_1 & b_3 & b_4 \\ a_1 & a_3 & a_4 \end{pmatrix}\right\}$$

$$+q^2(1-q)^2\left\{(b_2-a_2)(b_3-a_3)X_2^{w,\ell}\begin{pmatrix} b_1 & b_4 \\ a_1 & a_4 \end{pmatrix} + (b_2-a_2)(b_4-a_4)X_2^{w,\ell}\begin{pmatrix} b_1 & b_3 \\ a_1 & a_3 \end{pmatrix}\right.$$

$$+ (b_3-a_3)(b_4-a_4)X_2^{w,\ell}\begin{pmatrix} b_1 & b_2 \\ a_1 & a_2 \end{pmatrix}\Bigg\}$$

$$+ \left\{q^4+(1-q)^4\right\} X_4^{w,\ell}\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix} \qquad (21)$$

Figure 9 below defines the game flow diagram and the strategy functions for the 4-PG in the same way that figure 8 was used to define the same aspects of the 3-PG. In this diagram all possible game outcomes, and their probabilities of occurrence, are presented. It is assumed that players 1,2,3 and 4 are dealt hands x,y,z and w respectively, and that each player's actions are governed by probability functions of the form $f_j^i(x)$ which are defined in terms of variables a_r^i. Define, $\underset{\sim}{a}$, the joint strategy vector for the 4-PG as follows:

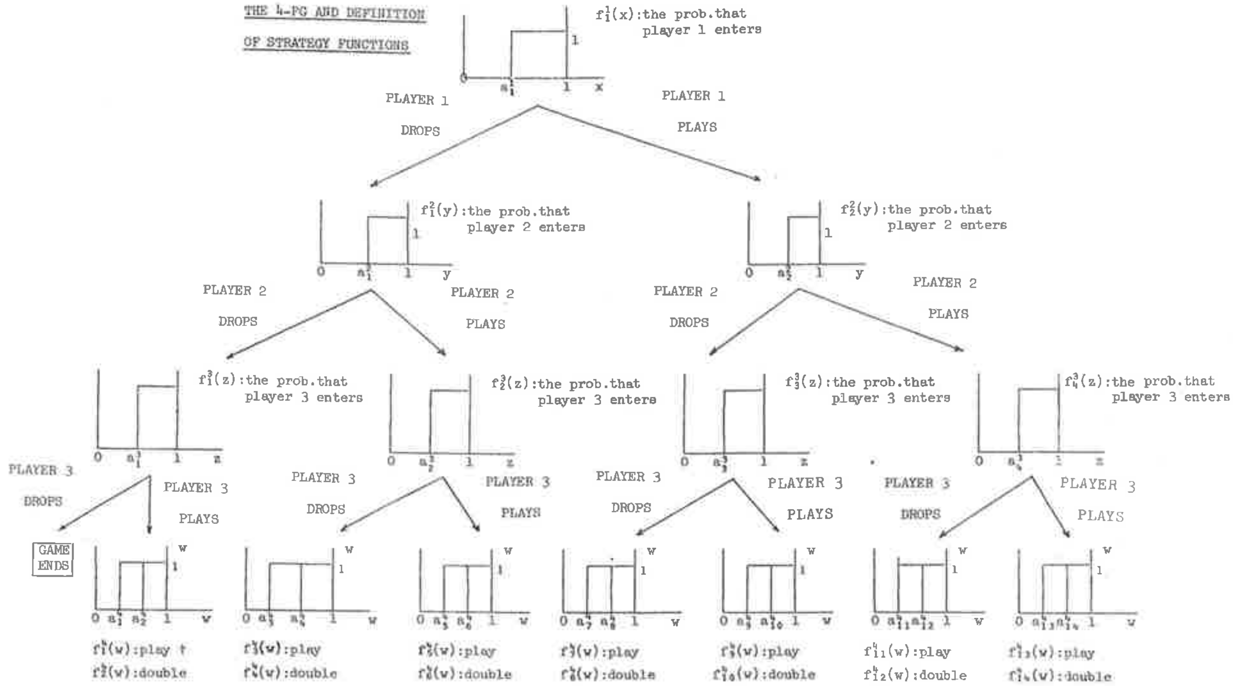
$$\underset{\sim}{a} = [a_1^1, a_1^2, a_2^2, a_1^3, a_2^3, a_3^3, a_4^3, a_1^4, a_2^4, a_3^4, a_4^4,$$
$$a_5^4, a_6^4, a_7^4, a_8^4, a_9^4, a_{10}^4, a_{11}^4, a_{12}^4, a_{13}^4, a_{14}^4]^T$$

The next step is to summarize all possible plays, and to specify the probability of each play occurring and the corresponding payoffs. This is done in table 14 which follows the same pattern as table 13 for the 3-PG, with the following exception.

It is clear that if player 1 declines to play in the 4-PG then the game becomes equivalent to a 3-PG (see figure 10). This fact may be used to reduce the amount of computation necessary to compute the payoff

GAME FLOW DIAGRAM FOR THE 4-PG AND DEFINITION OF STRATEGY FUNCTIONS

FIGURE 9



†note The above representation of the pair of functions $f_1^4(w)$ and $f_2^4(w)$ is chosen to conserve space. The meaning is that player 4 plays if $a_1^4 \le w < a_2^4$ and doubles if $a_2^4 \le w \le 1$.

TABLE 14

SUMMARY OF PLAY FOR THE 4-PG

	PLAY	PROBABILITY	Payoffs I	Payoffs II	Payoffs III	Payoffs IV
1.	D^1	$[1-f_1^1(x)]$	0	P_1^3 †	P_2^3 †	P_3^3 †
2.	$P^1D^2D^3D^4$	$f_1^1(x)[1-f_2^2(y)][1-f_3^3(z)][1-f_7^4(w)-f_8^4(w)]$	+1	0	0	-1
3.	$P^1D^2D^3P^4$	$f_1^1(x)[1-f_2^2(y)][1-f_3^3(y)]f_7^4(w)$	$\chi_{q_2}^{2,-2}(x,w)$	0	0	$\chi_{q_2}^{2,-2}(w,x)$
4.	$P^1D^2D^3D^4$	$f_1^1(x)[1-f_2^2(y)][1-f_3^3(y)]f_8^4(w)$	$\chi_{q_2}^{4,-4}(x,w)$	0	0	$\chi_{q_2}^{4,-4}(w,x)$
5.	$P^1D^2P^3D^4$	$f_1^1(x)[1-f_2^2(y)]f_3^3(y)[f_9^4(w)-f_{10}^4(w)]$	$\chi_{q_2}^{3,-2}(x,z)$	0	$\chi_{q_2}^{3,-2}(z,x)$	-1
6.	$P^1D^2P^3P^4$	$f_1^1(x)[1-f_2^2(y)]f_3^3(y)f_9^4(w)$	$\chi_{q_2}^{4,-2}(x,z,w)$	0	$\chi_{q_2}^{4,-2}(z,x,w)$	$\chi_{q_3}^{4,-2}(w,x,z)$
7.	$P^1D^2P^3D_b^4$	$f_1^1(x)[1-f_2^2(y)]f_3^3(y)f_{10}^4(w)$	$\chi_{q_3}^{8,-4}(x,z,w)$	0	$\chi_{q_3}^{8,-4}(z,x,w)$	$\chi_{q_3}^{8,-4}(w,x,z)$
8.	$P^1P^2D^3D^4$	$f_1^1(x)f_2^2(y)[1-f_4^3(z)][1-f_{11}^4(w)-f_{12}^4(w)]$	$\chi_{q_2}^{3,-2}(x,y)$	$\chi_{q_2}^{3,-2}(y,x)$	0	-1
9.	$P^1P^2D^3P^4$	$f_1^1(x)f_2^2(y)[1-f_4^3(z)]f_{11}^4(w)$	$\chi_{q_3}^{4,-2}(x,y,w)$	$\chi_{q_3}^{4,-2}(y,x,w)$	0	$\chi_{q_3}^{4,-2}(w,x,y)$
10.	$P^1P^2D^3D_b^4$	$f_1^1(x)f_2^2(y)[1-f_4^3(z)]f_{12}^4(w)$	$\chi_{q_3}^{8,-4}(x,y,w)$	$\chi_{q_3}^{8,-4}(y,x,w)$	0	$\chi_{q_3}^{8,-4}(w,x,y)$
11.	$P^1P^2P^3D^4$	$f_1^1(x)f_2^2(y)f_4^3(z)[1-f_{13}^4(w)-f_{14}^4(w)]$	$\chi_{q_3}^{5,-2}(x,y,z)$	$\chi_{q_3}^{5,-2}(y,x,z)$	$\chi_{q_3}^{5,-2}(z,x,y)$	-1
12.	$P^1P^2P^3P^4$	$f_1^1(x)f_2^2(y)f_4^3(z)f_{13}^4(w)$	$\chi_{q_4}^{6,-2}(x,y,z,w)$	$\chi_{q_4}^{6,-2}(y,x,z,w)$	$\chi_{q_3}^{6,-2}(z,x,y,w)$	$\chi_{q_4}^{6,-2}(w,x,y,z)$
13.	$P^1P^2P^3D_b^4$	$f_1^1(x)f_2^2(y)f_4^3(z)f_{14}^4(w)$	$\chi_{q_4}^{12,-4}(x,y,z,w)$	$\chi_{q_4}^{12,-4}(y,x,z,w)$	$\chi_{q_4}^{12,-4}(z,x,y,w)$	$\chi_{q_4}^{12,-4}(w,x,y,z)$

† P_1^3, P_2^3, P_3^3 are the payoffs for the 3-PG

functions for the 4-PG, by making use of the 3-PG payoff functions which have already been calculated, (see 5.2.3).

Let $P^4_i(\underset{\sim}{a})$ be the corresponding payoff to player i for joint strategy $\underset{\sim}{a}$. Then, using table 14 and figure 9, the functions $P^4_i(\underset{\sim}{a})$ may be calculated, in the usual way, and it is found that:

$$P^4_1(\underset{\sim}{a}) = (1-a^1_1)a^2_2a^3_3a^4_7 + a^2_2a^3_3XQ^{2,-2}_2\begin{pmatrix}1 & a^4_8\\ a^1_1 & a^4_7\end{pmatrix}$$

$$+ a^2_2a^3_3XQ^{4,-4}_2\begin{pmatrix}1 & 1\\ a^1_1 & a^4_8\end{pmatrix} + a^2_2a^4_9XQ^{3,-2}_2\begin{pmatrix}1 & 1\\ a^1_1 & a^3_3\end{pmatrix}$$

$$+ a^2_2XQ^{4,-2}_3\begin{pmatrix}1 & 1 & a^4_{10}\\ a^1_1 & a^3_3 & a^4_9\end{pmatrix} + a^2_2XQ^{8,-4}_3\begin{pmatrix}1 & 1 & 1\\ a^1_1 & a^3_3 & a^4_{10}\end{pmatrix} +$$

$$a^3_4a^4_{11}XQ^{3,-2}_2\begin{pmatrix}1 & 1\\ a^1_1 & a^2_2\end{pmatrix}$$

$$+ a^3_4XQ^{4,-2}_3\begin{pmatrix}1 & 1 & a^4_{12}\\ a^1_1 & a^2_2 & a^4_{11}\end{pmatrix} + a^3_4XQ^{8,-4}_3\begin{pmatrix}1 & 1 & 1\\ a^1_1 & a^2_2 & a^4_{12}\end{pmatrix} +$$

$$a^4_{13}XQ^{5,-2}_3\begin{pmatrix}1 & 1 & 1\\ a^1_1 & a^2_2 & a^3_4\end{pmatrix}$$

$$+ XQ^{6,-2}_4\begin{pmatrix}1 & 1 & 1 & a^4_{14}\\ a^1_1 & a^2_2 & a^3_4 & a^4_{13}\end{pmatrix} + XQ^{12,-4}_4\begin{pmatrix}1 & 1 & 1 & 1\\ a^1_1 & a^2_2 & a^3_4 & a^4_{14}\end{pmatrix}$$

$$P^4_2(\underset{\sim}{a}) = a^1_1P^3_1(a^2_1, a^3_1, a^3_2, a^4_1, a^4_2, a^4_3, a^4_4, a^4_5, a^4_6)$$

$$+ a^3_4a^4_{11}XQ^{3,-2}_2\begin{pmatrix}1 & 1\\ a^2_2 & a^1_1\end{pmatrix} + a^3_4XQ^{4,-2}_3\begin{pmatrix}1 & 1 & a^4_{12}\\ a^2_2 & a^1_1 & a^4_{11}\end{pmatrix} +$$

$$a^3_4XQ^{8,-4}_3\begin{pmatrix}1 & 1 & 1\\ a^2_2 & a^1_1 & a^4_{12}\end{pmatrix}$$

$$+ a^4_{13} XQ^{5,-2}_3 \begin{pmatrix}1 & 1 & 1\\ a^2_2 & a^1_1 & a^3_4\end{pmatrix} + XQ^{6,-2}_4 \begin{pmatrix}1 & 1 & 1 & a^4_{14}\\ a^2_2 & a^1_1 & a^3_4 & a^4_{13}\end{pmatrix} +$$

$$XQ^{12,-4}_4 \begin{pmatrix}1 & 1 & 1 & 1\\ a^2_2 & a^1_1 & a^3_4 & a^4_{14}\end{pmatrix}$$

$$P^4_3(\underset{\sim}{a}) = a^1_1 P^3_2(a^2_1, a^3_1, a^3_2, a^4_1, a^4_2, a^4_3, a^4_4, a^4_5, a^4_6)$$

$$+ a^2_2 a^4_9 XQ^{3,-2}_2 \begin{pmatrix}1 & 1\\ a^3_3 & a^1_1\end{pmatrix} + a^2_2 XQ^{4,-2}_3 \begin{pmatrix}1 & 1 & a^4_{10}\\ a^3_3 & a^1_1 & a^4_9\end{pmatrix} +$$

$$+ a^2_2 XQ^{8,-4}_3 \begin{pmatrix}1 & 1 & 1\\ a^3_3 & a^1_1 & a^4_{10}\end{pmatrix}$$

$$+ a^4_{13} XQ^{5,-2}_3 \begin{pmatrix}1 & 1 & 1\\ a^3_4 & a^1_1 & a^2_2\end{pmatrix} + XQ^{6,-2}_4 \begin{pmatrix}1 & 1 & 1 & a^4_{14}\\ a^3_4 & a^1_1 & a^2_2 & a^4_{13}\end{pmatrix} +$$

$$+ XQ^{12,-4}_4 \begin{pmatrix}1 & 1 & 1 & 1\\ a^3_4 & a^1_1 & a^2_2 & a^4_{14}\end{pmatrix}$$

$$P^4_4(\underset{\sim}{a}) = a^1_1 P^3_3(a^2_1, a^3_1, a^3_2, a^4_1, a^4_2, a^4_3, a^4_4, a^4_5, a^4_6)$$

$$- (1-a^1_1) a^2_2 a^3_3 a^4_7$$

$$+ a^2_2 a^3_3 XQ^{2,-2}_2 \begin{pmatrix}a^4_8 & 1\\ a^4_7 & a^1_1\end{pmatrix} + a^2_2 a^3_3 XQ^{4,-4}_2 \begin{pmatrix}1 & 1\\ a^4_8 & a^1_1\end{pmatrix} -$$

$$a^2_2 a^4_9 (1-a^1_1)(1-a^3_3)$$

$$+ a^2_2 XQ^{4,-2}_3 \begin{pmatrix}a^4_{10} & 1 & 1\\ a^4_9 & a^1_1 & a^3_3\end{pmatrix} + a^2_2 XQ^{8,-4}_3 \begin{pmatrix}1 & 1 & 1\\ a^4_{10} & a^1_1 & a^3_3\end{pmatrix} -$$

$$a^3_4 a^4_{11} (1-a^1_1)(1-a^2_2)$$

$$+ a^3_4 XQ^{4,-2}_3 \begin{pmatrix}a^4_{12} & 1 & 1\\ a^4_{11} & a^1_1 & a^2_2\end{pmatrix} + a^3_4 XQ^{8,-4}_3 \begin{pmatrix}1 & 1 & 1\\ a^4_{12} & a^1_1 & a^2_2\end{pmatrix} -$$

$$a^4_{13} (1-a^1_1)(1-a^2_2)(1-a^3_4)$$

$$+ XQ^{6,-2}_4 \begin{pmatrix}a^4_{14} & 1 & 1 & 1\\ a^4_{13} & a^1_1 & a^2_2 & a^3_4\end{pmatrix} + XQ^{12,-4}_4 \begin{pmatrix}1 & 1 & 1 & 1\\ a^4_{14} & a^1_1 & a^2_2 & a^3_4\end{pmatrix}$$

5.3.2 Special method to speed the solution of the 4-PG

It is not practical to apply the LAH algorithm directly to the task of solving the 4-PG for the following reason. Experimental evidence showed that:

(a) Each function evaluation for the 4-PG took approximately $\frac{1}{40}$ sec. of c.p. time.

