INFORMATION TO USERS

This material was produced from a microfilm copy of the original document. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the original submitted.

The following explanation of techniques is provided to help you understand markings or patterns which may appear on this reproduction.

1. The sign or “target” for pages apparently lacking from the document photographed is “Missing Page(s)”. If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting thru an image and duplicating adjacent pages to insure you complete continuity.

2. When an image on the film is obliterated with a large round black mark, it is an indication that the photographer suspected that the copy may have moved during exposure and thus cause a blurred image. You will find a good image of the page in the adjacent frame.

3. When a map, drawing or chart, etc., was part of the material being photographed the photographer followed a definite method in “sectioning” the material. It is customary to begin photoing at the upper left hand corner of a large sheet and to continue photoing from left to right in equal sections with a small overlap. If necessary, sectioning is continued again — beginning below the first row and continuing on until complete.

4. The majority of users indicate that the textual content is of greatest value, however, a somewhat higher quality reproduction could be made from “photographs” if essential to the understanding of the dissertation. Silver prints of “photographs” may be ordered at additional charge by writing the Order Department, giving the catalog number, title, author and specific pages you wish reproduced.

5. PLEASE NOTE: Some pages may have indistinct print. Filmed as received.

University Microfilms International
300 North Zeeb Road
Ann Arbor, Michigan 48106 USA
St. John's Road, Tyler's Green
High Wycombe, Bucks, England HP10 8HR
LAKSHMANAN, K. B., 1950-
DECISION MAKING WITH FINITE MEMORY DEVICES.
The Ohio State University, Ph.D., 1977
Computer Science

University Microfilms International, Ann Arbor, Michigan 48106
DECISION MAKING WITH FINITE MEMORY DEVICES

DISSERTATION

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

By

K. B. Lakshmanan, B.E., M.E., M.S.

* * * * *

The Ohio State University
1977

Reading Committee:
Prof. B. Chandrasekaran
Prof. M. T. Liu
Prof. L. J. White

Approved By

Adviser
Department of Computer
and Information Science
In fond memory of my brother

K. B. SUBRAMANIAN
ACKNOWLEDGMENTS

I would like to express my appreciation to the members of my dissertation reading committee: Professor B. Chandrasekaran, Professor Ming-Tsan Liu, and Professor Lee J. White. Professor Chandrasekaran served as the dissertation supervisor. I would like to thank him for the many useful suggestions he made during our discussions while the work was in progress. His comments and criticisms considerably improved the readability of this presentation. Professor White gave a critical reading of the first draft of the dissertation and his comments led to many improvements. He also had an influence on my choice of this area for dissertation research. Professor Liu had many helpful suggestions and almost always words of encouragement.

The moral support of the members of my family and my friends was invaluable during the course of this work. In particular I would like to thank Amudhu S. Gopalan. I would also like to acknowledge the help of Louise Hastings in typing the many drafts of the dissertation.

This research was supported by the Air Force Office of Scientific Research, Grant 72-2351.
VITA

May 24, 1950 . . . . . . Born - Tanjore, India

1971 . . . . . . . . . . . B.E., Electronics and Communication Engineering, University of Madras, India

1973 . . . . . . . . . . . M.E., Electrical Communication Engineering, Indian Institute of Science, Bangalore, India

1973-1974 . . . . . . Teaching Associate, Department of Computer and Information Science, The Ohio State University, Columbus, Ohio

1974-1977 . . . . . . Research Associate, Department of Computer and Information Science, The Ohio State University, Columbus, Ohio

1976 . . . . . . . . . . . M.S., Computer and Information Science, The Ohio State University, Columbus, Ohio

PUBLICATIONS


"Multiple Hypothesis Testing with Finite Memory," accepted for publication, Cybernetics and Information Science, 1977.

FIELDS OF STUDY

Major Field: Computer and Information Science

Studies in Pattern Recognition. Professors B. Chandrasekaran and L. J. White

Studies in Digital Computer Architecture and Organization. Professor M. T. Liu

Studies in Computer Systems Programming. Professor D. K. Hsiao
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>VITA</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>viii</td>
</tr>
<tr>
<td><strong>Chapter</strong></td>
<td></td>
</tr>
<tr>
<td>I.  INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Purpose</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Definition of Finite Memory</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Complexity Theory Viewpoint</td>
<td>12</td>
</tr>
<tr>
<td>1.4 Organization</td>
<td>14</td>
</tr>
<tr>
<td>II. PREVIOUS RESEARCH IN FINITE MEMORY INFERENCE</td>
<td>20</td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>20</td>
</tr>
<tr>
<td>2.2 Time-Varying Finite Memory</td>
<td>23</td>
</tr>
<tr>
<td>2.3 Time-Invariant Finite Memory</td>
<td>26</td>
</tr>
<tr>
<td>2.3.1 Hellman-Cover Results on General 2-Hypothesis Testing</td>
<td>35</td>
</tr>
<tr>
<td>III. MULTIPLE HYPOTHESIS TESTING</td>
<td>54</td>
</tr>
<tr>
<td>3.1 Definition of the Problem of K-Hypothesis Testing</td>
<td>54</td>
</tr>
<tr>
<td>3.2 A Lower Bound for $P(e)$</td>
<td>55</td>
</tr>
<tr>
<td>3.2.1 $P_L^*$ for Bernoulli Observations</td>
<td>63</td>
</tr>
<tr>
<td>3.3 Issues in Determining a Tight Lower Bound</td>
<td>66</td>
</tr>
<tr>
<td>3.4 Close-to-Optimal Automata</td>
<td>73</td>
</tr>
<tr>
<td>3.4.1 Close-to-Optimal Automata for Bernoulli Observations</td>
<td>76</td>
</tr>
<tr>
<td>3.4.2 Extension to K-Hypothesis Testing ($K &gt; 3$)</td>
<td>81</td>
</tr>
</tbody>
</table>

vi
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5 Symmetric Hypothesis Testing</td>
<td>83</td>
</tr>
<tr>
<td>3.6 Discussion</td>
<td>89</td>
</tr>
<tr>
<td>IV. COMPOUND HYPOTHESIS TESTING</td>
<td>91</td>
</tr>
<tr>
<td>4.1 Definition of the Problem</td>
<td>91</td>
</tr>
<tr>
<td>4.2 Simple Hypothesis Testing in a Minimax Formulation</td>
<td>97</td>
</tr>
<tr>
<td>4.3 Compound Hypothesis Testing</td>
<td>102</td>
</tr>
<tr>
<td>4.4 Some Variations on the Theme</td>
<td>111</td>
</tr>
<tr>
<td>4.5 Discussion</td>
<td>120</td>
</tr>
<tr>
<td>V. THE TWO-ARMED BANDIT PROBLEM</td>
<td>123</td>
</tr>
<tr>
<td>5.1 Definition of the Problem</td>
<td>123</td>
</tr>
<tr>
<td>5.2 A Particular Case of TABP—TABPO</td>
<td>129</td>
</tr>
<tr>
<td>5.3 The Two-Armed Bandit Problem</td>
<td>134</td>
</tr>
<tr>
<td>5.4 Discussion</td>
<td>139</td>
</tr>
<tr>
<td>VI. CONCLUDING REMARKS</td>
<td>142</td>
</tr>
<tr>
<td>6.1 Summary</td>
<td>142</td>
</tr>
<tr>
<td>6.2 Suggestions for Further Research</td>
<td>149</td>
</tr>
</tbody>
</table>

APPENDIX

| A | 154 |
| B | 158 |

REFERENCES | 160 |
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>A class of $\epsilon$-optimal automata for Bernoulli 2-hypothesis testing</td>
<td>29</td>
</tr>
<tr>
<td>2.2</td>
<td>A class of $\epsilon$-optimal automata for 2-hypothesis testing</td>
<td>48</td>
</tr>
<tr>
<td>3.1</td>
<td>Linear Machines</td>
<td>68</td>
</tr>
<tr>
<td>3.2</td>
<td>A class of close-to-optimal automata for Bernoulli 3-hypothesis testing</td>
<td>77</td>
</tr>
<tr>
<td>3.3</td>
<td>A class of automata considered by Sagalowicz</td>
<td>86</td>
</tr>
<tr>
<td>3.4</td>
<td>A class of sub-optimal automata for symmetric hypothesis testing</td>
<td>87</td>
</tr>
<tr>
<td>4.1</td>
<td>An $\epsilon$-minimax class of automata. Self transitions are deleted</td>
<td>100</td>
</tr>
<tr>
<td>4.2</td>
<td>A class of asymptotically optimal deterministic automata</td>
<td>110</td>
</tr>
<tr>
<td>4.3</td>
<td>A class of automata for problem (4.39)</td>
<td>113</td>
</tr>
<tr>
<td>4.4</td>
<td>Cover-Hellman automata: (a) for $p_A + p_B &lt; 1$, and (b) for $p_A + p_B &gt; 1$</td>
<td>117</td>
</tr>
<tr>
<td>4.5</td>
<td>A deterministic automaton for problem (4.40) with simultaneous tossing</td>
<td>119</td>
</tr>
<tr>
<td>5.1</td>
<td>Optimal automaton for the TABPO. Self transitions are deleted</td>
<td>133</td>
</tr>
<tr>
<td>5.2</td>
<td>Cover-Hellman automata: (a) for $p_A + p_B &lt; 1$, and (b) for $p_A + p_B &gt; 1$</td>
<td>136</td>
</tr>
<tr>
<td>5.3</td>
<td>A close-to-optimal automaton for the TABP. Self transitions are deleted</td>
<td>137</td>
</tr>
<tr>
<td>5.4</td>
<td>A close-to-optimal deterministic automaton for the TABP</td>
<td>140</td>
</tr>
</tbody>
</table>
ABSTRACT

In this investigation, the effects of finite-memory constraints on decision-theoretic problems are studied. The constraints on memory arise as a result of implementing the statistical algorithms on a digital computer. The decisions are based on a finite-valued statistic, which can be stored in a finite number of bits of computer memory. The decision maker may now be viewed as a time-invariant, finite-state stochastic automaton, whose transition and decision rules do not change with time or observed data, but can be randomized. The results presented may also be viewed as specifying the capabilities of finite-memory devices in statistical decision making similar to the results on the recognition capabilities of finite-state automata or as specifying the memory complexity of statistical decision theory algorithms.