(b) 2016 function evaluations were required per cycle of the LAH algorithm.

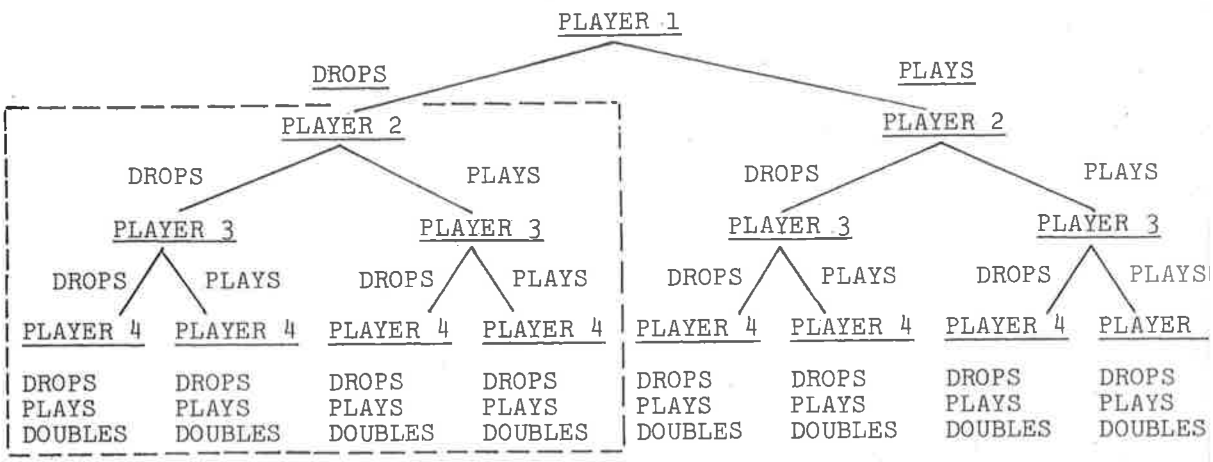
Hence, time per cycle was approximately 50 secs. Since many cycles (see chapter 3) are needed to find the e.p. it would be advantageous if some way could be found to reduce the cycle time.

In fact the number of function calls required per cycle may be reduced in the following way. Consider the tree diagram given in figure 10. The section of the tree diagram enclosed in the dashed rectangle is a subset of the 4-PG which arises when player 1 declines to play. This subset is exactly equivalent to the 3-PG. Thus, the solution already found for the 3-PG may be used in this section of the 4-PG. (Note that this fact has already been used in table 14). This then reduces the number of unknown variables in the 4-PG, and, as a consequence, experiments show that the number of function calls per cycle of the LAH algorithm is reduced to 666. The evaluation time per function remains at approximately $\frac{1}{40}$

FIGURE 10

EQUIVALENCE BETWEEN THE 3-PG AND

THE 4-PG WHEN PLAYER 1 DECLINES TO PLAY



of a second, and thus the overall execution time per cycle is reduced to 14 secs. This is a factor of 3 times faster than before.

Constraints

An examination of figure 9 shows that:-

$$0 \leqslant a_j^1 \leqslant 0 \quad \text{for all variables} \quad a_j^1$$

and

$$a_1^4 \leqslant a_2^4, \quad a_3^4 \leqslant a_4^4, \quad a_5^4 \leqslant a_6^4, \quad a_7^4 \leqslant a_8^4$$

$$a_9^4 \leqslant a_{10}^4, \quad a_{11}^4 \leqslant a_{12}^4, a_{13}^4 \leqslant a_{14}^4$$

5.3.3 Solution of 4-PG

With the above data it is possible to solve the 4-PG (with $q = 0.287$) by using the LAH algorithm. The minimum step size used by the algorithm was 0.00045 and the execution time was approximately 320 seconds of c.p. time. The solution is given in table 15 below.

5.4 Checking the 3-PG and 4-PG payoff functions

Both the 3-PG and 4-PG payoff functions were tested in the same way, hence details are only given for one of the above cases. The 3-PG case was chosen to maximize simplicity of presentation.

Two tests were carried out.

(a) Checking that the sum of the payoff functions is zero.

Since the 3-PG is a zero-sum game (see chapter 3), it follows that for any given joint strategy vector $\underset{\sim}{a}$ (see eq.12)

TABLE 15

SOLUTION TO THE 4-PG

Player	Expected Payoff	Variables			
1	0.113	$a_1^1=0.790$			
2	0.129	$a_1^2=0.715$	$a_2^2=0.815$		
3	0.169	$a_1^3=0.615$	$a_2^3=0.800$	$a_3^3=0.870$	$a_4^3=0.925$
4	-0.411	$a_1^4=0.620$	$a_2^4=0.815$	$a_3^4=0.730$	$a_4^4=0.830$
		$a_5^4=0.825$	$a_6^4=0.885$	$a_7^4=0.800$	$a_8^4=0.880$
		$a_9^4=0.830$	$a_{10}^4=0.900$	$a_{11}^4=0.830$	$a_{12}^4=0.920$
		$a_{13}^4=0.840$	$a_{14}^4=0.999$		

$$P_1^3(\underset{\sim}{a}) + P_2^3(\underset{\sim}{a}) + P_3^3(\underset{\sim}{a}) = 0 \qquad (22)$$

A random number generator was used to generate an acceptable strategy vector $\underset{\sim}{a}$, and this parameter was substituted into the L.H.S. of eq.22 to check its validity. This process was repeated a large number of times, and in each case eq.22 was satisfied.

(b) Use of simulation as an alternative method of calculating the payoff function.

The payoff functions for the 3-PG may be approximated by using simulation, in the manner to be described below. Given some strategy variable, $\underset{\sim}{a}$, many games are simulated, and the average expected payoff accruing to each player, as a result of this particular value of $\underset{\sim}{a}$, is then calculated. Each individual game is simulated in the following way. A random number generator (see appendix B) is used to deal hands x,y, and z to players 1,2 and 3 respectively. Next, the joint strategy vector, $\underset{\sim}{a}$, predicts the course of events in this game according to figure 8. For example, if $x < a_1^1$ then player 1 drops. Next, if $y \geqslant a_1^2$, say, player 2 will play, while player 3 will drop if $z < a_1^3$.

Finally, the random number generator is used to effect hand improvement according to eq.1 in chapter 4. This is done by generating a random number, r, $0 \leqslant r \leqslant 1$, and allowing the hand x to improve to $X = x+2$ if

$0 \leqslant r \leqslant q$ $(q = 0.287)$. If $q \leqslant r \leqslant 1$ then the hand does not improve, and $X=x$. After this has been done for all hands, the eventual winner is decided in the usual way.

Thus, for any given $\underset{\sim}{a}$ the payoff functions may be calculated in 2 ways. First, by using the analytic results given in 5.2.3 and secondly, using simulation to give an approximate answer. This experiment was carried out for several different values of $\underset{\sim}{a}$, and the 2 sets of results agreed each time. The details of the least favourable such result will now be given.

Let

$$\underset{\sim}{a} = [0.715, 0.615, 0.800, 0.645, 0.815, 0.730, 0.830, 0.825, 0.885]^T$$

The 3 payoff functions for this value of $\underset{\sim}{a}$ were calculated analytically. Next, the same payoff functions were approximated by simulating 100 games. This procedure was repeated several times, each time using progressively higher numbers of games in the simulation. Then the results obtained were analyzed in the following way. Consider the payoffs obtained for the 3 players by simulating 100 games. The relative error between each of the 3 player's payoffs, and the corresponding payoffs calculated from the payoff functions, may be computed, and the maximum relative error of the 3 determined. This turns out to be 64%.

This procedure is repeated for the payoffs obtained by simulating 10,000, 30,000 and 100,000 games, and the results are graphically represented by figure 11. Figure 11 shows that the payoffs obtained from simulation approach the analytic payoffs, as the number of games simulated increases. Since the simulation of 100,000 games took approximately 30 seconds of c.p. time, and convergence appeared slow, the experiment was terminated at this point. Clearly, simulation, as a method of obtaining payoff functions, is slow and not sufficiently accurate for the purposes of this study. It is, nevertheless, a useful standard against which the analytically evaluated payoff functions may be tested.

The same method was used to check the payoff functions for the 4-PG.

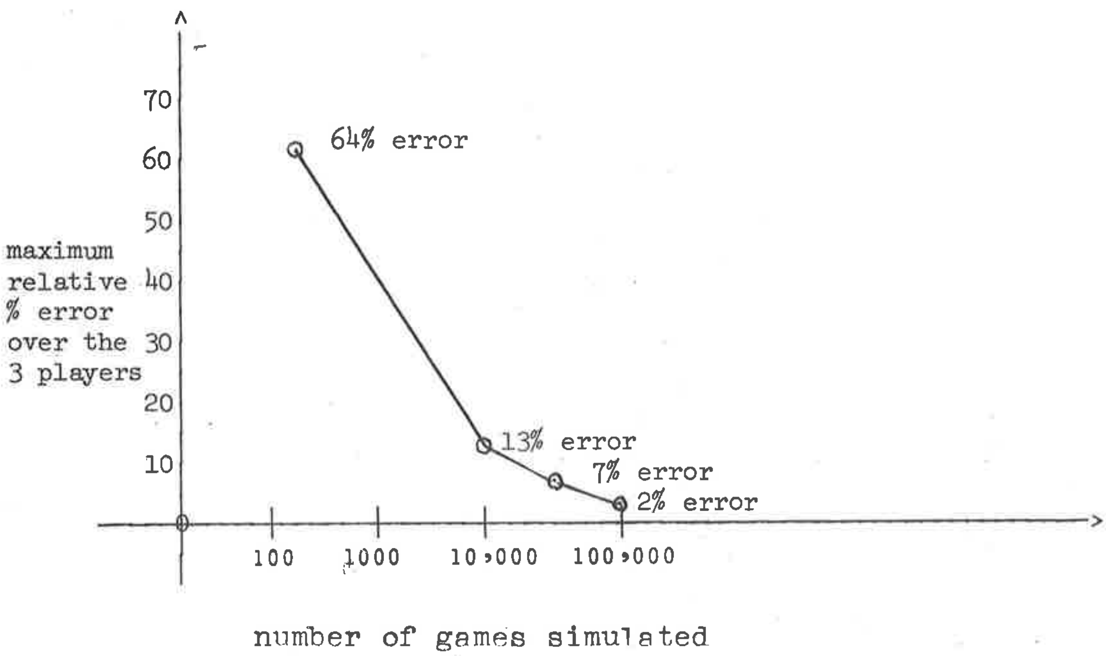
5.5 Discussion of results

5.5.1 Comparison between the 2-PG and 3-PG

Define the game 2-PG* to be the 3-PG where the first player has declined to play. Thus the 2-PG* is equivalent to the 2-PG (compare with the 4-PG reducing to the 3-PG when player 1 does not play, as described in figure 10). In this section the differences between the strategies adopted by player 1 in the 2-PG and 2-PG* will be discussed.

FIGURE 11

COMPARISON BETWEEN ANALYTIC PAYOFF FUNCTIONS FOR 3-PG, AND RESULTS OBTAINED FROM SIMULATION



In 4.3 it was shown that, in the case of the 2-PG, player 1 will play, only if his hand x is such that $0 \le x \le 0.03$ or $0.64 \le x \le 1.0$ (see chapter 4, table 11). However, because of differences in the assumed form of the strategy functions for the 2-PG* (see discussion 5.2.2), the solution to the 2-PG* shows that player 1 will only play if his hand x is such that $x \ge 0.61$ (see 5.2.4). It will be shown that, for practical purposes, these two strategies are identical.

First, in the case of a hand x, $x \ge 0.64$, both strategies require the player to enter the game and are thus identical. Now consider what happens when $0.61 \le x \le 0.64$. The 2-PG* strategy requires the player to enter, while the 2-PG strategy does not.

However, if player 1 does enter, and he is looked at, he can only possibly win if the second player's hand y is in the range $0.62 \le y \le 0.64$, since the second player will only play on a hand $y \ge 0.62$. Thus summarising the above, if player 1 has a hand x, $0.61 \le x \le 0.64$, and player 2 has a hand y, $0.62 \le y \le 0.64$, then player 1 should win half the time. That is, his chances of winning in such a situation are

$$\tfrac{1}{2} \times (0.64-0.61) \times (0.64-0.62) = 0.0003.$$

Hence, for practical purposes, player 1 may ignore the possibility of winning in this position.

Conversely, if x is such that $0 \le x \le 0.03$ then the 2-PG strategy requires a player to enter, while the 2-PG* strategy does not. Using the same arguments as above it follows that in this second case the practical result again is that player 1 may expect to lose if player 2 looks.

Thus, it has been demonstrated that the practical effects of player 1 either playing on hands x, $0.61 \le x \le 0.64$, or of playing on hands x, $0 \le x \le 0.03$, are the same. Since the frequencies of both occurrences are equal $(0.64-0.61 = 0.03)$, it follows that the two apparently dissimilar strategies have a very similar practical effect.