The particular problems studied are: multiple simple hypothesis testing, compound hypothesis testing, and the two-armed bandit problem. In all these cases, the aim is to derive nontrivial lower bounds on error probabilities achievable by automata of specified memory capacity, and to design automata whose performances match or are close to the bounds. The technical notion of close-to-optimality has the potential of making finite memory theory more broadly applicable.

The derivation of the greatest lower bound on the error probability for the general K-hypothesis testing problem is shown to lead to complicated optimization. A loose, yet nontrivial, lower bound
on the probability of error is then demonstrated. Close to optimal automata are constructed for a class of problems on Bernoulli observation space and another particular class called "symmetric hypothesis testing." The number of additional bits of memory required by these automata is independent of the problem parameters, and is only a function of $K$. Thus the class of automata is close to optimal for the class of $K$-hypothesis testing.

Compound hypothesis testing regarding the parameter $p$ of a Bernoulli random variable is considered in detail. We discuss the inadequacies of a Bayesian approach, which calls for the assumption of an arbitrary prior distribution on $p$. Therefore, the minimax principle is adopted. However, since there exist problems for which even trivial automata can be minimax optimal, additional optimality properties are exhibited for the automata presented in order to disqualify trivial solutions. Interesting compound hypothesis testing problems regarding the sum and the difference of the biases of two coins are studied. Close to optimal deterministic automata are also presented for all these problems.

The two-armed bandit problem involves conducting an infinite sequence of tosses, given two coins with unknown biases, so as to maximize the long-run proportion of heads. Both the Bayesian and the maxmin approach are shown to be either unnatural or inappropriate. In our approach, attention is restricted to the class of expedient machines in order to choose the best one among them. Expedient machines perform better than a scheme choosing the coins randomly at any instant. It is
also shown that exact information regarding the bias of one of the coins can save at most one bit of memory in the design of optimal procedures. A deterministic automaton that is close to optimal within two bits of memory is also constructed.

Interesting open problems for further research in this area are also indicated.
CHAPTER I
INTRODUCTION

1.1 Purpose

The area of this investigation, which could be termed finite memory statistical inference, lies at the interface between statistical decision theory and computer science. In particular, we study the effects of finite memory constraints on decision-theoretic problems such as hypothesis testing, estimation, etc. The finite memory constraints arise naturally as a result of implementing the statistical inference algorithms on a digital computer.

Suppose, for example, the outcome $X$ of a random experiment is a random variable that has a normal distribution with known variance $\sigma^2$ but unknown mean $\mu$. That is, $\mu$ is some constant, but its value is unknown. In order to infer some information about $\mu$, we decide to repeat the random experiment $n$ independent times, $n$ being a fixed positive integer, and under identical conditions. Let $X_1, X_2, \ldots, X_n$ denote, respectively, the outcomes obtained on these $n$ repetitions of the experiment. We then have a random sample $X_1, X_2, \ldots, X_n$ of size $n$ from a distribution which is normal with mean $\mu$ and variance $\sigma^2$.

Two kinds of estimates are in common usage. The first, called a point estimate, yields one value which is our best guess for the parameter based upon available information. The second kind of
estimate is called an interval estimate and yields a range of values which includes or captures the true but unknown \( \mu \). Another typical problem in statistical inference is hypothesis testing. It consists of deciding to accept, on the basis of observed data, one of several statements regarding the unknown parameter \( \mu \).

Sequential design of experiments refers to problems of inference characterized by the fact that as data accumulate the experimenter can choose whether or not to experiment further. If he decides to experiment further, he can decide which experiment to carry out next. In other words, these are situations where the experimenter has more than one random experiment to choose from at any stage and that the size and composition of the observed sample are not predetermined but depend, in some specified way, on the data themselves as they become available.

Classical statistical decision theory has dealt with these problems extensively. (See Ferguson, 1967). However, statisticians have almost always ignored the effects of memory restriction in storing the available data. This is understandable from a statistician's viewpoint. His main interest is in devising procedures that derive as much information as possible about the unknown parameter, given the random sample. On the other hand, an implementation of these decision procedures on a digital computer has to take into account the restrictions on computation time, storage capacity and other computational resources in order to ensure that "optimal" results are obtained under such restrictive environments.
Consider, for example, a 2-hypothesis testing problem regarding the parameter $\mu$ of a normal random variable. Let us suppose that the variance $\sigma^2 = 1$ and hence the probability density function for the distribution with mean $\mu$ is given by

$$h(x|\mu) = \frac{1}{(2\pi)^{\frac{1}{2}}} \exp((-x^2)/2), -\infty < x < \infty. \quad (1.1)$$

Let us also suppose that the mean $\mu$ is specified to be +1 or -1 under the two hypotheses $H_1$ and $H_2$, respectively. That is, we are interested in the hypothesis test

$$H_1: \mu = +1 \quad \text{vs.} \quad H_2: \mu = -1. \quad (1.2)$$

Given the random sample $X_1, X_2, \ldots, X_n$, the decision procedure responds with its decision, $H_1$ or $H_2$. However, for any finite sample size $n$ there is always a possibility of error. Let $\alpha_n$ and $\beta_n$ denote the probabilities of error of each kind, i.e.,

$$\alpha_n = \Pr\{\text{Decide } H_2 | H_1 \text{ is true}\}, \quad (1.3)$$

$$\beta_n = \Pr\{\text{Decide } H_1 | H_2 \text{ is true}\}. \quad (1.4)$$

It is well known that if the sample size $n$ is large the probabilities of errors can be made very small. The standard likelihood ratio decision procedure simply computes the ratio of the probabilities of occurrence of the sample $X_1, X_2, \ldots, X_n$ under either hypothesis and compares it against a threshold. For this problem, compute the likelihood ratio
and decide \( H_1 \) if \( \ell > 1 \), or \( H_2 \) otherwise. Also, since the observations are independent

\[
h(x_1,x_2,...,x_n | \mu) = \prod_{i=1}^{n} h(x_i | \mu).
\]

It can be shown that \( \alpha_n \to 0 \) and \( \beta_n \to 0 \), both exponentially, as \( n \to \infty \). To apply this procedure requires a memory capacity sufficient to store the entire sample of size \( n \). Observe that the memory capacity must grow indefinitely as \( n \to \infty \) to accommodate this. Such a solution is not acceptable if the available memory is limited.

In problems with a large sample size, the available memory can become a major limitation, especially if the processing is to be done on a small computer. The objective of this study is to devise decision schemes that consume a specified amount of memory in implementation, and yield results that are optimal for this memory. The precise definition of finite memory and optimality criterion will be made in the next section. The intuitive appeal of this problem is due to the obvious fact that all real data processing schemes store data in a finite storage facility.

1.2 Definition of Finite Memory

The subtleties and pitfalls in making a good definition of finite memory has already been pointed out by various authors (Cover, 1968, 1969; Chandrasekaran, 1970). In order to highlight the issues let us present again the 2-hypothesis testing on a normal random variable, considered above.
Given a random sample \( X_1, X_2, \ldots, X_n \) of size \( n \) drawn from a normal distribution of unknown mean \( \mu \) and a known variance \( \sigma^2 = 1 \), we are interested in the hypothesis test, regarding the parameter \( \mu \),

\[
H_1: \mu = +1 \quad \text{vs.} \quad H_2: \mu = -1.
\]

As we stated earlier the probabilities of errors of the two kinds, \( \alpha_n \) and \( \beta_n \), both tend to zero as \( n \to \infty \) if we employ a likelihood ratio procedure. However, the procedure requires storing all the previous observations as well as current ones, and hence the memory must grow unbounded as \( n \to \infty \). It is obvious that any truncation of memory to, say, the last \( k \) observations will preclude the convergence of \( \alpha_n \) and \( \beta_n \) to zero.

Statisticians are well aware of the technique of data reduction by means of **sufficient statistics**. For example,

\[
\overline{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i
\]

(1.6)

is a sufficient statistic for the mean of a normal distribution. It is called a sufficient statistic because it contains all the necessary information that can possibly be gained from the sample \( X_1, X_2, \ldots, X_n \), regarding the unknown parameter \( \mu \), either for the purpose of hypothesis testing or estimation. In fact, a decision procedure that decides the hypothesis \( H_1 \) if \( \overline{X}_n \geq 0 \), or \( H_2 \) otherwise is precisely equivalent to the likelihood ratio procedure mentioned above. However, in the case of normal distribution the random variable is a real number in the interval \((-\infty, \infty)\) and even in those cases in which \( X \) can assume only a
finite number of values, the sufficient statistic or the likelihood ratio can assume an infinite number of possible values, as the sample size \( n \to \infty \). This implies a need for an infinite memory. Further, as Cover (1969) points out, if one can store one real number, one may store any finite number of real numbers by the simple trick of interleaving the digits in the decimal expansion. In other words, if there is enough memory to store the sufficient statistic, then any finite number of observations can be stored in that memory. Thus, even the many-to-one mapping represented by the sufficient statistic has failed to reduce the required memory.

A realistic first step towards defining finite memory might be to consider rounding off the statistic at each stage. Observe that the sufficient statistic at each stage can be expressed recursively by

\[
\bar{X}_{n+1} = \left(\frac{n\bar{X}_n}{n+1}\right) + \left(\frac{X_{n+1}}{n+1}\right).
\]

(1.7)

Now, if at each stage \( \bar{X}_n \) may be recalled only to some arbitrary decimal place accuracy the algorithm

\[
\left\lceil \bar{X}_{n+1} \right\rceil = \left\lceil \left(\frac{n\bar{X}_n}{n+1}\right) + \left(\frac{X_{n+1}}{n+1}\right) \right\rceil
\]

(1.8)

results, where \( \lceil \bar{X}_n \rceil \) denotes the sequentially rounded off version of \( \bar{X}_n \). Consider the decision procedure which decides \( \mu = \pm 1 \) according as \( \bar{X}_n \gtrless 0 \). Cover (1969) showed that \( \lceil \bar{X}_n \rceil \) does not converge to \( \lceil \mu \rceil \) and that \( \alpha_n, \beta_n \) converge to nonzero limits. The fact that the probabilities of errors, \( \alpha_n \) and \( \beta_n \), do not go to zero even with an infinite sample may not be surprising, for by rounding off the
statistic at each stage we have indeed introduced the finite memory restriction. On the other hand, it will be demonstrated presently that it is possible to compress the data at any stage into a 2-valued statistic and yet achieve error probabilities arbitrarily close to zero for this infinite-sample case.

Consider, for example, a 2-valued statistic $T$ whose value at time instant $i$ is denoted by $T_i$. $T_i \in \{-1, 1\}$ and is given by

$$
T_i = -1, \text{ if } x_i < -R , \\
= +1, \text{ if } x_i > +R , \\
= T_{i-1}, \text{ otherwise,}
$$

where $R$ is some positive value. Let the decision made after receiving the entire sample of size $n$ be $\mu = \pm 1$ according as $T_n = \pm 1$. $T_0$, the starting value for the statistic may be chosen arbitrarily as either $-1$ or $+1$. Under the hypothesis $H_1$, the unknown mean $\mu = +1$, and hence

$$
\Pr(X_i > R) > \Pr(X_i < -R).
$$

Further,

$$
\lim_{R \to \infty} \frac{\Pr(X_i > R)}{\Pr(X_i < -R)} = \infty.
$$

Thus, as the sample size $n \to \infty$, for any positive value $R$,

$$
\lim_{n \to \infty} \Pr(T_n = +1) > \lim_{n \to \infty} \Pr(T_n = -1).
$$

Besides,

$$
\lim_{n \to \infty} \Pr(T_n = +1) = \lim_{n \to \infty} (1 - \alpha_n),
$$

$$
\lim_{n \to \infty} \Pr(T_n = -1) = \lim_{n \to \infty} \alpha_n.
$$
and this probability increases as \( R \) is increased. To be precise,

\[
\lim_{R \to \infty} \lim_{n \to \infty} (1 - \alpha_n) = 1.
\]

Hence, it is possible to decrease \( \alpha_n \) to arbitrarily small values. Similar arguments apply for \( \beta_n \). Thus, we have presented a scheme that compresses the available data into a 2-valued statistic and yet achieves arbitrarily small error probabilities for the infinite-sample case. It must be quickly pointed out that we do not imply that the finite memory decision schemes we construct always achieve zero or arbitrarily close to zero error probabilities as in the case of normal distribution. The above argument is presented only to demonstrate that the decision procedure that simply rounds off the statistic at each stage to desired accuracy shows an undesirable behavior for this problem and hence is not preferable as a general finite memory model.

In our approach to modeling the finite memory restriction, we will require the statistic on which the decisions are based to assume only a finite number of values. Then, the statistic can be stored in a finite number of bits of computer memory. In order to make precise our notion of finite memory we will consider a 2-hypothesis testing problem on a Bernoulli observation space in what follows. This problem is a typical example of a situation where we do not achieve a zero probability of error, under a finite memory restriction, even as the sample size tends to infinity.

Let \( X_1, X_2, \ldots \) be a sequence of independent, identically distributed random variables with possible values \( H \) and \( T \) such
that \( \Pr(X_1 = H) = p \), and \( \Pr(X_1 = T) = 1-p = q \). Consider a 2-hypothesis testing problem of the form

\[
H_1: \quad p = p_1 \quad \text{vs.} \quad H_2: \quad p = p_2,
\]

where \( 1 > p_1 > p_2 > 0 \). Let \( H_i \) have prior probability \( \pi_i, \pi_1 + \pi_2 = 1 \).

We wish to associate with the observed sequence \( X_1, X_2, \ldots \) a sequence of decisions \( d_1, d_2, \ldots \) about the true hypothesis \( H_t \). Assume that the observations are made at regular time intervals so that the sample number coincides with the discrete clock instant, i.e., observation \( X_n \) is made at time instant \( n \). Also, the decision \( d_n \) taken at time \( n \) is based on a statistic \( T \) that can assume only a finite number of values. In other words, the data are summarized by an \( m \)-valued statistic that is updated according to the rule

\[
T_n = f(T_{n-1}, X_n), \quad T_n \in S = \{1, 2, \ldots, m\},
\]

where \( T_n \) is the value of \( T \) at time \( n \) such that

\[
d_n = d(T_n), \quad d_n \in \{H_1, H_2\}.
\]

Clearly, the number of states, \( m \), is a measure of the memory size needed to store the above \( m \)-valued statistic and it corresponds to \( \log_2 m \) bits.

Observe that the pair \((f, d)\) describes a finite-state automaton with inputs \( X_n \), outputs \( d_n \) and the state space \( S \). The functions \( f \) and \( d \) correspond to the transition and output functions, respectively. The state of the automaton at time \( n \) is \( T_n \) with a state \( T_0 \in S \) designated the initial state. Thus we can present our finite-memory decision
schemes by displaying the state-transition diagram of the cor­responding automata. Further, the sequence \( T_n \) forms a discrete Markov chain over the state space \( S \), under either hypothesis. This enables us to use the theory of Markov chains extensively to prove the results of interest to us.

The objective is to determine \((f,d)\), the transition and decision functions, so as to minimize the expected asymptotic proportion of errors,

\[
P(e) = \mathbb{E}\left\{ \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} e_i \right\},
\]

where \( e_i = 1 \) or \( 0 \) according as \( d_i \neq H_t \) or \( d_i = H_t \), \( H_t \) denoting the true hypothesis. The limit in (1.12) exists with probability one, but its value depends on which hypothesis is true. Automata in which there are no absorbing states or classes of states are said to be ergodic. In the case of nonergodic automata, the value of the limit in (1.12) also depends on the absorbing states in which the machine becomes trapped. The expectation needs to be taken over these possibilities.

An alternate definition for the probability of error,

\[
P'(e) = \lim_{n \to \infty} \mathbb{E}\{e_n\},
\]

is also possible. But the limit in (1.13) may not always exist. An example is the case of periodic machines, where certain states can be occupied only at multiples of some time period. However, the optimal automata we seek in our investigation all turn out to be aperiodic and
ergodic. For such automata, the values $P(e)$ and $P'(e)$ agree and further,

$$P(e) = \lim_{n \to \infty} \Pr(d_n \neq H_t).$$  \hfill(1.14)

Hence, the computation of the stationary probabilities on the states allows us to determine $P(e)$ for any given scheme. Standard texts such as Parzen (1962), Kemeny and Snell (1960) and Karlin (1969) may be consulted for the appropriate techniques.

In this work, the transition and decision functions are allowed to be stochastic. That is, probabilities are specified with which the transitions or decisions are made. For example, the transition function might specify that the automaton must transit from state $i$ to state $j$ on an input observation $X = H$, with a probability $\delta$. One way of realizing such a randomization will be to observe another Bernoulli sequence $Y_1, Y_2, \ldots$ of independent, identically distributed random variables such that $\Pr(Y_1 = 1) = \delta$, and $\Pr(Y_1 = 0) = 1 - \delta$. Then, if the automaton is in state $i$ and an input $X = H$ is observed, it looks for the auxiliary random input $Y$. If $Y = 1$ the transition to state $j$ is made; otherwise the machine stays in the same state $i$. Observe that the sequence $Y_1, Y_2, \ldots$ must be independent of $X_1, X_2, \ldots$. In general, we might need an auxiliary random sequence $Y_1, Y_2, \ldots$, uniformly distributed on the interval $[0,1]$, to realize transitions with different probabilities.

The transition and decision functions are constrained to be both time- and data-invariant. In other words, at any time instant $n$ the functions $f$ and $d$ are independent of $n$ and the observed
past data $X_1, X_2, \ldots, X_{n-1}$, except for what is summarized through the statistic $T_{n-1}$. Hence, at any instant $n$ it is only necessary to know the past state $T_{n-1}$ and the present input $X_n$ in order to determine the state $T_n$ and the decision $d_n$. Allowing $f$ and $d$ to be dependent on time and/or data leads to much superior performance, because as we will argue in Chapter II such a dependency implies, in some sense, infinite data-processing memory, even though $m$, the memory for storing data, is finite. If an auxiliary stream of random variables is available for the computation of the functions $f$ and $d$, the schemes we present do not require any additional data-processing memory. Thus randomized transition rules will be permissible within the definition of finite memory as used in this dissertation.

1.3 Complexity Theory Viewpoint

As mentioned earlier, our objective is to study statistical decision theory problems where the decision maker is constrained to be a time-invariant, finite-state stochastic automaton. A statistician might view this study as a way of incorporating memory restriction on decision rules. A computer scientist might consider this work as specifying the ultimate capabilities of finite memory devices in statistical decision making, in a manner that parallels the results regarding the recognition and acceptance capabilities of finite-state automata in computability theory. Alternatively he might use this study to explain the tradeoffs among the measures TIME, MEMORY, and PROBABILITY OF ERROR.
One of the main aims of complexity theory is to determine rules that govern the tradeoffs among the measures TIME and MEMORY. A third dimension of PROBABILITY OF ERROR is added automatically in statistical problems. The classical statistical decision theory, in fact, studies the tradeoffs between TIME and PROBABILITY OF ERROR. Infinite-sample, finite memory theory, however, concerns the tradeoffs between MEMORY and PROBABILITY OF ERROR. Some work has also appeared in the literature—see Chapter II—in finite-sample, finite memory theory, which deals with the tradeoffs among all the measures mentioned above. But, all our results in this dissertation relate to only infinite-sample situations. That is, in the problems considered here, the memory capacity is finite, but the number of observations available is arbitrarily large. Allowing such an infinite-sample size, the problem of determining the optimal decision scheme that achieves the smallest probability of error for a given memory size is clearly equivalent to that of determining the smallest memory size necessary to achieve a probability of error less than a specified value. Viewed this latter way, the problem is one of memory complexity of statistical decision schemes.

We will also draw upon the prevalent notion of constructing close-to-optimal solutions if the optimal finite memory decision schemes are difficult to find. For specific classes of problems—for example, 3-hypothesis testing problems on Bernoulli observation space—we construct close-to-optimal automata in Chapter III. For any problem in this class, our design procedure will allow us to
construct close-to-optimal automata requiring only a finite number of additional bits of memory to match the performance of the optimal m-state randomized automaton. Furthermore, the additional memory requirement, if measured in bits, is independent of m and the specific problem parameters. In this sense they are close to optimal.