5.5.2 Practical Interpretation of the solution to the 4-PG

This section will discuss the practical interpretation of the solution to the 4-PG.

Chapter 1 presented a survey of the literature on poker-like games. It was then pointed out that this search failed to find any games that could be directly related to any commonly played variety of poker. In the ensuing discussion it will be shown how the solutions obtained here may be used in this way.

Before this can be done two preliminary results are required.

(1) Relating a real poker hand to some hand x in the 4-PG

Any real poker hand, h, may be related to a hand x in the 4-PG, in the following way. The work on poker simulation required the definition (see 2.8) of a function $f_b(h)$ which gave the probability of a hand h beating any other randomly dealt hand. However, any hand x in the 4-PG, by virtue of being evenly randomly distributed on (0,1) (see 4.1), is numerically equal to the probability of beating any other randomly dealt hand y. Thus, it follows that any poker hand h is equivalent to a hand $x = f_b(h)$ in the 4-PG.

The algorithms developed in chapter 2 may be used to evaluate $f_b(h)$, the probability that a poker hand h has of winning. Table 16 gives the values of these probabilities for various types of hands, and, as a result of the above discussion, an equivalence is thus established between h, a real poker hand, and x, a hand dealt in the 4-PG.

(2) Correspondence between a given hand in the 4-PG and a poker hand

As a consequence of (1) above it is possible to provide an interpretation of any hand x in the 4-PG in terms of a real poker hand h. This procedure is best illustrated by an example.

TABLE 16

EQUIVALENCE BETWEEN HANDS IN THE 4-PG AND REAL POKER HANDS

POKER HAND, h	EQUIVALENT HAND IN 4-PG $x = f_b(h)$
NOTHING	
PAIR OF 2's OR BETTER	0.50 OR HIGHER
PAIR OF 3's OR BETTER	0.53 OR HIGHER
PAIR OF 4's OR BETTER	0.57 OR HIGHER
PAIR OF 5's OR BETTER	0.60 OR HIGHER
PAIR OF 6's OR BETTER	0.64 OR HIGHER
PAIR OF 7's OR BETTER	0.67 OR HIGHER
PAIR OF 8's OR BETTER	0.71 OR HIGHER
PAIR OF 9's OR BETTER	0.74 OR HIGHER
PAIR OF 10's OR BETTER	0.78 OR HIGHER
PAIR OF J's OR BETTER	0.81 OR HIGHER
PAIR OF Q's OR BETTER	0.85 OR HIGHER
PAIR OF K's OR BETTER	0.88 OR HIGHER
PAIR OF Aces OR BETTER	0.92 OR HIGHER
THREE OF A KIND OR BETTER	0.98 OR HIGHER

It was shown in table 15 that the optimal solution to the 4-PG had $a_1^1 = 0.790$. By referring to the original strategy function definitions (figure 9) it may be seen that this requires player 1 to play if his hand x is such that $x \geq a_1^1$, and drop if $x < a_1^1$.

Table 15 shows that a pair of 10's or better is equivalent to a hand x, $x \geq 0.78$, while a pair of Jacks is equivalent to a hand x, $x \geq 0.81$. Thus $x \geq 0.79$ means that hands h weaker than a pair of 10's are never played on, while all hands stronger than a pair of 10's are always good enough. However, it is not immediately clear what action should be taken when holding exactly a pair of 10's, and this problem is resolved in the following way.

First, two hands both containing one pair of 10's may be ordered on the basis of the 3 remaining cards (see table 1, chapter 2). Thus $x \geq 0.79$ means that only pairs of 10's of above a certain strength (judged by the 3 non-paired cards) are sufficiently good to play on. Since the probability of obtaining better than or equal to one pair of 10's is 0.78 and better than or equal to a pair of Jacks is 0.81, then the probability of obtaining exactly 1 pair of 10's is 0.81-0.78 = 0.03. If now only pairs of 10's are considered when $x \geq 0.79$, then the probability of obtaining such a hand is 0.81-0.79 = 0.02. Thus, this analysis shows that $x \geq 0.79$ implies that only $\frac{0.02}{0.03} = \frac{2}{3}$

of all pairs of 10's dealt will be sufficiently good to play on.

Since, in a real game of poker, the 3 non-paired cards in the hand are usually discarded, from a practical point of view, all pairs of 10's are equivalent. Thus, using the above two ideas, it is possible to interpret $x \geq 0.79$ to mean that a pair of 10's is only played on $\frac{2}{3}$ of the time. This strategy would mean that a player could make a random choice on whether or not to play with a pair of 10's, provided that, on the average, he played $\frac{2}{3}$ of the time.

If, as mentioned by von Neumann, (27), one of the objects of bluffing is to create uncertainty in the opposing players, then, the above apparent randomness (i.e. playing only two thirds of the time with a pair of tens) in the strategy, could be interpreted as a form of bluff.

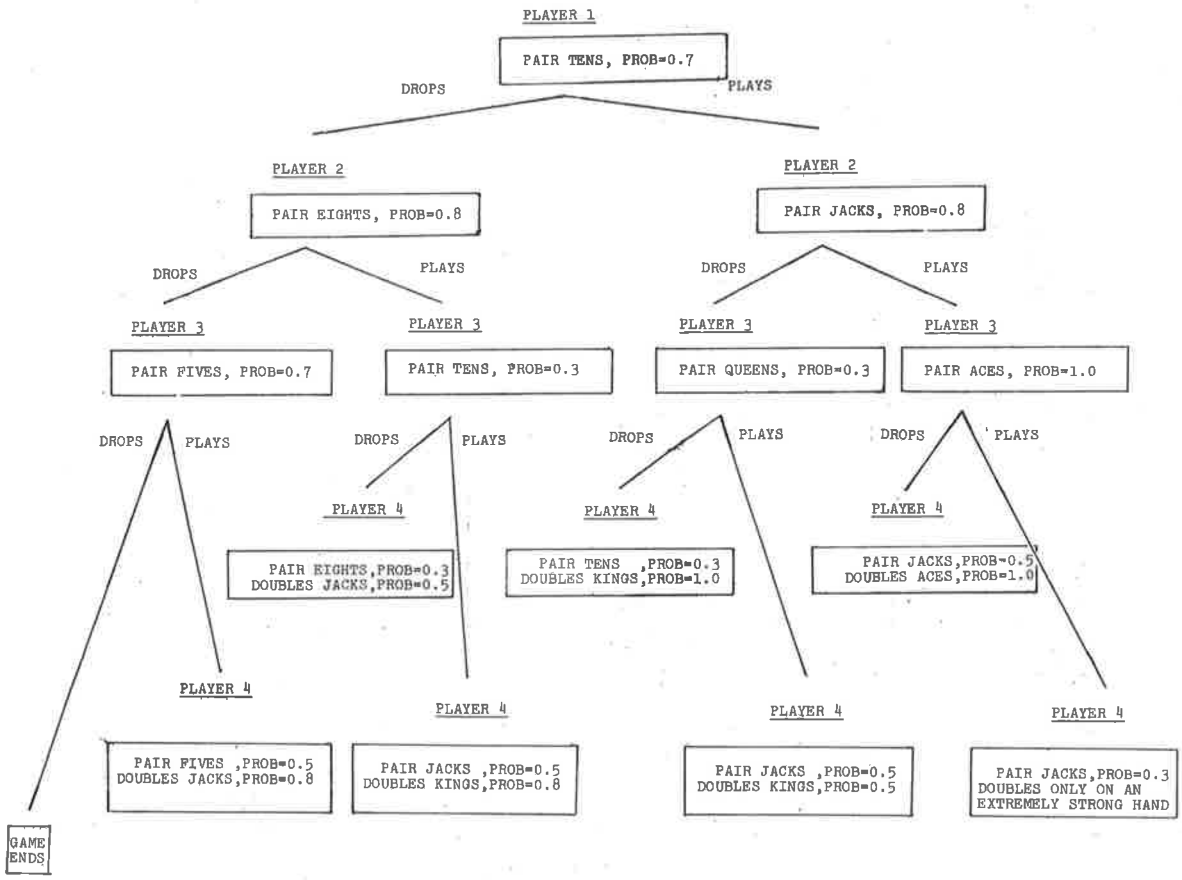
Each of the variables a_j^i may be interpreted in the above manner, and the resultant, overall, strategy, is given in figure 12. It should be noted that in this figure, only the probabilities of playing with the critical hands are specified, as it is automatically assumed that hands weaker than this are never played on, while stronger hands are always played on.

5.5.3 Discussion of solution to the 4-PG

This section will discuss the solution to the 4-PG given in table 15. A comparison will then be drawn

FIGURE 12 : SOLUTION TO THE 4-PG EXPRESSED IN TERMS OF POKER HANDS

In the diagram below statements of the form PAIR TENS, PROB=0.7 should be interpreted in the following way. A player should never play if his hand is weaker than a pair of tens, and always play if his hand is stronger than a pair of tens. However, if he has exactly a pair of tens then he should only play with probability 0.7.



between the analytic solution and the strategies adopted by experienced players, as recorded in books on the subject, (6,8,30).

By studying figure 12 the following points may be noted.

(a) <u>When to play first</u>

The minimum hand required before a player will enter the game, given that no other player has yet anted, is found from figure 12 and given in table 17 below.

(b) <u>Conditions under which the last player, drops, plays or doubles.</u>

Generally the last player continues to play if he has a hand at least as good as the expected hand of the first player to make an ante. The last player then generally doubles holding:

(i) Jacks or better if 1 other player is in.

(ii) Kings or Aces if 2 other players are in.

(iii) Only an extremely strong hand if more than 2 players are in.

(c) <u>Expectations of various players</u>

Table 15 shows that players have the following expectations (of winnings) in the 4-PG.

Player	Expectation
1	0.113
2	0.129
3	0.169
4	-0.411

TABLE 17

MINIMUM HAND WHEN FIRST TO PLAY: THEORETICAL RESULT

n, number of players to follow	Minimum hand required
3	pair of 10's
2	pair of 8's
1	pair of 5's

Coffin, (6), makes the following general comments about poker. First, since the last player is compelled to ante (while the others are not) he suffers a disadvantage. This is also evident in the solution because the last player is the only one to have a negative expectation for the game.

Second, since out of players 1,2 and 3, the later players have an advantage over the earlier players, in that they have less players ahead of them who have not yet anted, and can thus arrive at a more informed decision. Again, this effect is demonstrated in the expectations of the 3 players.

The most striking aspect of the solution is the degree of disadvantage incurred by player 4 in having to make an initial compulsory ante.

5.5.4 Comparison of analytic solution with the strategies used by experienced players

Over many years certain strategies have evolved for the type of game considered here (see (6,8,30)). Even though the rules of some of these games may differ slightly, the salient points of these strategies remain largely invariant, and are listed below. These practical strategies will then be compared to the theoretical strategies found here.

(a) When to play first

It is generally agreed, (6,8) that the minimum hand required when first to play, is as given in table 18 below.

It may be seen that table 18 and table 17 (theoretical results) agree for $n = 2$ and 3, with the exception of the minimum hand requirements for the second last player ($n=1$). Whereas, it is generally recommended that any pair is sufficient under these conditions, the theoretical result shows that a pair of 5's is required. This discrepancy may have the following explanation.

First, it is possible that the generally recommended strategy is not correct in this instance. Certainly, the concepts of good play in poker have changed over the last 100 years (see (8)).

Second, table 18 is quoted in various books as not only applying to the particular version of the game solved here (30,32), but also to a version of this game where the second to last player makes a compulsory ante, half the size of that put in by the blind. Obviously, in this case, the second last player would play on a somewhat weaker hand as he is already partially committed. Thus poker books generally consider that slight variations such as this in the rules do not affect the optimal strategy, and that the same strategies may be used in similar types of game. This idea appears correct when the first and second player in the 4-PG are considered, but seems to break down for the second last player.

TABLE 18

MINIMUM HAND WHEN FIRST TO PLAY; PRACTICAL CRITERIA

n, the number of players to follow	Minimum hand requirement	Corresponding hand x in the 4-PG
6	pair of Aces	0.92
5	pair of Kings	0.90
4	pair of Queens	0.86
3	pair of 10's	0.80
2	pair of 8's	0.71
1	any pair	0.50

Application of table 18 to the simulation of poker

It has been indicated in chapter 2 that the results of table 18 have been applied to the poker simulator. It was found that a quadratic function of n,

$$f(n) = -0.01n^2 + 0.13n + 0.50$$

would closely approximate the value of x, (hand strength) for any given value of n (number of players to follow). This function, $x = f(n)$, is employed in the poker simulator, as described in section 2.9.

(b) On playing after another player

This discussion will consider the strategy to be followed if another player has already entered the game. The last player's strategy in this situation, however, will not be treated here, but will be discussed separately in the next section.

Books on the subject, (30,32) agree that players should have better than the minimum expected hand of the last player to enter, before making an ante. If the solution given in figure 12 is examined, then it may be seen to agree with this principle. For example, if player 1 enters then he must have a hand of at least a pair of 10's. In this case player 2 will only enter if his hand y is at least a pair of Jacks. Similar patterns may be found throughout this solution.

This behaviour may be approximately described using a function $g(\ell) = \ell + \frac{1}{3}(1-\ell)$ where ℓ is the last minimum expected hand that was entered, and $g(\ell)$ gives the next minimum required hand to enter. This function is used in the poker simulator (chapter 2).

(c) Play by the last player

Books on the subject, (30,32) do not state any specific requirements on the minimum hands required by the last player. Only 2 general observations are made.

First, the last player should only play if his hand compares favourably with the minimum expected hands of other players who have made an ante. This same criterion is seen to hold for the solution to the 4-PG (figure 12). For example, if players 1 and 2 drop, while 3 plays, then player 4 expects player 3 to have at least a pair of 5's. Accordingly, he only plays on a pair of 6's.

Second, it is recommended that the last player should only double when there is a good chance that he has the best hand. An examination of figure 12 shows that this criterion also holds for the analytic solution. Consequently, it may be noted that if player 1 drops while players 2 and 3 play, then player 4 may assume that player 2 holds at least a pair of 8's, while player 3 must have at least a pair of 10's. Thus player 4 plays on a pair of Jacks or better, and doubles only on a pair of Kings or above.

CHAPTER 6

APPLICATIONS OF THE WORK CARRIED OUT IN THIS STUDY

6.1. Introduction

As a sequel to the work of this study, two practical applications will now be given, which illustrate the inter-disciplinary nature of this work.