1.4 Organization

A key paper in the development of finite memory theory (Heilman and Cover, 1970) dealt with the problem of testing between two hypotheses defined on any arbitrary probability space. That is, the infinite-sample random sequence $X_1, X_2, \ldots$ is drawn according to a probability measure $P$ defined on an arbitrary observation space $X$. The hypothesis test is of the form

$$H_1: P = P_1 \text{ vs. } H_2: P = P_2,$$

where $P_1$ and $P_2$ are two known probability measures. This general formulation includes the hypothesis testing problems on unknown parameters, mentioned in Section 1.2. Hellman and Cover determined $P^*$, the greatest lower bound on the probability of error for any m-state automaton. In general, there does not exist an automaton that achieves a probability of error equal to $P^*$. However, a sequence of automata can be constructed that achieve error probabilities arbitrarily close to $P^*$. Thus, $P^*$ is achievable by this class of automata in a limiting sense.
Before finite memory theory can have a general impact on statistical decision making, a number of problem situations need to be identified and the corresponding theory developed. Our objectives for this dissertation will be to develop optimal or near-optimal finite memory algorithms to three major classes of problems, all of which have some form of hypothesis testing as one of their components. The three classes of problems are:

1. multiple simple hypothesis testing,
2. compound hypothesis testing, and
3. the two-armed bandit problem.

A statistical hypothesis is basically an assertion about the distribution of one or more random variables. If a hypothesis completely specifies the distribution, it is called a simple hypothesis. Observe that all the hypotheses presented until now are simple hypotheses. Multiple simple hypothesis testing involves situations where more than two simple hypotheses are specified. On the other hand, in compound hypothesis testing problems, the hypotheses are of the form

\[ H_i: P \in S_i, \ i = 1,2, \]

where \( S_i, \ i = 1,2, \) are sets of possible values for \( P. \) We study a particular class of compound hypothesis testing dealing with the parameter of a Bernoulli random variable. The two-armed bandit problem, defined a little later, lies in the area of sequential design of experiments and involves, as a component, compound hypothesis testing.
The organization is as follows. Chapter II provides a review of existing literature on finite memory inference. Chapter III deals with the problem of multiple hypothesis testing. The hypotheses to be resolved are of the form

\[ H_i: \ P = P_i, \ i = 1, 2, \ldots, K, \]  

where \( P \) is some probability measure defined on a probability space \((X, \mathcal{B}, P)\). The finite collection of probability measures \( P_1, P_2, \ldots, P_K \) are the \( K \) possible values of \( P \) under the \( K \) hypotheses. This problem is a natural extension of the 2-hypothesis case \((K = 2)\) considered by Hellman and Cover (1970). However, the methods used in developing optimal, time-invariant, randomized finite-state automata for 2-hypothesis testing have resisted extension to the general case.

We discuss some issues in establishing lower bounds on error probabilities and the design of close-to-optimal automata. First a set of inequalities relating performance and machine parameters is derived, on the basis of which some loose, but nontrivial, bounds are obtained. For the case of Bernoulli observations, a class of automata called "linear machines" is investigated and shown to have useful properties. The difficulty in deriving tight bounds is shown to be related to that in determining all the constraints that exist among the parameters of realizable automata. It is also shown that the class of linear machines is close-to-optimal for 3-hypothesis testing in the sense that it requires at most one extra bit of memory to match the performance of
an optimal automaton. Arguments are provided to show that this result is extensible for $K > 3$. Finally, a set of sufficient conditions on the $K$ hypotheses are derived under which it is possible to construct sub-optimal automata, which, with the addition of a finite number of bits of memory, match the performance of optimal automata. Optimal performance can be easily matched by the addition of memory, but normally the added memory is a function of problem parameters and can become arbitrarily large for some problems. Thus, what is noteworthy in the sub-optimal automata developed is that the number of extra bits needed does not depend on the problem parameters. Examples of such problems are given and the corresponding automata are displayed.

Chapter IV studies a class of compound hypothesis testing problems on a Bernoulli observation space. The hypotheses involve the parameter $p$ regarding Bernoulli random variables, as is the case in (1.9) of Section 1.2. The first problem considered involves hypothesis testing of the form

$$H_1: p > p_1 \text{ vs. } H_2: p \leq p_2,$$

(1.16)

where $1 > p_1 > p_2 > 0$. This is an extension of the simple hypothesis testing problem presented in (1.9). No arbitrary prior distribution is assumed over the problem parameter $p$. The greatest lower bound for the probability of error is derived employing the familiar minimax principle and an $\varepsilon$-minimax class of automata is exhibited. A solution is said to be $\varepsilon$-minimax if it achieves an error probability not greater than $\varepsilon$ over the minimax value. It is also shown that for the case of
minimax automata can be constructed. Further, a lower error bound on the probability of error applicable to minimax automata is constructed. The automata we construct achieve error probabilities arbitrarily close to this bound. Some interesting compound hypothesis testing problems involving the bias towards heads \( p_A \) and \( p_B \), respectively, of two coins labeled A and B are also considered. Optimal randomized minimax automata are presented for the following hypothesis tests:

\[
H_1: p_A > p_B \text{ vs. } H_2: p_A < p_B .
\] (1.18)

and

\[
H_1: p_A + p_B > 1 \text{ vs. } H_2: p_A + p_B < 1 .
\] (1.19)

Close-to-optimal deterministic automata are also constructed for all these cases.

Chapter V applies our results on compound hypothesis testing to a problem in the sequential design of experiments, generally known as the two-armed bandit problem. This problem is so named because it models the situation that exists in certain types of slot machines in gambling casinos. Given two coins, labeled A and B, with unknown biases \( p_A \) and \( p_B \) respectively towards heads, the objective is to conduct an infinite sequence of tosses so as to maximize the long-run proportion of heads obtained. It is clear that we must attempt to perform the compound hypothesis test

\[
H_1: p_A > p_B \text{ vs. } H_2: p_A < p_B
\]
and also at the same time try to maximize the proportion of heads.
This presents a conflict of aims that is characteristic of most problems in the sequential design of experiments. The experimenter has two aims as he chooses the coins for each toss: to maximize the proportion of heads (by selecting the coin that he believes to have a larger bias) and to increase the knowledge of the coin biases (by selecting the coin about which he knows least). The aims could conflict. Thus, the problem models the conflict between estimation and control that arises in the design of adaptive control systems. Our interest is to design a finite-memory algorithm for this problem. Again, we do not assume any prior distributions on the coin biases, and thus do not adopt a Bayesian formulation. We restrict attention to so-called expedient finite-memory schemes, which have the property that their performance is strictly superior to tossing the two coins randomly (except when the coin biases are identical). We derive the least upper bound on the asymptotic proportion of the choice of the correct coin, for expedient machines, and construct schemes that achieve these bounds arbitrarily closely. It is shown that the memory saved by providing the exact bias of one of the coins is less than one bit. A deterministic automaton is constructed that requires at most two extra bits of memory, to match the performance of the optimal randomized automaton, and thereby eliminate the need for artificial randomization.

Chapter VI summarizes our contribution and briefly describes promising problems for further research in this area.
CHAPTER II

PREVIOUS RESEARCH IN FINITE MEMORY INFERENCE

2.1 Introduction

The notion of finite-memory restriction in the context of statistical decision making was introduced by Robbins (1956). His interest was in determining the optimal scheme for successively choosing one of two ways of action, the choice each time being based on the results of a fixed number of previous trials. Each of the two ways of action, at each choice, may lead to success or failure. The objective is to maximize the long-run proportion of successes obtained. The problem can also be viewed as one of finding the optimal strategy for the following situation: Given two coins labeled A and B with unknown biases, conduct an infinite sequence of tosses so as to maximize the proportion of heads. In this form the problem is generally known as the two-armed bandit problem (TABP). Robbins defined the memory to be of size \( r \) if the decision at any instant is dependent only upon the results of the previous \( r \) trials. Robbins proposed an \textit{ad hoc} scheme which generated alternate blocks of tosses with coins A and B, and in which the coin-changes occur whenever a sequence of \( r \) consecutive tails are obtained. Isbell (1959) and Smith and Pyke (1965) successively introduced more sophisticated rules for tossing and changing coins that resulted in improved performance. Samuels (1968) introduced
randomized rules and demonstrated that each previous rule which had been proposed could be improved by using a corresponding randomized rule.

It is clear that this definition of finite memory is different from the one outlined in Section 1.2. Unlike Robbins' two-armed bandit problem, the experimental outcome space can be infinite for some hypothesis testing situations we consider later and an infinite state memory is needed to store exactly the outcome of even a single observation. Further, as Cover (1969) points out, by the simple trick of interleaving the digits in decimal expansion any finite number of observations can be stored in the memory needed for a single observation. This almost defeats the original objective of finite memory restriction. However, defining memory as the total number of states (or the logarithm of that, if measured in bits) of the decision maker avoids the above conceptual difficulties. Thus, we will model the decision maker as a finite-state automaton.

The study of finite-state automata—called learning automata—in the context of statistical decision making has witnessed a boom in recent years. The specifications for a learning automaton include rules for selection of states and for making decisions. Depending upon the properties of the state transition rule and the decision rule, two main currents in this area of research can be discerned: one deals with what has come to be known as variable-structure automata, whose transition rules themselves change as a function of observed data, the algorithms which effect these changes being called "reinforcement
algorithms". Progress in this area has been recently reviewed by Narendra and Thathachar (1974). The other current, which is presently called finite-memory decision theory, originated with papers by Cover (1969) and Hellman and Cover (1970). It deals with the design of optimal transition and decision rules for automata, where these rules are not data-dependent, i.e., the algorithms implementing these rules do not change with the data. Variations on the theme correspond to whether the transition and decision rules are randomized or deterministic, time-variant or time-invariant or whether optimality is to be achieved for finite or infinite number of observations. While these two areas of research have developed independently of each other, they can be unified within a hierarchical framework, as demonstrated by Chandrasekaran (1970). The organizing principle in the hierarchy is the amount and the kind of constraints placed on the transition and decision rules.

In addition to these research activities, a series of publications has appeared in the Soviet literature on the behavior of automata in random media. The automata were required to choose one of several ways of action at any instant, to which the environment (called a "random medium"), whose probabilistic characteristics are not known a priori, responds with a penalty or non-penalty. The goal was to minimize the long-run proportion of penalties. It is clear that this problem is mathematically equivalent to the two-armed bandit situation above. The objective of the study was to model the adaptive or learning behavior of biological systems. Tsetlin (1973) provides an extensive
review of this material. In fact, the notion of variable-structure automata originated in the Soviet literature.

Observe that in the case of time-varying automata, since the transition function is dependent on the sample number (which is the same as the discrete time instant), there is a need for a clock that keeps track of it. The time-varying scheme considered by Cover, in fact, requires a clock whose memory size grows unbounded as the sample number tends to infinity. In effect, then, while this model restricts the memory for data gathering to be finite, it allows unbounded memory for data processing. The same is true of variable-structure automata, whose transition rule changes as a function of observed data. Hence, it appears more natural to us to consider data- and time-invariant automata to model the finite-memory restriction. In other words, we will restrict attention to automata whose transition and decision rules, at any time instant $n$, are independent of $n$ and the observed past data $X_1, X_2, \ldots, X_{n-1}$. However, because of its conceptual interest, we will provide a brief review of time-varying rules in the next section.

2.2 Time-Varying Finite Memory

Time-varying, but data-invariant, finite-memory decision rules were introduced by Cover (1968, 1969). Specifically, he considered a two-hypothesis testing problem of the form

$$H_1: P = P_1 \quad \text{vs.} \quad H_2: P = P_2,$$

(2.1)

where $P$ is the probability measure according to which a sequence of independent, identically distributed random variables $X_1, X_2, \ldots$ is
drawn. The data are summarized after each observation by an m-valued statistic \( T \) which is updated according to the rule

\[
T_n = f(T_{n-1}, X_n, n), \quad T_n \in S = \{1, 2, \ldots, m\}
\]

(2.2)

where \( T_n \) is the state of memory at time \( n \). The decision at time \( n \) is

\[
d_n = d(T_n), \quad d_n \in \{H_1, H_2\}.
\]

(2.3)

Observe the dependence of \( T_n \) on \( n \) in (2.2). Hence the name time-varying finite memory.

The objective is to determine a pair \((f, d)\), the transition and decision functions, so as to minimize the asymptotic proportion of errors,

\[
P(e) = E\left\{\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} e_i\right\}
\]

(2.4)

where \( e_i = 1 \) or 0 according as \( d_i \neq H_t \) or \( d_i = H_t \), \( H_t \) denoting the true hypothesis. Denote the infimum on \( P(e) \) over all \( m \)-state automata by \( P^* \):

\[
P^* = \inf_{(f, d)} P(e).
\]

(2.5)

Cover (1969) demonstrated that \( P^* = 0 \) for \( m = 4 \). This was accomplished by using two statistics, each two-valued—one to remember the currently favored hypothesis and the other to keep track of the success or failure of test blocks of trials. Thus, the memory has 4 states or 2 bits. Koplowitz (1975) has recently shown that three states are sufficient for this purpose. He also showed that \((m + 1)\) states are
necessary and sufficient to resolve \( m \) hypotheses, for the case of Bernoulli random variables.

Cover's result on 2-hypothesis testing has been extended to the more difficult compound hypothesis testing and estimation problems. Hirschler (1974) showed that a four state time-varying memory is sufficient to resolve the compound hypothesis testing problem of the form

\[
H_1: p = p_0 \quad \text{vs.} \quad H_2: p \neq p_0,
\]

regarding the parameter of a Bernoulli random variable. Hirschler and Cover (1975) also showed that eight states are sufficient to determine the rationality or irrationality of the parameter of a Bernoulli random variable. Wagner (1972) constructed a set of rules for the estimation of the mean of a distribution. His approach was to partition the parameter space and perform a multiple hypothesis testing. For Bernoulli observations, he showed that his scheme requires \( k \) bits of memory to determine the parameter \( p \) with an error smaller than \( (1/2)^k \). He did not, however, prove his scheme to be optimal. Koplowitz and Roberts (1973) considered the problem of sequential estimation in the mean-square sense. That is, they wished to minimize the mean-square error \( J_n \) at each stage \( n \), where

\[
J_n = E\{(x_n - a_n)^2\}.
\]

Necessary and sufficient conditions are derived for optimizing both the updating of memory and the estimate. Samaniego (1976) studied the
problem of mode identification of a discrete random variable. For a K-valued discrete random variable, under the assumption of a unique mode, he viewed the problem as one of K-hypothesis testing and demonstrated that $4K(K-1)$ states are sufficient to achieve a zero probability of error.

Mullis and Roberts (1974) have formulated a more general finite memory information processing model that allows the input data sequence also to be generated by a finite Markov chain. They considered finite-time problems, and time-average problems over an infinite time interval. A variational approach that generates locally optimal solutions was used to minimize the cost. More recently, Cover, Freedman and Hellman (1976) studied the structure and the performance of optimal schemes for finite sample problem. They established the existence of an optimal rule that is deterministic and also exhibited its structure to be of the likelihood ratio form. That is, the transition depends only on the likelihood ratio and under appropriate numbering of the states of memory, higher likelihood ratio observations cause transitions to higher numbered states and lower likelihood ratio observations to lower numbered states.

2.3 Time-Invariant Finite Memory

Recall the definitions of time-invariant finite memory from (1.10) and (1.11). The 2-hypothesis problem of the form (2.1) has been solved by Hellman and Cover (1970) under such a memory restriction. Their general approach and results are of considerable importance in the development of our results in multiple hypothesis testing.
A general summary of their results is presented, in some detail, in Section 2.3.1. In this section, we confine ourselves to their results on the particular example of Bernoulli observations with equal prior probabilities for the two hypotheses and proceed with the review on other work in this area.

Consider the 2-hypothesis testing problem discussed in Section 1.2, and of the form

\[ H_i: \ p = p_i, \quad i = 1, 2, \]

regarding the parameter \( p \) of a Bernoulli random variable. Assume, without loss of generality, that the two known values \( p_1 \) and \( p_2 \) are such that \( 1 > p_1 > p_2 > 0 \). Further, let the two hypotheses have equal prior probabilities. Hellman and Cover (1970) derived a lower boundary on the operating characteristic of any \( m \)-state automaton, relating the parameters of the automaton to the statistics of the problem and also showed that the smallest achievable probability of error over all \( m \)-state automata is given by

\[ P^* = \left[ 1 + \gamma_{12}^{4(m-1)} \right]^{-1}. \]  

(2.7)

The parameter \( \gamma_{12} \) is a measure of the ability to resolve between \( H_1 \) and \( H_2 \), and, for this problem, is given by

\[ \gamma_{12} = p_1 q_2 / p_2 q_1. \]

(2.8)

where \( q_i = 1 - p_i, \quad i = 1, 2 \). Observe that \( \gamma_{12} > 1 \) and hence \( P^* \to 0 \) exponentially, as \( m \to \infty \).
Optimal automata that achieve $P^*$ as their probability of error do not exist. However, an $\varepsilon$-optimal class of automata can be constructed. That is, for any $\varepsilon > 0$, there exists an automaton in this class for which $P(\varepsilon) \leq P^* + \varepsilon$. Figure 2.1 depicts one such $\varepsilon$-optimal class of automata. The automata involve transitions only to adjacent states. The transitions are downwards on $X = T$ and upwards on $X = H$. In the extreme states 1 and $m$ where such transitions are not possible, self transitions result. These are deleted in the diagram for clarity.

Further, the transitions away from the extreme states 1 and $m$ involve artificial randomization. In state 1, an input $X = H$ results in a transition to state 2 only with a probability $\delta$, $0 < \delta \leq 1$. Similarly, the transition from state $m$ to $m-1$ on $X = T$ is made with a probability $k\delta$, where $k$ is set to its optimal value

$$k^* = \left(\frac{p_1p_2}{q_1q_2}\right)^{1/(m-1)}$$

(2.9)

The other transitions are all deterministic. The decision made in state $i = 1, 2, \ldots, m$ is $H_1$ or $H_2$ according as $i > m/2$ or $i \leq m/2$. Then, while $P(\varepsilon) > P^*$ for any $\delta > 0$, $P(\varepsilon) + P^*$ as $\delta \to 0$.

Further, while the parameter $\gamma_{12}$ is finite for this problem, there exist problems for which $\gamma_{12} = \infty$ and these instances are said to involve unbounded likelihood ratios. An example of such a situation is the 2-hypothesis testing on a univariate normal random variable with mean $\mu = +1$ (under $H_1$) and $\mu = -1$ (under $H_2$) and fixed known variance $\sigma^2 = 1$. The likelihood ratio $\ell_{12}(x)$, the ratio of the two density functions, is given by $\exp(2x)$, which is unbounded. Therefore,
Fig. 2.1 A class of $\varepsilon$-optimal automata for Bernoulli 2-hypothesis testing.
\( \bar{L}_{12} \) and \( L_{12} \), respectively the supremum and the infimum of the likelihood ratio \( L_{12}(x) \), equal \( \infty \) and 0. This results in
\[
\gamma_{12} = \frac{\bar{L}_{12}}{L_{12}} = \infty.
\]
Observe that if \( \gamma_{12} = \infty \), \( P^* = 0 \) for any \( m \geq 2 \).

In the above discussion, even though the transition and the decision functions were constrained to be data- and time-invariant, they were allowed to be stochastic. Hence, unless we assume the availability of an auxiliary stream of random variables, we have to take into account the memory needed for the randomizer, in which case the Hellman-Cover scheme becomes far from optimal. The above argument by Chandrasekaran (1970) stressed the importance of studying the performance of purely deterministic finite-state automata.

Hellman and Cover (1971) showed that there can be arbitrarily large discrepancies between the performance of randomized and deterministic automata. To be precise, they showed that for any memory size \( m < \infty \), and \( \delta > 0 \), there exist problems such that all \( m \)-state deterministic automata have probability of error \( P(e) \geq \frac{1}{2} - \delta \), while the optimal two-state randomized automaton has \( P(e) \leq \delta \). On the other hand, Hellman (1972) showed that the deterministic automata are asymptotically optimal in the sense that for any hypothesis testing problem there exists a finite \( b \) such that, for any \( B \), the optimal deterministic rule with \( B + b \) bits in memory has a lower error probability than the optimal rule with \( B \) bits in memory. Thus \( (b/B) \) gives the fraction of memory lost by deterministic rules. This is negligible for large \( B \) and, in fact, \( (b/B) \rightarrow 0 \) as \( B \rightarrow \infty \). Recall that for
the optimal randomized rule the probability of error tends to zero exponentially, as the number of states, \( m \), approaches \( \infty \). Also, observe that \( m = 2^B \). Hence, to prove the above fact that there exists a finite \( b \), independent of \( B \), for any particular problem, it is sufficient to present a deterministic automaton whose probability of error also goes to zero, exponentially in \( m \), albeit at a slower rate. However, it must be stressed that this additional memory requirement of \( b \) bits for the deterministic rules depends on the problem parameters and can become arbitrarily large, for some problems.