6.2. Application of game theoretic methods to a problem in networks

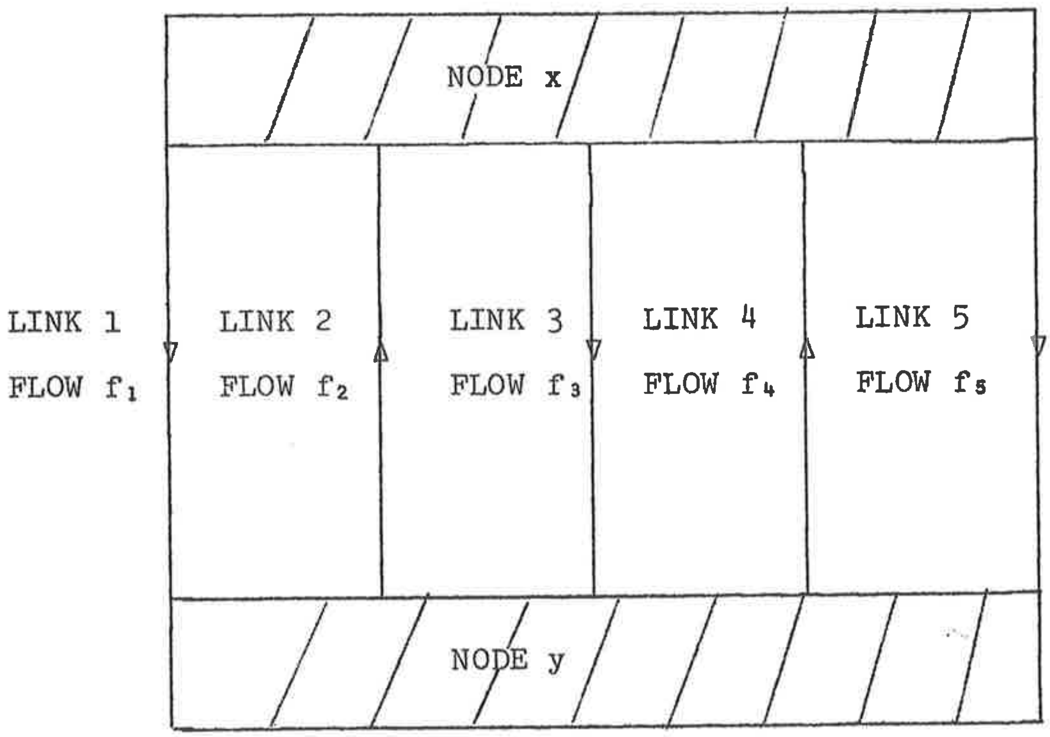
In this section a game theoretic approach to a particular network problem will be formulated. The treatment presented here will only give sufficient detail to illustrate the principle underlying this new approach. A more detailed paper (Harris (16)) on the subject which applies this method to a problem in telephone networks, has been written, following a discussion of these ideas with the author.

6.2.1. Theoretical background

The ideas involved in the application of game theoretic methods to networks may be conveniently explained by means of an example taken from Dafermos, (7).

Figure 13 below shows two towns, labelled node x and node y. There are 5 one-way roads between node x and node y, called links. Each link can only carry traffic in a single direction, and the amount of traffic being carried on link i is called the flow, f_i.

FIGURE 13 AN EXAMPLE OF A SIMPLE NETWORK



Define the vector $\underset{\sim}{f}$ to be the total flow vector where $\underset{\sim}{f} = [f_1, f_2, f_3, f_4, f_5]^T$.
Suppose that for any given total flow vector $\underset{\sim}{f}$ there is a cost $c_i(\underset{\sim}{f})$ associated with each link i and assume that the network is such that:

$$c_1(\underset{\sim}{f}) = 4f_1^2 + 2f_1f_2 + 900f_1$$
$$c_2(\underset{\sim}{f}) = 6f_2^2 + 2f_1f_2 + 900f_2$$
$$c_3(\underset{\sim}{f}) = 6f_3^2 + 3f_3f_4 + 820f_3$$
$$c_4(\underset{\sim}{f}) = 8f_4^2 + 3f_3f_4 + 820f_4$$
$$c_5(f) = 5f_5^2 + 1220f_5 \qquad (1)$$

This means, for example that, $c_1(\underset{\sim}{f})$, the cost of carrying traffic f_1 on link 1, depends not only on, f_1, the traffic carried by link 1, but also on f_2, the traffic on link 2. In practice, this type of interaction may occur if links 1 and 2 form one two way road, and thus the flow on one side of the road will, to some degree, affect the flow of traffic on the other side.

Next define $d_{(x,y)}$ as the travel demand from node x to node y. In this particular network it is required that 120 units of flow be transported from node x to node y, and 120 units from node y to node x. Thus

$$d_{(x,y)} = 120$$
$$d_{(y,x)} = 120 \qquad (2)$$

Since, from figure 13, total flow from node x to node y is $f_1 + f_3 + f_5$ and from node y to node x it is f_2+f_4, then eq.2 becomes

$$f_1 + f_3 + f_5 = 120$$
$$f_2 + f_4 = 120 \qquad (3)$$

where $f_i \geq 0$ for $i=1,...5$.

The problem is to apportion the flow vector $\underset{\sim}{f}$ in such a way that while eq.3 is satisfied, the costs associated with the network, given by eq.1, are in some sense minimized. Two criteria of minimization which are commonly used (see Potts, (28) and Dafermos, (7)), are given below, while a new, game theoretic approach, will be defined in the next section.

(1) System Optimization

System optimization consists of minimizing the function

$$c(\underset{\sim}{f}) = \sum_{i=1}^{5} c_i(\underset{\sim}{f}) \qquad (4)$$

and therefore gives the lowest possible overall cost, while ensuring that the constraints are satisfied.

(2) User optimization

Dafermos, (7), states that for the problem being considered here, if each unit of flow is considered to be an individual car, and the cost per unit flow represents average time taken to complete the journey, then the user optimized pattern will occur if each individual is free to choose his own route independently. It is assumed that he does this in such a way that his own travelling time is minimized. A mathematical formulation of this criterion will not be given here as it is not relevant to this study, but it may be found in Dafermos, (7).

6.2.2 Game theoretic network optimization

A new criterion of minimization, which proved to be of theoretical and practical importance (see (16)), is proposed in this section.

Consider the network given in figure 13 as a 2-person non-cooperative game, defined in the following way. Player 1 must transport 120 units of flow ($d_{(x,y)} = 120$) from node x to node y. He is free to direct this flow in any way he chooses within the limitations of the constraints, along links 1, 3 and 5. Player 2 must likewise transport 120 units of flow from node y to node x ($d_{(y,x)} = 120$). Again he is free to direct this flow in any way that he desires along links 2 and 4. Thus according to earlier definitions, the cost to player 1 for any given flow configuration $\underset{\sim}{f}$ is given by $c_1(\underset{\sim}{f})+c_3(\underset{\sim}{f})+c_5(\underset{\sim}{f})$, and player 1 seeks to minimize this cost, and hence maximize his payoff function, $P_1(\underset{\sim}{f})$, where

$$P_1(\underset{\sim}{f}) = -c_1(\underset{\sim}{f}) - c_3(\underset{\sim}{f}) - c_5(\underset{\sim}{f}) \qquad (5)$$

Similarly, player 2 seeks to maximize his payoff function, $P_2(\tilde{f})$, where

$$P_2(\tilde{f}) = -c_2(\tilde{f}) - c_4(\tilde{f}) \qquad (6)$$

Hence, the 2 players are in a competitive game situation since the costs sustained by each player are not only a function of their own actions, but also depend on the actions taken by the opposing player. This game was solved by means of the LAH-algorithm (see table 19) and the solution also found algebraically by Harris (details will not be given here as the algebraic method used to find the solution was the same as that given in 5.5).

6.2.3. Discussion of results

The three solutions to this problem, using the 3 different criteria of optimization, are displayed in table 19. The following points should be noted.

(i) System optimization provides the overall minimum total cost, $c(\tilde{f})$.

(ii) Game optimization provides a lower total cost, $c(\tilde{f})$, than user optimization. Although it has not been proved it is conjectured that this may have the following explanation. In the game theoretic optimization there are two conflicting players. However, the definition of user optimization, as given earlier, means that each of the individual users is in

TABLE 19

RESULTS FOR NETWORK OPTIMIZATION

Define the total cost $c(\underset{\sim}{f}) = \sum_{i=1}^{5} c_i(\underset{\sim}{f})$

	SYSTEM OPTIMIZED	GAME OPTIMIZED	USER OPTIMIZED
f_1	50.2	54.5	61.3
f_2	66.1	66.2	64.3
f_3	35.2	40.6	47.9
f_4	53.9	53.8	55.7
f_5	34.6	24.9	10.8
$c(\underset{\sim}{f})$	317,500	318,300	322,000

competition with every other user. This greater degree of competition in the user optimized system would tend to create a lesser degree of cooperation, and hence higher costs.

Thus, it has been shown that game theoretic methods can have application in other, not directly related, areas of applied mathematics, and the numerical techniques developed in this study (e.g. LAH algorithm) can be applied to such problems. In conclusion Harris, (16), has shown that the game theoretic optimization principle has important theoretical and practical applications in the theory of telephone networks. It is considered by Harris that the game theoretic principle can be used to clarify certain heuristic methods which are used by telephone network planning authorities. Also, the game theoretic solution may be used as a close approximation to the system optimized solution.

6.3. Business model

As has been observed in chapter 1, even though previous writers have remarked on the similarity between poker-like games and problems of business operations, (29), no explicit examples could be discovered in the literature. Thus, in this section, a business model is constructed to parallel the 4-PG solved in chapter 5. The broad principles on which the model is based are as follows.

Consider a situation involving 4 competing manufacturers denoted by 1,2,3 and 4. Each manufacturer must decide whether or not to market a particular product and thereby involve himself in a situation in which he will either make a profit or a loss depending on various assumptions, which are given below.

(a) The first assumption is that a highly specialized product that has a limited appeal is being marketed. The market has the property that it will ultimately buy from only one of the manufacturers. Three practical examples of such a situation are:

 (i) When the market consists of only one customer for example, a government department.

 (ii) When the market is so small that eventually only the most successful manufacturer will be able to operate in it. The remaining manufacturers will be forced to cut their losses and quit.

 (iii) When a monopoly situation can be established by one of the manufacturers.

(b) The second assumption is that when the winning manufacturer emerges, his total profit will equal the amount spent on advertising and promotion by the other manufacturers who have entered this competition. This assumption may be justified in the following way.

It is known that in certain situations, sales, and therefore profits, may be directly related to advertising expenditure. If the major non-returnable cost involved in this business is that of advertising, and if all the advertising is such that it benefits all manufacturers equivalently (i.e. develops new sections of the market which will eventually all be serviced by the final winner) then it is possible that profits will equal total amount spent on advertising. A historical example of this is given by Winkler, (36), and concerns the development of the tobacco industry in the U.S.A., around 1900, which was eventually dominated by one company. During certain periods of intense competition which preceded this, little or no profits were made, but huge markets were developed by all manufacturers as a result of the price cutting and heavy advertising which took place. Finally, a single company was successful in obtaining a monopoly of over 90% of all the tobacco trade, and eventually made enormous profits. It would not be unreasonable to assume that these profits were directly related to the huge sums spent in advertising and promotion by all manufacturers during the period of intense competition.

(c) Each manufacturer, before he decides whether or not to enter the market, needs to rate his product in some way. For the purposes of this model it will be assumed that this rating is given by some number x, $0 \leq x \leq 1$, where this value has the following properties.

(i) A clear and common understanding of this value exists among all players. That is, each manufacturer, if he had to value any one of the products, would give it the same value as all the other manufacturers. This would, in practice, arise because those who consistently over-valued or under-valued their products would go out of business, and only those who could give an accurate value, would remain.

For example pawn-brokers and used car salesmen must exercise this ability, while new car manufacturers must also learn to make accurate, consistent value judgements of this type.

(ii) By common agreement this value x has the following meaning. This is best illustrated by an example. Consider the case of a used car salesman who specialises in appraising a particular model of car. Because of his experience he is able to rate any given car, in relation to all other cars of that type, and state that this car is better than say p% of cars of this type. Then, in this case, define x to be equal to $p/100$.

(d) It is assumed that one of the four manufacturers is prepared to initiate the competitive situation in the following way. He postulates that there may exist a need for some particular product. Although he knows that he will be able to manufacture it he does not yet know just how good his product will be (i.e. he does not know his own value of x). However, he is prepared, at the cost of 1 unit, to mount a small advertising campaign, which will establish whether this particular product will in fact find a market. At this point the other three manufacturers are informed of the results of this survey, and thus the competitive situation is created.

(e) The other three manufacturers, knowing their own value of x, must now make a decision whether or not to enter the market. This will immediately involve them in a cost of 2 units for advertising. It is assumed that the competitors have certain fixed policies established over a period of time, that decisions are made in some definite pre-determined order. Conditions are such that all are aware of any decisions which are made.

(f) When all manufacturers have made known their intentions, the last manufacturer, now also knowing his value of x, has 3 choices.

(i) He may choose to drop out and forfeit his initial outlay of 1 unit.

(ii) He may decide to continue, but on equal terms with his rivals. Thus he increases his expenditure by 1 unit to match the total amounts outlayed by the others.

(iii) If his product happens to be particularly good, and because of his good strategic position (i.e. he was first into the market and has had the greatest amount of time to consider the situation, knowing the actions taken by the other players) he may increase his outlay by a further 3 units. This will introduce a tendency for his competitors to match his expenditure (at a cost of a further

2 units) as they are already fully committed and must continue. Not to match this increased expenditure would mean that they would certainly lose.

(g) The winning manufacturer is determined in the following way. Initially, each manufacturer computes the value of x applying to his own product. Denote these values by x_1, x_2, x_3, and x_4 respectively. At this stage none of the manufacturers know the x_i values calculated by any of their rivals. There is some fixed chance, q, equal for all competitors, of making a last minute technological breakthrough, which, is of such a magnitude that, if achieved, it will ensure that particular competitor of winning the market. However, if two or more manufacturers achieve a simultaneous breakthrough, then the winner is the one with the higher initial x_i. If no technological advances are made then the highest x_i will win.

Eq.1, chapter 4, defined the function $T_q(x)$ where, given some random number r, $0 \le r \le 1$, then

$$T_q(x) = \begin{cases} x \; ; & q \le r \le 1 \\ 2+x \; ; & 0 \le r \le q \end{cases}$$

Thus, using the above definition it follows that the winner may be defined as follows. Each value of x_i is replaced by $T_q(x_i)$, and then the highest value wins.

Now, using (a) to (g) above, it is possible to formulate a business operation which completely parallels the 4-PG. Familiarity with the 4-PG and points (a) to (g) above, will be assumed in the ensuing discussion.

6.3.1. Definition of business model

Suppose that four manufacturers denoted 1,2,3 and 4 are identified with players 1,2,3 and 4 in the 4-PG. In the ensuing discussion the 4-PG will be described and interpreted in terms of the business model.

First, player 4 makes an ante of 1 unit. This corresponds to manufacturer 4 initializing the competitive situation by outlaying 1 unit as described in (d) above. Next, the four players in the 4-PG are dealt hands x_1, x_2, x_3 and x_4. This corresponds to the four manufacturers calculating the goodness of their products in terms of the variables x_1, x_2, x_3 and x_4 as described in (c). Now, beginning with player 1 each player, in turn, decides whether or not to enter the game by making an ante of 2 units. This same situation occurs in the business operation (see (e)), with each manufacturer, in turn, deciding whether or not to enter the competition at a cost of 2 units.