Chandrasekaran and Harley (1970) studied the problem of achieving the performance bound without randomization. They proposed a procedure in which the expedient of a "laboratory" processing a large number of problems is used to achieve error probabilities close to the lower bound for each problem. Horos and Hellman (1972) proposed a confidence model in which errors were weighted according to the confidence with which the decisions were made. They showed that under such a model optimal schemes are purely deterministic. Shubert (1974a) studied an interesting variant of a Bernoulli 2-hypothesis testing problem in which the machine observes not only the input sequence but also two reference random sequences with bias \( p_1 \) and \( p_2 \), respectively. In this case, for a hypothesis testing problem of the form (2.6), he showed that a deterministic automaton can perform better than the optimal randomized machine, with an additional memory requirement of one bit.
An inference problem of the form (2.6) is called Bernoulli symmetric hypothesis testing if the two given values $p_1$ and $p_2$ are such that $p_1 = 1 - p_2$, and if the two hypotheses have equal prior probabilities. For this case, (2.9) yields $k^* = 1$, and hence Figure 2.1 represents an $\epsilon$-optimal class of randomized automata when the coefficient $k$ is set $k^* = 1$. The deterministic automaton that results by letting $\delta = 1$ is known as the saturable counter (Hellman, 1972). Shubert and Anderson (1973) studied a form of deterministic automata called "generalized saturable counter". These automata allow transitions to nonadjacent states also and show a better performance than a saturable counter. Chandrasekaran and Lam (1975) proposed another class of automata and conjectured that the optimal deterministic rule lies within this class. They showed how members of this class can be constructed to give a steady state probability of error that decreases asymptotically faster in the number of states than the best previously known deterministic algorithm. But even for this simple case the optimal deterministic automaton, i.e., the deterministic automaton that achieves the smallest probability of error among all $m$-state deterministic automata, is not known. Hellman (1972) attempted to derive some general guidelines for the design of asymptotically optimal, deterministic rules. The rules developed were similar to quantized sequential probability ratio tests.

The finite sample problem has also been studied to some extent. Flower and Hellman (1972) examined the problem for the Bernoulli observations. They found that for optimal schemes randomization is
still needed for the transitions away from the extreme states. Samaniego (1974) showed that the Hellman-Cover machine structure is optimal for \( m = 3 \) even for the finite sample case if attention is restricted to symmetric problems and machines. Samaniego (1975) also later showed that the optimal level of the parameter \( \delta \) that regulates the probability of transitions out of the extreme states, tends to zero at the rate \((\ln n)/n\) in symmetric testing problems where \( n \) is the sample size. The particular problem of testing between two Gaussian distributions differing only by a shift has been examined by Freedman (1971), who demonstrated that the minimum probability of error goes to zero as \( \exp(-\ln n)^{1/2} \). Cover, Freedman and Hellman (1976) showed that the exact knowledge of the sample size can be used to obtain a lower error probability than in the infinite sample problem. This particular behavior appears to be the result of the specific definition for the probability of error they employed for the finite-sample problem. An attempt has also been made to characterize the structure of optimal finite-sample schemes. Under certain mild assumptions or in special cases they presented the structure of the optimal two-state, time-invariant rule.

The problem of determining \( P^* \) and a class of automata whose performance is arbitrarily close to \( P^* \) has remained an open problem for the general \( K \)-hypothesis testing, \( K > 2 \). Particular cases involving unbounded likelihood ratios have, however, been solved by Sagalowicz (1970) and Yakowitz (1974). Sagalowicz obtained the necessary and sufficient conditions for the existence of 3-state automata to test
three hypotheses and achieve an arbitrarily small but non-zero probability of error. He also considered symmetric K-hypothesis testing and obtained the best K-state symmetric automaton. Yakowitz extended the basic result of Sagalowicz for the K-hypothesis testing problem with unbounded likelihood ratios. Shubert (1974b) obtained an upper bound on the lowest achievable probability of error for a 3-state automaton testing a 3-hypothesis problem, using the minimax principle. He also proposed a version of minimax theorem applicable to finite memory problems.

Cover (1970) considered the finite memory estimation problem and showed that it can be broken down into two parts, one involving multiple hypothesis testing and the other an elaborate quantization problem. Hellman (1974) studied the Gaussian estimation problem and showed that the minimum mean-squared error is bounded below by that obtained by the best m-level quantizer. Samaniego (1973) investigated the estimation of the parameter of a Bernoulli distribution restricting attention to a certain form of automata to demonstrate locally admissible and minimax schemes.

A particular version of the two-armed bandit problem (TABP), discussed in Section 2.1, has also been studied by Cover and Hellman (1970), in the finite memory context. This particular version involves knowing the two bias values for the two coins, but not knowing which coin has which bias. A recent review of this topic is provided by Witten (1976). TABP is considered a good abstraction of a fundamental conflict between data gathering and control that arises in many areas.
such as adaptive control, clinical medical trials, and artificial intelligence (Chernoff, 1975; Yakowitz, 1969). For in the TABP the conflict arises because the choice of the coin at any instant can be governed by two desires: to maximize the immediate payoffs (control) by choosing the coin deemed to have the larger bias on the basis of available information or to improve the current estimates of the bias values (data gathering) by choosing the coin about which least is known, in the hope of gaining information which could lead to ultimate profit. The problem thus belongs to the area of sequential design of experiments (Degroot, 1970). As we noted before, the work of Tsetlin and his co-workers in the Soviet Union involves a slightly different formulation of precisely the same mathematical problem.

2.3.1 Hellman-Cover Results on General 2-Hypothesis Testing

We will present below in considerable detail the approach and the results of Hellman and Cover (1970) on the 2-hypothesis testing problem.

Let \( X_1, X_2, \ldots \) be a sequence of independent, identically distributed random variables drawn according to the probability measure \( P \) and let the hypothesis testing be of the form

\[
H_1: P = P_1 \quad \text{vs.} \quad H_2: P = P_2,
\]

where \( P_1 \) and \( P_2 \) are two known probability measures with prior probabilities \( \pi_1 \) and \( \pi_2 \), respectively, \( \pi_1 + \pi_2 = 1 \).
The data are summarized after each observation by an m-valued statistic $T$ which is updated according to the rule

$$T_n = f(T_{n-1}, X_n), T_n \in S = \{1, 2, \ldots, m\},$$

where $T_n$ is the state of the memory at time $n$. The decision at time $n$ is

$$d_n = d(T_n), d_n \in \{H_1, H_2\}.$$

The objective is to determine the pair $(f, d)$, the transition and decision functions, so as to minimize the expected asymptotic proportion of errors,

$$P(e) = E\left\{\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} e_i\right\},$$

where $e_i = 1$ or $0$ according as $d_i \neq H_c$ or $d_i = H_c$, $H_c$ denoting the true hypothesis. The functions $f$ and $d$ may be randomized, but must be, at any instant $n$, independent of $n$ and the observed past data $X_1, X_2, \ldots, X_{n-1}$. A state $T_0 \in S$ is designated the start state. In the case in which $f$ describes an aperiodic, ergodic process on the state space $S$,

$$P(e) = \lim_{n \to \infty} \Pr\{d_n \neq H_c\},$$

and the specification of $T_0$, the start state, is immaterial. Further, the error probability can not be lowered by randomization in $d$, the decision function, or $T_0$, the start state. This follows from the usual decision theoretic considerations. For example, if in state $i$
a decision \( d(i) = H_1 \) results in an error probability \( s_1 \) and a decision \( d(i) = H_2 \) results in an error probability \( s_2 \), any randomized procedure that decides \( H_1 \) and \( H_2 \) with probabilities \( t_1 \) and \( t_2 \), respectively, results in an expected error probability \( t_1 s_1 + t_2 s_2 \).

Clearly, \( t_1 s_1 + t_2 s_2 \geq \min(s_1,s_2) \). As a consequence, a deterministic procedure that always chooses the decision resulting in a lower error probability is at least as good as any randomized decision rule. Hence, it is sufficient to restrict attention to automata with deterministic decision functions while determining a lower bound for the probability of error. However, this does not imply that the optimal automata we display must necessarily have deterministic decision functions.

The transition function can be simply specified by a set of stochastic matrices \( [P_{ij}(x)] \), where \( i,j = 1,2,...,m, p_{ij}(x) \geq 0 \), and

\[
\sum_{j=1}^{m} p_{ij}(x) = 1, \text{ for all } i.
\]

Here \( p_{ij}(x) \) is the probability of transition from state \( i \) to state \( j \) on the input \( x = x \). Taking the expectation over \( x \), the state transition matrices under \( H_1 \) and \( H_2 \), \( [P^1_{ij}] \) and \( [P^2_{ij}] \), respectively, are obtained. That is, the elements \( p^1_{ij} \) and \( p^2_{ij} \) are given by

\[
p^t_{ij} = \int p_{ij}(x) dP_t(x), \quad t = 1,2.
\]

Letting \( \mu^t_i \) denote the stationary probability of state \( i \) under the hypothesis \( H_t \), define \( \mu^t \), the vectors of stationary state probabilities, by
\[ \mu^t = (\mu_1^t, \mu_2^t, \ldots, \mu_m^t), \quad t = 1, 2. \]

The vectors \( \mu^t \) can be obtained by solving

\[ \mu^t = \mu^t [P_{ij}]^t, \quad t = 1, 2. \]

Let the state space \( S \) be partitioned into two sets \( S_1 \) and \( S_2 \) such that decision \( H_t \) is made in the subset of states \( S_t, t = 1, 2. \) The long-run probability of error \( P(e) \) is now simply given by

\[ P(e) = \pi_1 \sum_{i \in S_2} \mu_1^i + \pi_2 \sum_{i \in S_1} \mu_1^i. \quad (2.10) \]

Define \( P^* \) to be the greatest lower bound on \( P(e) \):

\[ P^* = \inf_{(f,d)} P(e). \]

The approach will be to determine a set of constraints applicable to \( m \)-state automata and then to minimize \( P(e) \) subject to these constraints. The resulting minimum value \( P^* \) provides a lower bound on \( P(e) \). It is then demonstrated that this lower bound is indeed the greatest lower bound by displaying a sequence of automata that achieves a probability of error as close as one might wish to this lower bound.