Player 4 may now drop out, play on, or double, thereby forcing his competitors to double. This also occurs in the business model and is described in (f), where manufacturer 4 also has the option of dropping out (and forfeiting 1 unit), remaining in the market (at a total cost of 2 units), or doubling his expenditure (to a total cost of 4 units). At this stage of the 4-PG all hands x_i are improved by the transformation $T_q(x_i)$ as given in eq.1 chapter 4. Exactly the same procedure is used in the business model and this is described in (g). The winner is then identified in the same way in both cases. The 4-PG allows the winning player to take all bets made, while, the business model allows the winning manufacturer a profit equal to the sum totals of all amounts spent by his competitors, as described in (b).

Hence, the solution found for the 4-PG may be directly applied to the business model, and each of the four manufacturers can consider his value of x_i (goodness of product) to be equivalent to a hand x_i in the 4-PG, and take actions in the business situation corresponding to the ones that he would take in the 4-PG.

6.3.2. Discussion of business model

The discussion above showed how a business operation could be defined to parallel the 4-PG in such a way that the optimal strategy for the 4-PG could be applied to the business model. This suggests that other business models of this type may also be studied and solved by means of the theoretical methods developed in this study. The above example also indicates, in a practical way, the close connection between business and poker.

CHAPTER 7

SUMMARY, CONCLUSIONS AND DISCUSSION OF NEW WORK

7.1 Summary

This section will summarise and discuss the work carried out in this study. The initial investigation of the problem of solving poker-like games, revealed the following points.

(a) A review of the literature showed that poker was amongst the most difficult of card games, and that a significant amount of work had been carried out in this area by Bellman, Friedman, Karlin, von Neumann, Restreppo and others. However, the games solved could not be applied to any commonly played variety of poker. This was because the problems posed by poker-like games were sufficiently formidable to restrict the workers noted above to relatively simple analyses.

(b) Apart from the work of Findler, no treatment of poker simulation was noted in the literature. This was surprising because:-

(i) Simulation of poker offers a worker the capability of evaluating optimal strategies and of testing the validity of theoretical results.

(ii) A poker simulator could be adapted to play interactively with human opponents.

Thus simulation appeared to offer attractive possibilities of giving further insight into the theory of the game of poker.

(c) A review of the literature showed that the main difficulty in applying game theoretic methods to poker arose in the solution of the resultant equations. It followed that the application of numerical methods might yield useful results. However, this approach had not previously been used for poker, and no suitable algorithms were found to exist. It was probable therefore, that the numerical approach might require much new work.

(d) Finally, despite comments by various writers (19,29) on the connection between poker and business operations, no application of poker results to a practical problem, could be found. This suggested that an important sequel to a theoretical study of poker should be an application of theoretical results to relevant problems.

The foregoing considerations led to the work of this thesis being carried out in four main parts as follows.

7.2 Approach to the problem posed in this thesis

(a) Simulation of poker

The first part of this work, see chapter 2, dealt with the simulation of poker. During this research the following results were obtained.

(i) probabilities of winning with poker hands were defined and calculated

(ii) values were obtained for computation times for the simulation of poker

(iii) a method of determining optimal strategies using simulation was presented but was shown to be impractical, as too much computer time was required

(iv) the simulation program was modified to play poker interactively. However, this part of the work could not be carried through to a satisfactory conclusion because:

1. sufficiently long poker sessions between the machine and human players could not be arranged
2. poker needs to be played for money if the results are to have practical significance, but this could not be arranged in the circumstances of the work of this thesis.

As a result of the work done it was concluded that simulation offered a possible method of solution but there were considerable computational difficulties involved. Accordingly, a preliminary study was made of a game-theoretic approach and this appeared to offer good prospects of achieving a solution more easily than simulation. As a result the simulation studies were discontinued.

(b) The game theoretic approach for solving games and numerical methods

The game-theoretic approach to the study of poker-like games required the solution of certain complicated equations. Research showed that these equations could be most conveniently solved using numerical methods, but a survey of the literature did not reveal any applications of such methods to poker-like games. Hence, much work was carried out in this area, ab initio. This work is described in chapter 3 and forms an important part of this study.

The main results of this work are as follows.

(i) Two numerical algorithms capable of solving poker-like games were programmed. The first of these was a modification of Rosen's method (31). However it was demonstrated that this algorithm was too slow for the purposes of this work.

(ii) Consequently a new iterative method called the lookahead (LAH) algorithm was formulated. This operated on a lookahead principle (as described in chapter 3) and proved to be three times faster than Rosen's modified method.

(c) Solution of realistic poker-like games

Chapters 4 and 5 describe the work carried out in applying game theoretic methods to a commonly played variety of poker. This research was structured as follows.

(i) Since the game considered was too complicated to treat in its entirety, the rules were carefully simplified, in such a way that much of the essential character of the game remained unaltered.

(ii) In the course of deriving payoff functions for this game a multi-dimensional integral, which commonly arises in games of this type, was encountered. Its evaluation for the n-person case presented considerable difficulties and merited treatment in a separate appendix (Appendix A).

This result played an important part in the method used for solving the 3 and 4 person poker-like games, and could be extended to poker-like games for 5, 6 and 7 players.

(iii) Next, the lookahead algorithm was applied to the payoff functions and the game solved. This solution had several new features.

1. Previous research in this area had only considered poker-like games with no more than three players. They were not sufficiently realistic to have any broad practical application to real poker with more than two players. In this study, by using the results obtained during the work on poker simulation, it was possible to relate the solutions obtained to a real game of poker. It was then found that the theoretical optimal strategies agreed with the strategies recommended by experienced players, although there were some areas of difference such as minimum hand required by the second to last player. (30). For example, minimum hand requirements, when first to enter, agreed closely with the work of Reece and Watkins, (30). Also the disadvantage of playing last was demonstrated analytically, and it was confirmed

that players closest to the last player had the highest expectations.

2. Computer running times required to determine the solution were recorded, and it was noted that, using present methods, too much computer time would be required, if more than 4 person games were to be solved.

(d) Practical applications of this work

Two applications of this study to practical problems are presented in chapter 6. The first application describes a game theoretic approach to a problem in networks. This work defines the network to be a game between 2 or more players. This game can then be solved by the methods given in this study. The results obtained by this approach have proved useful in the analysis of telephone networks, as they define a new criterion of optimality, (16).

The second application defines a business operation involving four manufacturers, who must decide whether or not to market a new product. This business operation is defined to parallel the 4-person poker-like game. Thus an exact criterion of optimality that can be applied by each manufacturer when deciding whether or not to enter the market follows from the solution to the poker-like game.

Furthermore, this solution also indicates to each manufacturer the amount of his expected profit.

7.3 Further areas of research

(a) Simulation

As a result of the work carried out on the simulation of poker, it appears that further research could be directed in the following areas.

(i) In chapter 2 a theoretical method is described by which simulation could be used to determine optimal poker strategies. But experimental work showed that this approach was computationally too slow. A new approach to this problem would be to study the logical processes by which an experienced poker player analyses the game, and is able to adapt his strategy to give himself the best results. Perhaps, as a result of such work, fewer games would need to be simulated to determine optimal strategies. Thus the total amount of computation would be reduced and it would then be feasible to carry out these calculations, on a computer, in a reasonable time.

(ii) Ultimately, the success or failure of any poker playing program can only be measured by play against human opponents. This can best be carried out by using interactive poker programs under realistic playing conditions. Money should be used in such experiments, and the poker sessions should extend over many hours. Results obtained from such experiments could be analysed to give further information about poker, to determine weaknesses in the program and to indicate ways in which the program logic might be improved.

(b) Numerical methods of solving games

Since it would be of great practical importance to solve more complicated games, new research in the area of numerical methods of solution should be carried out. One point of departure would be to carry out further work on the lookahead algorithm. For example, the various combined effects of altering the degree of lookahead, the ratio between successive step sizes and various other parameters of the algorithm could be studied. In addition, completely new approaches to this problem, could be attempted. A possible source of new ideas lies in the analysis of the human, intuitive, approach to the solution of games, by which gifted managers achieve favourable outcomes in situations of great complexity, (35).

(c) Solution of more complicated games

It would be feasible to solve a 7-person poker-like game, by game-theoretic methods. There are, however, two obstacles.

The first difficulty is the large number of strategy variables (many hundreds), which would arise in the analysis of this game. New work may discover means of representing the strategies using a smaller number of variables.

The second difficulty arises in solving these games. Obviously, improved numerical methods, will be of assistance. But, another approach might be to devise a technique which uses the algebraic method of solution, solved on the computer, by the methods of symbolic manipulation.

(d) Applied games theory

Game theory can be applied to a wide circle of problems. Until now, as a consequence of the difficulty of finding solutions, work in this area has met with little practical success. However, work on applications should continue, because any progress would be of practical use, as has been particularly demonstrated in chapter 6.

In particular, the new work on the game-theoretic approach to network problems has yielded useful results and work in this area is being continued by another researcher.

APPENDIX A : EVALUATION OF PAYOFF FUNCTION INTEGRAL

A.1. Introduction

It was found that, in the course of this study (see chapters 3,4,5), a certain integral arose repeatedly. An intensive search of the literature revealed no publication dealing with this or any other related integral. Accordingly, a method for the evaluation of this integral was developed, and is presented in this appendix. This integral is defined in the following way.

Let

$$\chi_n^{w,\ell}(x_n,\ldots,x_1) = \begin{cases} w; & \text{if } x_n \geq x_i \text{ for all } i=1,\ldots n \\ \ell; & \text{if } x_n < x_i \text{ for any } i\neq n \end{cases} \qquad (1)$$

then the required integral is

$$X_n^{w,\ell}\begin{bmatrix} b_n & \ldots & b_1 \\ a_n & \ldots & a_1 \end{bmatrix} = \int_{x_n=a_n}^{b_n} \cdots \int_{x_1=a_1}^{b_1} \chi_n^{w,\ell}(x_n,\ldots x_1)\,dx_1\ldots dx_n \qquad (2)$$

where $a_i \leq b_i$ for $i=1,\ldots n$.

This integral will be evaluated in the following way. First, the case $n=2$ will be considered. It will be shown that there is a relatively straight forward, graphical method which can evaluate this integral, and which, in theory, could be generalised. However, it will then be proven that it is not practically possible to extend this method to the general case (any value of n).

Accordingly this integral (for n=2) is evaluated by using a more difficult and less obvious technique, which can, however, with some hardship, be generalised to give the result for arbitrary values of n. The validity of the expression found is then checked by comparison with the results from a monte-carlo method, which is itself, however, shown to be too slow and inaccurate to be used here.

A.2. Calculation of integral for n=2

The integral will be calculated, for the case n=2, by two methods. First, by a graphical method, and second, by an algebraic method.

A.2.1. Calculation of integral for n=2 by a graphical method

If n=2 then eqs. 1 and 2 take the form

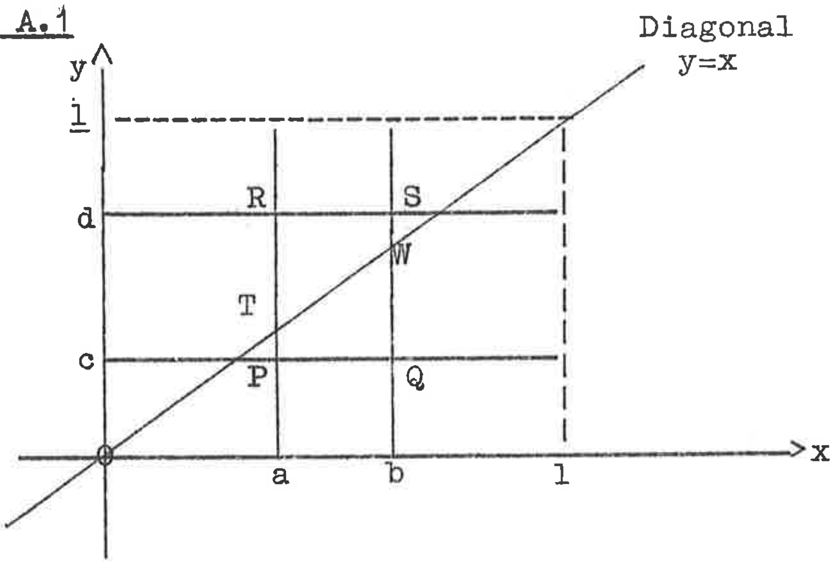
$$\chi_2^{w,\ell}(x,y) = \begin{cases} w; & \text{if } x \geq y \\ \ell; & \text{if } x < y \end{cases} \qquad (3)$$

and

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = \int_{x=a}^{b} \int_{y=c}^{d} \chi_2^{w,\ell}(x,y)\,dx\,dy \qquad (4)$$

Consider the case $c \leq a \leq d$ and $c \leq b \leq d$, then figure A.1 below illustrates the domain of integration.

FIGURE A.1



The double integral is taken over the quadrilateral PQSR. However, in that part of PQSR which is below the line TW, $x \geqslant y$, and hence, from eq.3, $\chi_2^{w,\ell}(x,y)$ is constant over this region, and equals w. Similarly, $\chi_2^{w,\ell}(x,y)$ is constant over the region inside TWSR and equals ℓ. Accordingly,

$$\int_{x=a}^{b} \int_{y=c}^{d} \chi_2^{w,\ell}(x,y)\,dxdy$$

$$= \iint_{\substack{\text{area inside} \\ \text{PQSR}}} \chi_2^{w,\ell}(x,y)\,dxdy$$

$$= \iint_{\substack{\text{area inside} \\ \text{PQWT}}} \chi_2^{w,\ell}(x,y)\,dxdy + \iint_{\substack{\text{area inside} \\ \text{TWSR}}} \chi_2^{w,\ell}(x,y)\,dxdy$$

$$= \iint_{\substack{\text{area inside} \\ \text{PQWT}}} w\,dxdy + \iint_{\substack{\text{area inside} \\ \text{TWSR}}} \ell\,dxdy$$

$= w$ (area inside PQWT) $+ \ell$ (area inside TWSR).

By the geometry of figure A.1 it may be seen that:-

area inside PQWT $= \frac{1}{2}(b-a)^2 + (b-a)(a-c)$ (5)

area inside TWSR $= \frac{1}{2}(b-a)^2 + (b-a)(d-b)$ (6)

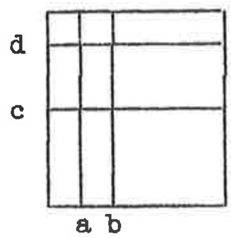
Thus

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = w[\tfrac{1}{2}(b-a)^2+(b-a)(a-c)]+\ell[\tfrac{1}{2}(b-a)^2+(b-a)(d-b)]$$

There are, in all, 6 possibilities of the above type, and these are enumerated below along with the corresponding values of

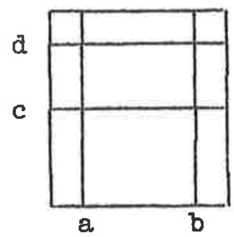
$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix}.$$

1. $0 \leq a \leq c$; $0 \leq b \leq c$;



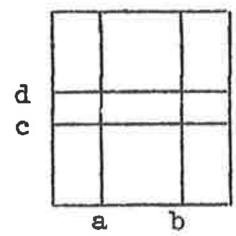
$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = \ell(b-a)(d-c)$$

2. $0 \leq a \leq c$; $c \leq b \leq d$;



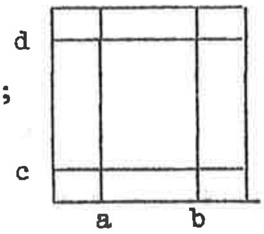
$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = w.\tfrac{1}{2}(b-a)^2+\ell[(b-a)(d-c) - \tfrac{1}{2}(b-a)^2]$$

3. $0 \leq a \leq c$; $d \leq b \leq 1$;



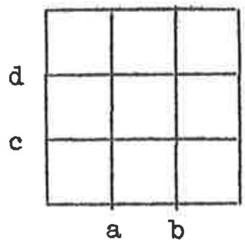
$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = w[\tfrac{1}{2}(d-c)^2+(d-c)(b-d)] + \ell[\tfrac{1}{2}(d-c)^2+(d-c)(c-a)]$$

4. $c \le a \le d;\ c \le b \le d;$



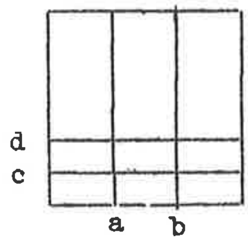
$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = w[\tfrac{1}{2}(b-a)^2+(b-a)(a-c)] + \ell[\tfrac{1}{2}(b-a)^2+(b-a)(d-b)]$$

5. $c \le a \le d;\ d \le b \le 1$



$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = \ell[\tfrac{1}{2}(d-a)^2] + w[(b-a)(d-c)-\tfrac{1}{2}(d-a)^2]$$

6. $d \le a \le 1;\ d \le b \le 1$



$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = w(b-a)(d-c)$$

The above results may now be employed to evaluate this integral, and a fortran function was written to do this. However, if this method of evaluation is applied to the integral with $n=3$ then it may be seen that 90 cases arise, and it would be extremely difficult to tabulate all of these, let alone attempt to apply this method to the case when $n=4$ or more. Hence, as integrals of dimension 3 and higher must be evaluated later, an alternative method is developed in the next section.

A.2.2. Calculation of integral for n=2 by an algebraic method

An algebraic method for evaluating $X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix}$ will now be given. This method is based on the idea of expressing the integral as the sum of 4 separate sub-integrals, each of which may be simply evaluated.

First, write

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = \int_{x=a}^{b} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy + \int_{x=a}^{b} \int_{y=x}^{d} \chi_2^{w,\ell}(x,y)\,dx\,dy \qquad (7).$$

Now define

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = \int_{x=a}^{b} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy \qquad (8).$$

Thus, by eq.8

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & d \end{bmatrix} = \int_{x=a}^{b} \int_{y=d}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy$$

and hence

$$-J_1^{w,\ell}\begin{bmatrix} b & x \\ a & d \end{bmatrix} = \int_{x=a}^{b} \int_{y=x}^{d} \chi_2^{w,\ell}(x,y)\,dx\,dy \qquad (9)$$

Substituting eqs.8 and 9 into eq.7

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} - J_1^{w,\ell}\begin{bmatrix} b & x \\ a & d \end{bmatrix} \qquad (10)$$

In order to evaluate J_1 define

$$g = \max(\min(b,c),a) \tag{11}$$

As will become apparent later this particular choice of g has been made because, since $a \leq b$ and $c \leq d$ (see eq.2) it follows that:-

(i) if $c \in [a,b]$ then $g=c$

(ii) if $c < a$ then $g=a$

(iii) if $c > b$ then $g=b$

Next expand eq.8 by introducing g into the limits of integration

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = \int_{x=a}^{b} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy$$

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = \int_{x=a}^{g} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dxdy + \int_{x=g}^{b} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dxdy \tag{12}$$

and let

$$K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} = \int_{x=a}^{g} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy$$

and

$$K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix} = \int_{x=g}^{b} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)\,dx\,dy$$

then eq.12 becomes

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} + K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix} \tag{13}$$

First consider $K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix}$

It will be shown that

$$K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} = \ell(g-a)\left[\frac{g+a}{x} - c\right] \qquad (14)$$

for the 2 possible cases $c \geq a$ and $c < a$.

(a) $c \geq a$

In this case it follows from eq.11 that $g = \min(b,c)$, and thus $c \geq g$. Hence as $y \in [c,x]$ and $x \in [a,g]$ then $x<y$, and so $\chi_2^{w,\ell}(x,y) = \ell$ from eq. 6.

Now

$$K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} = \int_{x=a}^{g} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)dx\,dy$$

$$= \int_{x=a}^{g} \int_{y=c}^{x} \ell \; dxdy$$

$$= \ell \left[\frac{x^2}{2} - cx\right]_a^g$$

$$= \ell(g-a)\left[\frac{g+a}{2} - c\right]$$

and hence eq.14 holds.

(b) $c<a$

Again from eq.11, $c<a$ implies that $g=a$.

Hence $K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} = \int_{x=a}^{a} \int_{y=c}^{x} \chi_2^{w,\ell}(x,y)dx\,dy$

$= 0$ as the variable x has zero range.

Thus, again eq.14 holds.

Hence eq.14 always holds.

Next, using the same method it will be shown that

$$K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix} = w(b-g)\left[\frac{b+g}{2} - c\right]. \qquad (15)$$

Again consider the 2 possible cases $c \leqslant b$ and $c > b$.

(a) $c \leqslant b$

From eq.11, $g = \max(a,c)$, thus $g \geqslant c$.

Hence as $y \in [c,x]$ and $x \in [g,b]$ then $x \geqslant y$, and so $\chi_2^{w,\ell}(x,y) = w$ from eq. 4.

Thus, as before, $K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix}$ may be calculated,

and it is found that

$$K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix} = w(b-g)\left[\frac{b+g}{2} - c\right]$$

(b) $c > b$

The same argument as was used earlier may be employed to show that $g = b$ and that consequently eq.15 holds.

Hence eq.15 holds for all cases.

From eq.13

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = K_1^{w,\ell}\begin{bmatrix} g & x \\ a & c \end{bmatrix} + K_2^{w,\ell}\begin{bmatrix} b & x \\ g & c \end{bmatrix}$$

where from eq.11, $g = \max(a,\min(b,c))$ and using eqs.14 and 15

$$J_1^{w,\ell}\begin{bmatrix} b & x \\ a & c \end{bmatrix} = \ell(g-a)\left[\frac{g+a}{2} - c\right] + w(b-g)\left[\frac{b+g}{2} - c\right] \tag{16}$$

By a similar means it may be shown that

$$\text{if} \quad h = \max(a,\min(b,d)) \tag{17}$$

$$\text{then } J_1^{w,\ell}\begin{bmatrix} b & x \\ a & d \end{bmatrix} = \ell(h-a)\left[\frac{h+a}{2} - c\right] + w(b-h)\left[\frac{b+h}{2} - c\right]. \tag{18}$$

By substituting eqs.16 and 18 into eq.10 it is found that

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix} = \ell(g-a)\left[\frac{g+a}{2} - c\right] + w(b-g)\left[\frac{b+g}{2} - c\right]$$

$$- \ell(h-a)\left[\frac{h+a}{2} - c\right] - w(b-h)\left[\frac{b+h}{2} - c\right] \tag{19}$$

where g and h are defined by eqs.11 and 17 respectively.

Thus, the integral

$$X_2^{w,\ell}\begin{bmatrix} b & d \\ a & c \end{bmatrix}$$

is given by eq.19.

A.3 Generalisation of the algebraic method

The method used above for the case $n=2$, may be extended, with some difficulty, to the general n-dimensional case. The general approach is to express the integral as the sum of several integrals, each of which may be individually evaluated. This is carried out in several stages.

First, the integral will be split into 2^{n-1} subintegrals, in an analogous manner to the case for $n=2$. Secondly, a set S will be defined, which may be used to express more conveniently these 2^{n-1} subintegrals. Finally, each of these subintegrals will be individually evaluated to give the final result.

In what follows it will be convenient to denote $\chi_n^{w,\ell}(x_n,\ldots x_1)dx_{n-1}\ldots dx_1$ by $**$.

From eq.2

$$X_n^{w,\ell}\begin{bmatrix} b_n & b_1 \\ a_n\ldots & a_1 \end{bmatrix} = \int_{x_n=a_n}^{b_n}\cdots\int_{x_1=a_1}^{b_1}\chi_n^{w,\ell}(x_n,\ldots x_1)dx_n\ldots dx.$$

Hence expanding this integral

$$X_n^{w,\ell}\begin{bmatrix} b_n & b_1 \\ a_n\ldots & a_1 \end{bmatrix} = \int_{x_n=a_n}^{b_n}\left\{\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_1=a_1}^{x_n} ** + \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_1=x_n}^{b_1} **\right\} dx_n$$

$$= \int_{x_n=a_n}^{b_n}\left\{\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_1=a_1}^{x_n} ** - \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_1=b_1}^{x_n} **\right\} dx_n \qquad (20)$$

Thus eq. 20 may be further expanded to give

$$X_n^{w,\ell}\begin{bmatrix} b_n & b_1 \\ a_n\ldots & a_1 \end{bmatrix} = \int_{x=a_n}^{b_n}\left\{\left[\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=a_2}^{x_n}\int_{x_1=a_1}^{x_n} ** + \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=x_n}^{b_2}\int_{x_1=a_1}^{x_n} **\right]\right.$$

$$-\left[\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=a_2}^{x_n}\int_{x_1=b_1}^{x_n}** + \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=x_n}^{b_2}\int_{x_1=b_1}^{x_n}**\right]\Bigg\} dx_n$$

$$= \int_{x=a_n}^{b_n}\Bigg\{\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\int_{x_1=a_1}^{x_n}** - \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=b_2}^{x_n}\int_{x_1=a_1}^{x_n}**$$

$$-\int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=a_2}^{x_n}\int_{x_1=b_1}^{x_n}** + \int_{x_{n-1}=a_{n-1}}^{b_{n-1}}\cdots\int_{x_2=b_2}^{x_n}\int_{x_1=b_1}^{x_n}**\Bigg\} dx_n$$

By repeating the above process the final result is

$$X_n^{w,\ell}\begin{bmatrix} b_n & & b_1 \\ a_n & \cdots & a_1\end{bmatrix} = \int_{x_n=a_n}^{b_n}\Bigg\{\int_{x_{n-1}=a_{n-1}}^{x_n}\cdots\int_{x_1=a_1}^{x_n}** - \int_{x_{n-1}=a_{n-1}}^{x_n}\cdots\int_{x_2=a_2}^{x_n}\int_{x_1=b_1}^{x_n}**$$

$$-\int_{x_{n-1}=a_{n-1}}^{x_n}\cdots\int_{x_2=b_2}^{x_n}\int_{x_1=a_1}^{x_n}** + \int_{x_{n-1}=a_{n-1}}^{x_n}\cdots\int_{x_2=b_2}^{x_n}\int_{x_1=b_1}^{x_n}** + \cdots$$

$$+(-1)^{n-1}\int_{x_{n-1}=b_{n-1}}^{x_n}\cdots\int_{x_1=b_1}^{x_n}**\Bigg\} dx_n \qquad (21)$$

Eq.21 may be expressed in a more convenient form, but to do this some new notation must be introduced.

Define the set

$S = \{c_{n-1}, \ldots c_1\}$ where $c_j = a_j$ or b_j. (22)

Let the set J be the set of all possible S, (note that there are 2^{n-1} different S which may be formed as there are 2 possibilities for each element c_j, and there are $n-1$ elements in S).

For any $S \in J$ define a function

$$p(S) = \begin{cases} +1 \text{ ; if the no. of } b_j \text{ elements in } S \text{ is even} \\ -1 \text{ ; if the no. of } b_j \text{ elements in } S \text{ is odd} \end{cases} \tag{23}$$

Define $Y_{n-1}^{w,\ell}(c_{n-1}, \ldots c_1, x_n) =$

$$\int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} \chi_n^{w,\ell}(x_n, \ldots x_1)\, dx_1 \ldots dx_n$$

Then by inspecting eq.21 it may be seen that this equation can be written

$$X_n^{w,\ell}\begin{bmatrix} b_n & & b_1 \\ a_n & \ldots & a_1 \end{bmatrix} = \int_{x=a_n}^{b_n} \left[\sum_{S \in J} p(S) Y_{n-1}^{w,\ell}(c_{n-1}, \ldots c_1, x_n) \right] \tag{24}$$

Now define

$$Z_n^{w,\ell}\begin{bmatrix} b_n & & \\ a_n & c_{n-1} \ldots c_1 \end{bmatrix} = \int_{x=a_n}^{b_n} Y_{n-1}^{w,\ell}(c_{n-1}, \ldots c_1, x_n)\, dx_n \tag{25}$$

then eq.24 becomes

$$X_n^{w,\ell}\begin{bmatrix} b_n & & b_1 \\ a_n & \cdots & a_1 \end{bmatrix} = \sum_{S\in J} p(S) Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\cdots c_1 \end{bmatrix} \qquad (26)$$

In evaluating

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\cdots c_1 \end{bmatrix}$$

the order of evaluating with respect to $c_{n-1},\ldots c_1$, is immaterial, hence there is no loss in generality in assuming that $c_{n-1} \geq \max\{c_{n-2},\ldots c_1\}$. (27)

The following lemma will be used to evaluate

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\cdots c_1 \end{bmatrix}.$$

<u>Lemma</u>

Define

$$d = \min(\max(a_n, c_{n-1}), b_n) \qquad (28)$$

and

$$W_n\begin{bmatrix} b & \\ a & c_{n-1}\cdots c_1 \end{bmatrix} = \int_{x=a}^{b} \prod_{i=1}^{n} (x-c_i)\,dx \qquad (29)$$

Then

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\cdots c_1 \end{bmatrix} = \ell . W_n\begin{bmatrix} d & \\ a_n & c_{n-1}\cdots c_1 \end{bmatrix} +$$

$$+\ w . W_n\begin{bmatrix} b_n & \\ d & c_{n-1}\cdots c_1 \end{bmatrix} \qquad (30)$$

Proof

Consider the following 3 cases.