It is necessary to relate the problem statistics and machine performance parameters in order to obtain the constraints applicable to \( m \)-state automata. The following definitions and lemmas, indeed, do that.
For simplicity, let the probability measure \( P_1 \) possess the density function \( h_1(x), i = 1,2 \). If not, the discussion below may be carried on in terms of Radon-Nikodym derivative density functions as was done by Hellman and Cover. Define the likelihood ratio

\[
\ell_{12}(x) = \frac{h_1(x)}{h_2(x)}.
\]  

(2.11)

Let \( \ell_{12}^{\ast} \) and \( \ell_{12} \) denote the essential supremum and the infimum of the likelihood ratio \( \ell_{12}(x) \). Define a parameter \( \gamma_{12} \):

\[
\gamma_{12} = \frac{\ell_{12}^{\ast}}{\ell_{12}}.
\]  

(2.12)

\( \gamma_{12} \geq 1 \), with equality holding true only if the two probability measures are equal almost everywhere. The parameter \( \gamma_{12} \) will be seen to be a natural measure of resolvability between \( H_1 \) and \( H_2 \).

**Lemma 2.1**

The probabilities of transition from state \( i \) to state \( j \), \( P_{ij}^1 \) and \( P_{ij}^2 \), satisfy the inequality,

\[
\ell_{12} \leq \frac{P_{ij}^1}{P_{ij}^2} \leq \frac{\ell_{12}^{\ast}}{\ell_{12}}, \text{ for all } i, j.
\]  

(2.13)

If both \( P_{ij}^1 \) and \( P_{ij}^2 \) are zero, the ratio is undefined.

**Proof.**

\[
\frac{P_{ij}^1}{P_{ij}^2} = \frac{\int P_{ij}^1(x)h_1(x)dx}{\int P_{ij}^2(x)h_2(x)dx}.
\]
Hence,

\[
\frac{p_{ij}^1}{p_{ij}^2} \leq \frac{\int p_{ij}(x) h_2(x) dx}{\int p_{ij}(x) h_1(x) dx} = \frac{\beta_{12}}{\gamma_{12}}.
\]

The other inequality is obtained by similar analysis.

Define the state likelihood ratio for state \( i \) to be

\[
\lambda_{i}^{12} = \frac{\mu_i^1}{\mu_i^2}, \quad i = 1, 2, \ldots, m,
\]

the ratio of the stationary probabilities under \( H_1 \) and \( H_2 \). Automata in which any state \( i \) can be reached from any other state \( j \) in a finite number of transitions with nonzero probability are said to be irreducible or ergodic. Irreducibility implies that \( \mu_i^1 > 0 \) and \( \mu_i^2 > 0, \quad i = 1, 2, \ldots, m \). State likelihood ratios may be undefined for some states in the case of reducible automata.

**Lemma 2.2**

For an irreducible automaton in which the states are numbered in nondecreasing order according to the state likelihood ratio \( \lambda_{i}^{12} \), the following relation holds:

\[
1 \leq \lambda_{i+1}^{12} / \lambda_{i}^{12} \leq \gamma_{12}, \quad i = 1, 2, \ldots, m-1.
\]

Proof. The lower bound follows from the assumption of nondecreasing order. Further, if the state space \( S \) is partitioned into two sets \( C \) and \( C' \), then in steady state the probability of transition from \( C \) to \( C' \) must equal the probability of transition from \( C' \) to \( C \), under either hypothesis. That is,
Suppose the lemma were false. Then for some \( i \in S \)

\[
\mu_j^1 / \mu_j^2 \leq c \quad \text{for all } j \in C = \{1, 2, \ldots, i\} \tag{2.18}
\]

\[
\mu_j^1 / \mu_j^2 > c \gamma_{12} \quad \text{for all } j \in C' = \{i+1, \ldots, m\}, \tag{2.19}
\]

where \( c = \lambda_{12}^{12} \).

From (2.13), (2.16) and (2.18)

\[
\sum_{j \in C} \sum_{k \in C'} \mu_j^1 P_{jk} \leq \sum_{j \in C} \sum_{k \in C'} (c \mu_j^2) \left( \frac{\overline{P}_{jk}}{\overline{P}_{jk}} \right),
\]

so that

\[
\sum_{j \in C} \sum_{k \in C'} \mu_j^1 P_{jk} \leq c \gamma_{12} \sum_{j \in C} \sum_{k \in C'} \mu_j^2 P_{jk}. \tag{2.20}
\]

Similarly,

\[
\sum_{j \in C} \sum_{k \in C'} \mu_j^1 P_{jk} > c \gamma_{12} \sum_{j \in C} \sum_{k \in C'} \mu_j^2 P_{jk}. \tag{2.21}
\]

From (2.16) the left sides of (2.20) and (2.21) are equal, and from (2.17) the right sides of (2.20) and (2.21) are equal, a contradiction. Hence the lemma. \( \Box \)

The following lemma is obvious from the proof of the previous one.
Lemma 2.3

\[
\frac{\lambda_{i+1}^{12}}{\lambda_i^{12}} = \gamma_{i2}, \quad i = 1, 2, \ldots, m-1,
\]

if and only if

\[
p_{ij}^{1} = \gamma_{12} p_{ij}^{2} \quad \text{for} \quad 2 \leq j = m+1 \leq m,
\]

\[
= \gamma_{12} p_{ij}^{2} \quad \text{for} \quad 1 \leq j = i-1 \leq m-1,
\]

\[
= p_{ij}^{2} = 0 \quad \text{for} \quad |i-j| \geq 2.
\]

Define

\[
g_{12}^{i2} = \min_{i \in S} \lambda_{i}^{12} \tag{2.22}
\]

and

\[
g_{12}^{i2} = \max_{i \in S} \frac{\lambda_{i}^{12}}{\min_{i \in S} \lambda_{i}^{12}} \tag{2.23}
\]

The quantity \(g_{12}^{i2}\) will be denoted the spread for reasons which will be clear shortly. Recall that the state likelihood ratio \(\lambda_i^{12}\) is simply the ratio between the state occupancy probabilities of state \(i\) under \(H_1\) and \(H_2\). The optimal decision rule in any state \(i\) will be

\[
d(i) = H_1 \text{ if and only if } \pi_i^{1} \mu_i^{1} > \pi_i^{2} \mu_i^{2}, \quad \text{and } d(i) = H_2, \text{ otherwise.}
\]

Except for degenerate situations, which we deal with later,

\[
\min_{i \in S} \lambda_i^{12} < \pi_2 / \pi_1,
\]

and

\[
\max_{i \in S} \lambda_i^{12} > \pi_2 / \pi_1,
\]

implying that the optimal decision rule given above will require us to decide the hypothesis \(H_1(H_2)\) in the state in which the maximum (minimum)
of the state likelihood ratio $\lambda_{12}$ occurs. Further, the confidence in decisions is largest in these two states and if the spread, which represents the ratio between the maximum and the minimum value of the state likelihood ratio, can be maximized, and if most decisions in steady state are made in these two states, the probability of error can be lowered considerably. The trick of artificial randomization, in fact, allows the concentration of the steady state probabilities into just two states, under either hypothesis, and thus the notion of spread is central to the theory of randomized automata.

It is intuitively clear that an irreducible automaton is better than a reducible one because of the lack of use of the transient states in the latter case. This fact can be mathematically presented as follows.

**Lemma 2.4**

The spread of a reducible automaton is less than or equal to $(\gamma_{12})^{m-2}$.

The proof of this lemma is nonessential for our development later and hence we will not present it here. On the other hand, if we include irreducible automata the following result holds:

**Lemma 2.5**

The spread of an $m$-state automaton is less than or equal to $(\gamma_{12})^{m-1}$. 
Proof. If the automaton is irreducible, \((m-1)\) applications of Lemma 2.2 yield the result. If the automaton is reducible, Lemma 2.4 shows it to be at least one state inferior. Hence the lemma. □

The above two lemmas and our discussion on spread implies that it is sufficient to concentrate on irreducible automata, in order to derive tight lower bounds. We are now in a position to derive the constraints applicable to \(m\)-state automata and demonstrate a lower bound on \(P(e)\).

Since \(g^{12}\) is the minimum state likelihood ratio, and
\[
\sigma^{12} \leq (\gamma^{12})^{m-1},
\]

\[
g^{12} \leq \mu_1^{1/\mu_1} \leq g^{12} (\gamma^{12})^{m-1}, \text{ for all } i \in S. \tag{2.24}
\]

If \(P_{12}\) and \(P_{21}\) denote the probabilities of error of the two kinds, i.e.,

\[
P_{12} = P(e|H_1) \text{ and } P_{21} = P(e|H_2),
\]

it follows from (2.24)

\[
P_{12} = \sum_{i \in S_2} \mu_i^1 \geq g^{12} \sum_{i \in S_2} \mu_i^2 = g^{12} (1 - P_{21}); \tag{2.25}
\]

\[
P_{21} = \sum_{i \in S_1} \mu_i^2 \geq \frac{1}{g^{12} (\gamma^{12})^{m-1}} \sum_{i \in S_1} \mu_i^1 = \frac{1}{g^{12} (\gamma^{12})^{m-1}} (1 - P_{12}). \tag{2.26}
\]

Recall our earlier partition of the state space \(S\) into subsets \(S_1\) and \(S_2\) such that decision \(H_t\) is made in the subset of states \(S_t\), \(t = 1, 2\). Multiplying (2.25) and (2.26) we obtain the lower boundary
for the operating characteristic of any $m$-state automaton:

$$P_{12}P_{21} \geq (\gamma_{12})^{-(m-1)}(1-P_{12})(1-P_{21}). \tag{2.27}$$

Observe that while (2.25) and (2.26) imply (2.27), (2.27) by itself does not imply (2.25) and (2.26). In other words, any $m$-state automaton that satisfies (2.25) and (2.26) has to satisfy (2.27), but there could exist an automaton satisfying (2.27) but not (2.25) and (2.26) individually. However, this is not the case as will be evident from the arguments to follow.

Straightforward Lagrange minimization of $P(e) = \pi_1 P_{12} + \pi_2 P_{21}$, subject to the inequality constraint (2.27) yields the minimizing values. However, the entire derivation holds true only under the nondegeneracy condition $(\gamma_{12})^{m-1} \geq \max\{\pi_1/\pi_2, \pi_2/\pi_1\}$. If $(\gamma_{12})^{m-1} < \max\{\pi_1/\pi_2, \pi_2/\pi_1\}$, no state likelihood ratio can cause a reversal of the a priori decision. That is, for all $i$ either

$$\pi_1^{\mu_1} > \pi_2^{\mu_1} \quad \text{or} \quad \pi_1^{\mu_1} < \pi_2^{\mu_1}.$$  

As a result, for optimal rules either $S_1$ or $S_2$ is a null set and hence (2.27) becomes a trivial constraint. The rule of deciding the hypothesis with larger prior probability achieves the smallest possible error, $\min\{\pi_1, \pi_2\}$. In other words, under a degenerate situation no $m$-state automaton can gather more information than that provided a priori. Thus for any $m$-state automaton, $P(e) \geq P^*$ where
\[ p^* = \frac{2(\pi_1\pi_2 \gamma_{12}^{m-1})^\frac{1}{2} - 1}{\gamma_{12}^{m-1} - 1}, \text{ if } \gamma_{12}^{m-1} \geq \max \{\pi_1/\pi_2, \pi_2/\pi_1\}; \]

= \min \{\pi_1, \pi_2\}, \text{ otherwise.} \quad (2.28)

So far we have only shown \( p^* \) to be a lower bound on \( P(e) \).