(a) $c_{n-1} \leq a_n$

From the definition of d it follows that $d = a_n$. As $a_n \geq c_{n-1}$, and $c_{n-1} = \max\{c_{n-1}, \ldots c_1\}$ (from eq.27) it follows that if $x_n \in [a_n, b_n]$ and

$$x_{n-1} \in [c_{n-1}, x_n], \ldots x_1 \in [c_1, x_n]$$

then $x_n \geq \max\{x_{n-1}, \ldots x_1\}$

and hence $\chi_n^{w,\ell}(x_n, \ldots x_1) = w$ from eq.1. (31)

Hence from eq.25

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1} \ldots c_1 \end{bmatrix} =$$

$$\int_{x_n=d}^{b_n} \int_{x_{n-1}=c_{n-1}}^{x} \ldots \int_{x_1=c_1}^{x} \chi_n^{w,\ell}(x_n, \ldots x_1)\, dx_n \ldots dx_1$$

$$= \int_{x_n=d}^{b_n} \int_{x_{n-1}=c_{n-1}}^{x_n} \ldots \int_{x_1=c_1}^{x_n} w\, dx_n \ldots dx_1 \quad \text{from eq.31.}$$

$$= w \int_{x=d}^{b_n} \prod_{i=1}^{n-1} (x - c_i)\, dx$$

$$= w\, W_n\begin{bmatrix} b_n & \\ d & c_{n-1} \ldots c_1 \end{bmatrix}.$$

And as $W_n\begin{bmatrix} d \\ a_n \; c_{n-1}\ldots c_1 \end{bmatrix} = 0$ because $a_n = d$ it follows that eq.29 holds in this case.

(b) $c_{n-1} \in [a_n, b_n]$ (32)

From the definition of d,

$d = \min(\max(a_n, c_{n-1}), b_n)$ and using eq.32

it follows that $d = \min(c_{n-1}, b_n) = c_{n-1}$. (33)

When $x_n \in [a_n, c_{n-1}]$ and $x_{n-1} \in [c_{n-1}, x_n]$,

then $x_{n-1} \geqslant x_n$, and thus from eq.1

$$\chi_n^{w,\ell}(x_n, \ldots x_1) = \ell \; . \tag{34}$$

Similarly, when $x_n \in [c_{n-1}, b_n]$ and $x_{n-1} \in [c_{n-1} x_n]$, $x_{n-2} \in [c_{n-2}, x_n], \ldots x_1 \in [c_1, x_n]$, since in eq.27 c_{n-1} has been defined in such a way that $c_{n-1} \geqslant \max c_{n-2}, \ldots c_1$ it follows that

$$x_n \geqslant x_{n-1}, \; x_n \geqslant x_{n-2}, \ldots x_n \geqslant x_1$$

and thus $\chi_n^{w,\ell}(x_n, \ldots x_1) = w$ from eq.1. (35)

Hence from eq.25

$$Z_n^{w,\ell}\begin{bmatrix} b_n \\ a_n \; c_{n-1}\ldots c_1 \end{bmatrix} =$$

$$\int_{x_n=a_n}^{b_n} \int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} \chi_n^{w,\ell}(x_n, \ldots x_1)\, dx_n \ldots dx_1$$

and expanding this integral

$$= \int_{x_n=a_n}^{d} \int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} \chi_n^{w,\ell}(x_n, \ldots x_1)\, dx_n \ldots dx_1$$

$$+ \int_{x_n=d}^{b_n} \int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} \chi_n^{w,\ell}(x_n,\dots x_1)dx\dots dx_1$$

Now employing the fact that $d = c_{n-1}$ and applying eqs.34 and 35 this integral may be written

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\dots c_1 \end{bmatrix} = \int_{x_n=a_n}^{d} \int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} \ell dx_n\dots dx_1$$

$$+ \int_{x_n=d}^{b_n} \int_{x_{n-1}=c_{n-1}}^{x_n} \cdots \int_{x_1=c_1}^{x_n} w dx_n\dots dx_1$$

$$= \ell \int_{x_n=a_n}^{d} \prod_{i=1}^{n-1} (x_n-c_i)dx_n + w \int_{x_n=d}^{b_n} \prod_{i=1}^{n-1} (x_n-c_i)dx_n$$

$$= \ell\ W_n\begin{bmatrix} d & \\ a_n & c_{n-1}\dots c_1 \end{bmatrix} + w\ W_n\begin{bmatrix} b_n & \\ d & c_{n-1}\dots c_1 \end{bmatrix}$$

Hence eq.29 holds.

(c) $\underline{c_{n-1} \geqslant b_n}$

This case is rather similar to (a).

First $d = \min(\max(a_n,c_{n-1}),b_n) = \min(c_{n-1},b_n) = b_n$.

Now if $x_n \in [a_n,b_n]$ and $x_{n-1} \in [c_{n-1},x_n]$,

$$x_{n-2} \in [c_{n-2},x_n],\dots x_1 \in [c_1,x_n],$$

then since $c_{n-1} \geqslant \max\{c_{n-2},\dots c_1\}$, it follows that $\chi_n^{w,\ell}(x_n,\ldots x_1) = \ell.$

Hence from eq.25, carrying out the calculation in the same way as before it can be shown that

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\ldots c \end{bmatrix} = \ell\ W_n\begin{bmatrix} d & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix}$$

Furthermore $b_n = d_n$ implies that $W_n\begin{bmatrix} b_n & \\ d & c_{n-1}\ldots c_1 \end{bmatrix} = 0$

thus

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix} = \ell W_n\begin{bmatrix} d & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix} + $$

$$+\ wW_n\begin{bmatrix} b_n & \\ d & c_{n-1}\ldots c_1 \end{bmatrix}$$

and eq.30 holds.

This completes the proof of the lemma.

Now eqs.28 - 30 may be used in conjunction with eqs. 22,23 and 26 to give:

$$X_n^{w,\ell}\begin{bmatrix} b_n & b_1 \\ a_n\ldots & a_1 \end{bmatrix} = \sum_{S\in J} p(S) Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix} \qquad (36)$$

where

$$Z_n^{w,\ell}\begin{bmatrix} b_n & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix} = \ell W_n\begin{bmatrix} d & \\ a_n & c_{n-1}\ldots c_1 \end{bmatrix}$$

$$+\ wW_n\begin{bmatrix} b_n & \\ d & c_{n-1}\ldots c_1 \end{bmatrix}$$

$$d = \min(\max(a_n, c_{n-1}), b_n)$$

$$W_n\left[\begin{matrix} b & \\ a & c_{n-1}\ldots c_1\end{matrix}\right] = \int_{x=a}^{b} \prod_{i=1}^{n-1}(x-c_i)dx$$

and S,J,p(S) have been defined earlier.

Example

Eq.36 may be used to find $X_3^{w,\ell}\left[\begin{matrix} b_3 & b_2 & b_1 \\ a_3 & a_2 & a_1\end{matrix}\right]$.

Using eq.22 let

$$S_1 = \{a_2, a_1\}, \quad S_2 = \{a_2, b_1\}, \quad S_3 = \{b_2, a_1\},$$
$$S_4 = \{b_2, b_1\},$$

then by the definition given earlier $J = \{S_1, S_2, S_3, S_4\}$,

and using eq.23

$$p(S_1) = 1,\ p(S_2) = -1,\ p(S_3) = 1,\ p(S_4) = -1.$$

Also from eq.29

$$W_3\left[\begin{matrix} b & & \\ a & c_2 & c_1\end{matrix}\right] = \int_{x=a}^{b} \prod_{i=1}^{2}(x-c_i)dx$$

$$= \int_{x=a}^{b} (x^2-(c_1+c_2)x+c_1c_2)dx$$

$$= \tfrac{1}{3}(b^3-a^3)-\tfrac{1}{2}(c_1+c_2)(b^2-a^2)+c_1c_2(b-a)$$

$$= (b-a)\left[\frac{a^2+ab+b^2}{3} - \frac{(c_1+c_2)(a+b)}{2} + c_1c_2\right].$$

Thus from eq.30

$$Z_3^{w,\ell}\begin{bmatrix} b_3 & & \\ a_3 & c_2 & c_1 \end{bmatrix} = \ell W_3\begin{bmatrix} d & & \\ a_3 & c_2 & c_1 \end{bmatrix} + wW_3\begin{bmatrix} b_n & & \\ d & c_2 & c_1 \end{bmatrix}$$

where from eq.28 $d = \min(\max(a_3,c_2),b_3)$.

Hence substituting into eq.36

$$X_3^{w,\ell}\begin{bmatrix} b_3 & b_2 & b_1 \\ a_3 & a_2 & a_1 \end{bmatrix} = Z_3^{w,\ell}\begin{bmatrix} b_3 & & \\ a_3 & a_2 & a_1 \end{bmatrix} - Z_3^{w,\ell}\begin{bmatrix} b_3 & & \\ a_3 & a_2 & b_1 \end{bmatrix} \tag{37}$$

$$- Z_3^{w,\ell}\begin{bmatrix} b_3 & & \\ a_3 & b_2 & a_1 \end{bmatrix} + Z_3^{w,\ell}\begin{bmatrix} b_3 & & \\ a_3 & b_2 & b_1 \end{bmatrix}$$

A fortran subroutine was programmed to evaluate this function.

The integral $X_4^{w,\ell}\begin{bmatrix} b_4 & b_1 \\ a_4 \ldots & a_1 \end{bmatrix}$ may be evaluated in the same way.

From eq.36

$$X_4^{w,\ell}\begin{bmatrix} b_4 & b_1 \\ a_4 \ldots & a_1 \end{bmatrix} = Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & a_3 & a_2 & a_1 \end{bmatrix} - Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & a_3 & a_2 & b_1 \end{bmatrix}$$

$$- Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & a_3 & b_2 & a_1 \end{bmatrix} + Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & a_3 & b_2 & b_1 \end{bmatrix}$$

$$- Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & b_3 & a_2 & a_1 \end{bmatrix} + Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & b_3 & a_2 & b_1 \end{bmatrix}$$

$$+ Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & b_3 & b_2 & a_1 \end{bmatrix} - Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & b_3 & b_2 & b_1 \end{bmatrix}$$

Now from eq.29

$$W_4\begin{bmatrix} b & & & \\ a & c_3 & c_2 & c_1 \end{bmatrix} = \int_{x=a}^{b} \prod_{i=1}^{3} (x-c_i)\,dx$$

$$= \frac{b^4-a^4}{4} + \frac{(c_1+c_2+c_3)(b^3-a^3)}{3}$$

$$+ \frac{(c_1c_2+c_1c_3+c_2c_3)(b^2-a^2)}{2} - c_1c_2c_3(b-a) \qquad (38)$$

Thus using eqs.30 and 38,

$$Z_4^{w,\ell}\begin{bmatrix} b_4 & & & \\ a_4 & c_3 & c_2 & c_1 \end{bmatrix} = \ell W_4\begin{bmatrix} d & & & \\ a_4 & c_3 & c_2 & c_1 \end{bmatrix} + wW_4\begin{bmatrix} b_4 & & & \\ d & c_3 & c_2 & c_1 \end{bmatrix}$$

with $d = \min(\max(a_3, c_2), b_3)$.

A fortran function was written to evaluate this integral.

A.4. Use of monte-carlo method to check validity of the algebraic expression

A standard monte-carlo method has been used to show that the analytic expressions obtained for

$$X_3^{w,\ell}\begin{bmatrix} b_3 & b_2 & b_1 \\ a_3 & a_2 & a_1 \end{bmatrix} \quad \text{and} \quad X_4^{w,\ell}\begin{bmatrix} b_4 & & b_1 \\ a_4 & \ldots & a_1 \end{bmatrix}$$

are correct. As the method used is a standard one, details of the procedure will not be given here but may be found if required in (33). Furthermore the random generator used has been tested and found to be satisfactory. Details are given in appendix B.

The table below gives the results of an investigation where values of

$$X_3^{w,\ell}\begin{bmatrix} b_3 & b_2 & b_1 \\ a_3 & a_2 & a_1 \end{bmatrix}$$

were calculated for differing values of a_i, b_i using the 2 methods. The monte-carlo method employed 10,000 random points (see (33) for explanation), and w was set to +5 and ℓ to -2.

	a_1	b_1	a_2	b_2	a_3	b_3	Monte-Carlo Method	Analytic Method
(1)	0.000	1.000	0.000	1.000	0.000	1.000	0.3698	0.3333
(2)	0.500	1.000	0.500	1.000	0.500	1.000	0.0429	0.0417
(3)	0.00	1.000	0.500	1.000	0.500	0.900	-0.1357	-0.1247
(4)	0.100	0.900	0.700	0.700	0.200	0.800	0.0000	0.0000
(5)	0.000	1.000	0.200	0.200	0.000	0.500	0.0000	0.0000
(6)	0.400	0.700	0.100	0.200	0.200	0.500	0.0416	0.0415
(7)	0.400	0.700	0.500	0.600	0.200	0.900	0.0032	0.0023
(8)	0.400	0.700	0.200	0.500	0.600	0.800	-0.0255	-0.0255
(9)	0.400	0.700	0.600	0.800	0.400	0.700	-0.0269	-0.0267
(10)	0.400	0.700	0.500	0.600	0.200	0.800	0.0087	0.0083
(11)	0.200	0.400	0.300	0.800	0.100	0.306	-0.0325	-0.0330

Closer agreement between the 2 sets of figures is obtained when the number of random points used by the monte-carlo method is increased. For example, in the worst possible case, the 7th line of the table shows that for 10,000 points the monte-carlo method gives a value of .0032 while repeating the calculation using 10^5 points (and taking 20 sec. of c.p. time) gives a value of .0024 which is much nearer to the correct answer. However this was not done with the remaining set of results for reasons of economy. Thus the results serve to confirm the accuracy of the analytic expression given by eq.37.

Note that the monte-carlo method could not itself be used to evaluate these integrals because it is:

(a) too slow (2 sec. per 10,000 points)

(b) not sufficiently accurate unless exceptionally large numbers of points are used.

The validity of the analytic expression obtained for

$$X_4^{w,\ell}\begin{bmatrix} b_4 & & b_1 \\ a_4 & \cdots & a_1 \end{bmatrix}$$

was confirmed in the same way.

APPENDIX B

RANDOM NUMBER GENERATOR

This appendix will discuss the random number generator used to implement the simulation techniques employed in this study.

The random number generator used is of the linear congruential type, see Knuth (21), and is provided as a standard library subroutine (RN2) on the CDC-6000 series computers (see CDC Reference Manual (5)).

The following statistical test was applied to this generator. Since this is a standard test (14,17,18,21), only a brief outline will be given here, in which a familiarity with the details of this test will be assumed.

The random number generator provided a sequence of numbers which were tested for uniform distribution over the interval (0,1) as follows. The interval was divided into ten equal subintervals, and f_i, the number of random numbers falling into the i-th interval was counted.

If n is the length of the sequence, then the statistic

$$\chi_1^2 = \frac{10}{n} \sum_{i=1}^{10} (f_i - \frac{n}{10})^2$$

has a χ^2 distribution with 9 degrees of freedom. This experiment is repeated m times and the m values of χ_1^2

thus obtained are compared with the theoretically expected distribution of χ^2 values. The results obtained with $n=1000$ and $m=1000$ show that the assertion that the original random number sequence is uniformly distributed is correct at the 60% level, a satisfactory result (see Snedecor and Cochran (34)).

A second test was carried out where f_{ij} was defined to be the number of numbers in the i-th interval, followed by a number in the j-th interval. The χ^2 statistic thus becomes

$$\chi^2 = \frac{10^2}{n} \sum_{i,j=1}^{10} \left(f_{ij} - \frac{n}{10^2}\right)^2 .$$

Proceeding as before, it was found that the assertion that consecutive random numbers are not pairwise correlated, was correct at the 90% level, again a satisfactory result.

<u>Generation of n random numbers from a single random number</u>

The following, standard, technique, is used to generate n, non-correlated random numbers from one single random number. This method will be described for the case $n=2$ from which the generalization of this method for $n > 2$ is an obvious one.

The method works by dividing the interval $(0,1)$ into m^2 equal subintervals $I_1,\ldots I_{m^2}$ such that these subintervals cover the interval $(0,1)$. A 1-1 mapping,

f, is then defined from any interval I_k to some integer pair (i,j) where (i,j) belongs to the set $S = \{(0,0)\ldots(0,n),(1,0)\ldots(1,n)\ldots(n,0)\ldots(n,n)\}$ with $n=m-1$. This mapping is defined in the following way.

Suppose that k has the unique representation

$$k = m(i-1) + j \quad \text{where} \quad 0 \le j \le m-1,$$

then $f(I_k)$ maps on to (i,j).

Now define $r_1 = \frac{i}{m-1}$ and $r_2 = \frac{j}{m-1}$,

then since $0 \le i \le m-1$ and $0 \le j \le m-1$, it follows that $0 \le r_1 \le 1$ and $0 \le r_2 \le 1$. Thus the function f may be used to produce the two random numbers r_1, r_2 from the single random number r. Also, since f is a 1-1 mapping, the components of the random pair r_1,r_2 are not correlated if the original sequence is not correlated.

The above method is used to generate random numbers which are used in simulating the hands dealt in the 2-PG, 3-PG and 4-PG (see chapters 4 and 5). In actual implementation the above process is easily carried out by dividing the computer word into 2 halves, where r_1 is calculated directly from the first half of the word, while r_2 is calculated from the second half.

APPENDIX C

CALCULATION OF PROBABILITIES

This appendix describes the use of a monte-carlo method to calculate the probabilities of winning associated with any given poker hand (see chapter 2). Two types of probabilities are required.

(a) $f_b(h)$, the probability that a given hand has of beating any other random <u>unimproved</u> hand.

(b) $f_a(h)$, the probability that a given hand h has of beating any random <u>improved</u> hand.

Since the methods used to find $f_b(h)$ and $f_a(h)$ are very similar, only the calculation of $f_b(h)$ will be described in detail.

The method used to calculate $f_b(h)$ is based on the following idea.

Given any hand h, $f_b(h)$, may be approximately determined thus. First, a very large number, N, of random hands are generated. The method of hand generation is the same as was used for the poker simulator (see chapter 2). It has been shown in chapter 2 that the hands produced in this way are random. Next, L, the number of times that the given hand h beats the randomly generated hands, and LD, the number of times that the given hand draws with the random hands, is found. Then $f_b(h)$ is approximated by (L+LD/2)/N. Since this process is time consuming (for large N) some measures were taken to speed the calculation.

1. Instead of generating N random hands each time that $f_b(h)$ is calculated, the set of N random hands is only generated once, and then stored. All $f_b(h)$ values are then calculated with respect to this same set of random hands.

2. The process of determining whether one hand beats another hand is repeated many times during the execution of this algorithm. For this reason the following method was devised to speed up this hand comparison process.

It is possible to define a function q(h) (see later) which assigns a unique integer to every hand h. Furthermore, the function q(h) has the property that, hand h_1 beats hand h_2 if and only if $q(h_1) > q(h_2)$, and hand h_1 draws with hand h_2 if and only if $q(h_1) = q(h_2)$. The definition of this function, and the proof that it has the above properties, will be given later in this text.

Thus, instead of storing N random hands, the N corresponding numbers, as defined by the function q(h), are stored. Then, given any hand h, q(h) may be calculated for that hand, and the hand comparison process may be achieved by testing the number of times that the integer q(h) is greater than or equal to the given integers in the set of N numbers already generated.

The method used to store the N numbers was to put them in an array (A(I), I=1,N). An alternative approach would be to have an array C(1), C(2),...C(K) where the element C(J) counts the number of times that the integer J occurs in the set of N numbers. However, since K(maximum of q(h)) is approximately 13^7 (see definition of q(h) later in text), it is clearly impractical to employ the latter method.

A drawback of the method used here is that it does not give a reliable estimate of how often a particular type or class of hand occurs when the percentage of occurrences of that particular class among the N random hands generated is small. But, for the purposes of this study, invariably only large classes of hands are dealt with, for example, the percentage of hands weaker than a pair of 8's. Hence this particular disadvantage, in this case, is of no practical consequence.

The process of finding $f_b(h)$ will now be summarised.

(i) Generate a random hand h, and calculated q(h). Repeat this process N times (N ≈ 30,000) and store the resultant numbers in the array (A(I), I=1,N).

(ii) Given any hand h for which $f_b(h)$ is required, first calculate $q(h)$. Now compute L, the number of times that $q(h) > A(I)$, and LD, the number of times that $q(h) = A(I)$, for I=1,N. Then $f_b(h)$ is approximated by (L+LD/2)/N.

A very similar process is used to compute $f_a(h)$ except this time the array (A(I), I=1,...N) is computed in the following way. A random hand is dealt, and if this hand is a pair or better then it is randomly improved, taking into account cards already held, to some new hand h*. If the hand is weaker than 1 pair it is discarded. This selective process is used because, in poker, players seldom, if ever, play on less than a pair (see chapter 5). As before this process was repeated N times (not counting hands weaker than 1 pair) and each time q(h*) was stored in the array (A(I),I=1,...N). Now $f_a(h)$ may be calculated in the same way as $f_b(h)$.

Definition of the function q(h).

The function q(h) is calculated according to table 20. This table will not be discussed in detail as it is not of central importance to this thesis. However, by careful examination of this table the following points may be validated.

TABLE 20

DEFINITION OF HAND SEQUENCING FUNCTION

Hand Type	Definition of variables	Value of q(h)
1) less than 1 pair	I1,I2,I3,I4,I5 are hand values†, in descending order of magnitude	$q(h)=I1\cdot 13^4+I2\cdot 13^3+I3\cdot 13^2+I4\cdot 13^1+I5$ max.value of $q(h) < 12(13^4+..+13^0)$ $= 12(13^5-1) < 13^6$ §
2) 1 pair	I1 = value of pair I2,I3,I4, are hand values of unpaired cards in descending order of magnitude	$q(h)=13^6+I1\cdot 13^3+I2\cdot 13^2+I3\cdot 13+I4$ max.value of $q(h)$ $< 13^6+12(13^3+13^2+13+1)$ $< 13^6+13^5$
3) 2 pair	I1=value of higher pair I2=value of lower pair I3=value of unpaired card	$q(h)=13^5+13^6+I1\cdot 13^2+I2\cdot 13+I3$ max $q(h) < 13^5+13^6+13^4$
4) 3 of a kind	I1 = value of treble I2,I3 unmatched cards in descending order	$q(h)=13^4+13^5+13^6+I1\cdot 13^2+I2\cdot 13+I3$ max $q(h) < 2.13^4+13^5+13^6$
5) straight	I1 = value of highest card (Ace counts as 0 if it is first card in straight)	$q(b) = 2.13^4+13^5+13^6+I1$ max $q(h) < 13+2.13^4+13^5+13^6$
6) flush	I1,I2,I3,I4,I5 are hand values in descending order of magnitude	$q(h)=13+2.13^4+13^5+13^6+I1.13^4+I2.13^3+I4.13+I5$ max $q(h) < 13+2.13^4+13^5+2.13^6$
7) full book	I1 = value of three of a kind I2 = value of two of a kind	$q(h)=13+2.13^4+13^5+2.13^6+I1.13+I2$ max $q(h) < 13+13^5+2.13^4+13^5+2.13^6$
8) 4 of a kind	I1 = value of 4 of a kind	$q(h)=13+13^3+2.13^4+13^5+2.13^6+I1$ max $q(h) < 2.13+13^3+2.13^4+13^5+2.13^6$
9) routine	I1 as for straight (see above)	$q(h)=2.13+13^3+2.13^4+13^5+2.13^6+I1$ max $q(h) < 3.13+13^3+2.13^4+13^5+2.13^6 < 13^7$

§ The sum of the G.P. $a+ar+...+ar^n$ is given by $S_n = \frac{a(1-r^{n-1})}{1-r}$

† hand values are assigned in the following way (irrespective of suit)

CARD	2	3	4	5	6	7	8	9	10	J	Q	K	A
VALUE	0	1	2	3	4	5	6	7	8	9	10	11	12

1. Each unique hand (irrespective of suit) gives a unique integer $q(h)$. If h_1 and h_2 are 2 different hands then $q(h_1) \neq q(h_2)$, and if h_1 and h_2 are identical hands then $q(h_1) = q(h_2)$.

2. The value of $q(h)$ for hands of differing types† is so calculated that given hands h_1 and h_2 where h_1 is a stronger type of hand than h_2, then $q(h_1) > q(h_2)$.

3. If 2 hands h_1 and h_2 are of the same type, but hand h_1 beats hand h_2, then $q(h_1) > q(h_2)$.

Now, by using points 1 to 4 above it will be shown that:

hand h_1 beats hand h_2 if and only if $q(h_1) > q(h_2)$, and h_1 draws with h_2 if and only if $q(h_1) = q(h_2)$.

This statement will be proved in 4 steps.

(i) If h_1 beats h_2 then by 1,2 and 3 above, $q(h_1) > q(h_2)$

(ii) If $q(h_1) > q(h_2)$ then by 1. and (i) above, hand h_1 beats hand h_2.

(iii) If hand h_1 draws with hand h_2 then $q(h_1) = q(h_2)$ as from (i) and (ii) above $q(h_1) \leq q(h_2)$ and $q(h_1) \geq q(h_2)$, thus

---

†For definition of hand types see table 1.

$q(h_1)$ must equal $q(h_2)$.

(iv) If $q(h_1) = q(h_2)$ then hands h_1 and h_2 must draw by 1 above.

Thus, the proof is complete.

INDEX OF FIGURES

INDEX OF TABLES

BIBLIOGRAPHY

1. de BALBINE, G., "Note on random permutations", Mathematics of Computation, Vol. 21, No. 100 (1967) pp.710-712.

2. BELLMAN, R., BLACKWELL, D., "Some two-person games involving bluffing", Proc. Nat. Acadm. Sci., Vol. 35, (1949) p.600.

3. BOREL, E., and collaborators, "Traite du Calcul des Probabilities et de ses Applications", Vol. IV, 2, Chapt. 5 (1938) Paris.

4. CDC 6000 Series: Fortran Reference Manual.

5. CDC 6000 Series: Statistical Subroutines Reference Manual.

6. COFFIN, G.S., "The poker game complete", Faber (1955).

7. DAFERMOS, S.C., "An extended traffic assignment model with applications to two-way traffic", Transportation Science, Vol. 5, No. 4, Nov.71 U.S.A.

8. ELLINGER, M.,"Poker: How to play and win" Faber and Faber Ltd., London 1951.

9. EPSTEIN, R.A., "The theory of gambling and statistical logic", Academic Press (1967) p.223.

10. FINDLER, N.V., "Computer model of gambling and bluffing", IRE Transactions on Electronic Computers, Vol. EC-10, No. 1 (1961) pp.97-98.

11. FINDLER, N.V., and others, "Studies on decision making using the game of poker" (an abstract of an unpublished paper) SIGART (ACM newsletter) No. 31, Dec.1971.

12. FRIEDMAN, L., "Optimal bluffing strategies in poker", Management Science, Vol. 17, No. 12, (1971) pp.764-771.

13. GILLIS, MAYBERRY and von NEUMANN, "Two variants of poker", Annals of Mathematical Statistics, No. 28, (1953) Princeton University Press, pp.13-50.

14. GOOD, I.J., "The serial test for sampling numbers and other tests for randomness", Proceedings of Cambridge Philosophical Society, Vol. 49 (1953) pp.276-284.

15. HANDSCOMB, D.C., "Methods of Numerical Approximation" London, Methuen, New York (1964).

16. HARRIS, R.J., "Concepts of optimality in alternate routing networks", Proceedings of 7th International Teletraffic Congress, Stockholm (1973).

17. HULL, T.E. AND DOBELL, A.R., "Mixed Congruential Random Number Generators for Binary Machines", Journal of the ACM, Vol. 11, No. 1 (1964) pp.31-40.

18. HULL, T.E. AND DOBELL, A.R., "Random Number Generators" SIAM Review, Vol. 4, No. 3 (1962) pp.220-251.

19. KARLIN, S., "Mathematical Methods and Theory in Games, Programming and Economics", Vol. 2, Addison-Wesley, Reading, Mass. 1959 p.238.

20. KARLIN, S., RESTREPPO, R., "Multistage poker models", Annals of Mathematics, Study No. 39 (1957) pp.337-363.

21. KNUTH, D.E., "The Art of Computer Programming", Vol. 2 Addison-Wesley, Sect. 3.3.1.

22. KUHN, H.W., "A simplified two-person poker", Annals of Mathematics Studies, Vol. 24, No. 2 (1951) pp.286-295.

23. MARSCHAK, T., "An example of a modified gradient method of linear programming", Stanford Math. Studies in Social Sciences, Vol. 2 (1958) pp.146-151.

24. NASH, J., and SHAPLEY, L.S., "A simple three-person poker game", Annals of Mathematics Studies Vol. 24, Princeton University Press (1951) pp.105-116.

25. NASH, J., "Non-cooperative games", Annals of Mathematics Study Vol. 24 No. 2 (1951) pp.286-295.

26. von NEUMANN, J., "Zur [Theorie] der Geseltschaftsspiele", Math. Annalen, Vol. 100 (1928), pp.295-320.

27. von NEUMANN, J. and MORGENSTERN, O., "The Theory of Games and Economic Behaviour", 2nd Ed., (1947) Princeton University Press, Princeton, pp.186-219.

28. POTTS, R.B. AND OLIVER, R.M., "Flows in transportation Networks", Academic Press (1972).

29. PRUITT, W.E., "A class of Dynamic Games", Naval Research Logistics Quarterly, V.8 (1) 1961 March pp.55-78.

30. REECE, T. and WATKINS, A., "Poker: Game of Skill", Faber and Faber, London (1962).

31. ROSEN, J.B., "Existence and uniqueness of equilibrium points for concave n-person games" Econometrica Vol. 33 (1965) pp.520-525.

32. RUBENS, J., "Win at poker", Funk and Wagnalls (1968).

[illegible] "The monte-carlo method", Pergamon,

INSERT

37. FINDLER, N.V. and others, "Studies in Decision Making Using the Game of Poker", Proc. IFIPCongress 71, Ljubljana, Yugoslavia, pp.1448-1459; North-Holland, Amsterdam.

38. FINDLER, N.V. AND KLEIN, H., "Experiments with Inductive Discovery Processes Leading to Heuristics in a Poker Program", Proc. Comp. Science Conf. Part 1, Columbus, Ohio(1973). 6