We now proceed to demonstrate that \( p^* \) is the greatest lower bound. First, we demonstrate that optimal decision rules that achieve \( p^* \), in general, do not exist.

**Lemma 2.6**

If \( m > 2 \), and \( (\gamma_{12})^{m-1} > \max \{\pi_1/\pi_2, \pi_2/\pi_1\} \), then \( P(e) > p^* \).

Proof. Assume that the states of the automaton are numbered in non-decreasing order according to the state likelihood ratio. From the derivation of \( p^* \), it is seen that \( P(e) = p^* \) implies that equality must hold in (2.27). This in turn implies that

\[ \mu^1_i = \mu^2_i = 0, \quad i \neq 1 \text{ or } m \]  

(2.29)

Since irreducible automata all have \( \mu^t_i > 0, \ t = 1,2, \) any automaton achieving \( P(e) = p^* \) must be reducible. But reducible automata do not achieve the spread \( (\gamma_{12})^{m-1} \) and hence \( p^* \) is unachievable in all but degenerate cases.

However, an \( \varepsilon \)-optimal class of automata can be demonstrated, i.e., for every \( \varepsilon > 0 \), there exists an automaton in this class for which \( P(e) \leq p^* + \varepsilon \). Thus, the lower bound \( p^* \), although unachievable, is still \( \varepsilon \)-achievable and hence the greatest lower bound on \( P(e) \).
Also, the fact that $P^*$ is the greatest lower bound on $P(e)$ proves our earlier assertion that there does not exist an automaton that satisfies (2.27) but not constraints (2.25) and (2.26) individually.

Figure 2.2 depicts one such $\varepsilon$-optimal class, with self loops deleted for clarity. First, define two sets $A_{12}(\varepsilon)$ and $A_{21}(\varepsilon)$ such that

$$\frac{\Pr(A_{12}(\varepsilon) | H_1)}{\Pr(A_{12}(\varepsilon) | H_2)} < \frac{\beta_{12}}{1 + \varepsilon}$$  \hspace{1cm} (2.30)

$$\frac{\Pr(A_{21}(\varepsilon) | H_1)}{\Pr(A_{21}(\varepsilon) | H_2)} > \left(\frac{1}{\beta_{12}} + \varepsilon\right)^{-1}$$  \hspace{1cm} (2.31)

Clearly, $A_{21}(\varepsilon)$ and $A_{12}(\varepsilon)$ are sets such that an input $X \in A_{21}(\varepsilon)$ highly favors the hypothesis $H_1$ whereas $X \in A_{12}(\varepsilon)$ highly favors $H_2$. Such a definition of the input sets $A_{12}(\varepsilon)$ and $A_{21}(\varepsilon)$ is necessary only in the cases where the supremum and the infimum of the likelihood ratio occur on events with probability measure zero. However, for a 2-hypothesis problem on a Bernoulli observation space of the form (2.6) these two sets $A_{21}(\varepsilon)$ and $A_{12}(\varepsilon)$ reduce to $\{H\}$ and $\{T\}$, respectively.

Every member of the $\varepsilon$-optimal class of automata has the following structure: the machine transits from state $i$ to $i + 1$ if $i < m - 1$ and $X \in A_{21}(\varepsilon)$, from $i$ to $i - 1$ if $i \geq 2$ and $X \in A_{12}(\varepsilon)$, and stays in the same state otherwise. Further, if in state 1 and $X \in A_{21}(\varepsilon)$, the machine transits to state 2 with a probability $0 < \delta < 1$; if in state $m$ and $X \in A_{12}(\varepsilon)$, it transits to state $m - 1$ with
Fig. 2.2 A class of ϵ-optimal automata for 2-hypothesis testing.
probability \( k_\delta \). The decision made in state \( i = 1, 2, \ldots, m \) is \( H_1 \) or \( H_2 \) according as \( i > m/2 \) or \( i \leq m/2 \). Further, let \( \delta \rightarrow 0 \) as \( \epsilon \rightarrow 0 \) and set \( k \) to its optimal value \( k^* \) given below:

\[
k^* = \begin{cases} 
(\gamma_1 \gamma_2)^{\frac{1}{4}(m-1)} \left( \frac{(\pi_1 \pi_2 \gamma_{12})^{\frac{1}{2}} - \pi_1}{\pi_1 \gamma_{12}^{\frac{1}{2}(m-1)} - (\pi_1 \pi_2)^{\frac{1}{2}}} \right), & \text{if } \gamma_{12}^{m-1} \leq \pi_1 \pi_2 \pi_1 / \pi_2, \\
0, & \text{if } \gamma_{12}^{m-1} < \pi_1 \pi_2 \quad \text{and} \quad \pi_2 \quad \pi_1 \pi_2, \\
\infty, & \text{if } \gamma_{12}^{m-1} < \pi_2 / \pi_1 \quad \text{and} \quad \pi_1 < \pi_2, 
\end{cases}
\] (2.32)

where

\[
\gamma_1 = \frac{\Pr\{A_{11}(\epsilon) \mid H_1\}}{\Pr\{A_{12}(\epsilon) \mid H_1\}},
\] (2.33)

and

\[
\gamma_2 = \frac{\Pr\{A_{21}(\epsilon) \mid H_2\}}{\Pr\{A_{12}(\epsilon) \mid H_2\}}.
\] (2.34)

Again, for the particular case of Bernoulli random variables in (2.6)

\[
\gamma_1 = \frac{p_1}{q_1},
\] (2.35)

and \( \gamma_2 = \frac{p_2}{q_2} \),

where \( q_i = 1 - p_i, \ i = 1, 2. \)

Letting \( \delta \rightarrow 0 \) as \( \epsilon \rightarrow 0 \) results in \( P(\epsilon) \rightarrow P^* \).
The following points regarding the solution are worth noting. The structure of the ε-optimal solution is highly conservative in the sense that the transitions occur only on high information events. Recall that the two input set $A_{21}(ε)$ and $A_{12}(ε)$ highly favor the hypotheses $H_1$ and $H_2$, respectively. Any input that does not belong to one of these sets can result only in self transition.

The deterministic automaton that results by letting $δ = 1$ and $k = 1$ in Figure 2.2 also realizes the maximum spread $σ_{12}^2 = (γ_{12})^{m-1}$ as $ε \to 0$. In other words, randomization plays no part in achieving the maximum spread. On the other hand, it is the artificial randomization that allows us to concentrate the steady state occupancy probabilities into just two extreme states, by letting $δ \to 0$, and to capitalize on the spread achieved to lower the probability of error. The randomization coefficient $k$ serves to compensate for the asymmetries in the problem statistics, such as unequal prior probabilities, and ε-achieve the lower bound. However, in the case of continuous distributions there appears to be a way to avoid this randomization by suitably redefining the input sets that lead to the transitions away from the extreme states (Hellman and Cover, 1970). Such is not the case for discrete distributions and hence deterministic automata, even though they can achieve the maximal spread, remain distinctly sub-optimal.

As we observed before, the ε-optimal class concentrates the steady state probabilities in the two extreme states, 1 and $m$, under either hypothesis, as $δ \to 0$. This implies that the actual
decision rule in states 2 to (m-1) is not crucial as far as the perfor-
"mance of the automata with $\delta \to 0$ is concerned. In fact, consider
an $\epsilon$-optimal class obtained by modifying the one presented before
(Figure 2.2), only as far as the decision rule in states 2 to (m-1) is
concerned, as follows: decide hypothesis $H_1$ or $H_2$, with equal
probability. The resulting $\epsilon$-optimal class still $\epsilon$-achieves $P^*$ as
$\delta \to 0$.

Observe that in all cases above we only allowed $\delta$ tend to
zero, but never set $\delta$ to zero. If $\delta = 0$, the resulting automaton
has absorbing states and so becomes reducible. As noted before in
Lemma 2.4, reducible automata are at least one state inferior to
irreducible ones. It is interesting to note that while the perform-
ance improves as $\delta \to 0$, setting $\delta = 0$ results in inferior
performance. Intuitively, this is as a result of the fact that with
$\delta = 0$ no learning is possible after a finite number of observations
due to absorption, while with $\delta$ small, but non-zero, it is always
possible to move out of the trap states and correct the past mistakes.
On the other hand, the practical limitation on letting $\delta$ become
very small is that convergence to steady state is slowed down. In
fact, as $\delta \to 0$ the resulting convergence time increases without
bound. Hence, if the number of observations allowed is also finite,
the schemes we present here are probably far from optimal.

We conclude this section with two examples to clarify the
issues.
Example 2.1

Consider the 2-hypothesis testing problem discussed in Section 1.2. The random variable $X$ is a univariate normal with variance $\sigma^2 = 1$. The hypotheses are on the unknown mean $\mu$:

$$H_1: \mu = +1 \; \text{vs.} \; H_2: \mu = -1.$$  

The two hypotheses have equal prior probabilities. In this case the likelihood ratio is given by

$$\Lambda_{12}(x) = \exp(2x).$$  

Hence,

$$\bar{\Lambda}_{12} = \infty$$
$$\underline{\Lambda}_{12} = 0$$

and $$\gamma_{12} = \frac{\bar{\Lambda}_{12}}{\underline{\Lambda}_{12}} = \infty.$$  

This results in $P^* = 0$ for any $m \geq 2$. Further define the two sets:

$$A_{21}(\epsilon) = \{x \mid x \geq R\},$$

and $$A_{12}(\epsilon) = \{x \mid x \leq -R\},$$

where $R$ is some positive threshold value. Consider a two-state automaton in which a transition from state 1 to state 2 occurs for $X \in A_{21}(\epsilon)$ and a transition from state 2 to state 1 occurs for $X \in A_{12}(\epsilon)$. It is clear from a steady state analysis of this automaton that $P(\epsilon) \to 0$ as $R \to \infty$. Thus this 2-state scheme can achieve a probability of error as close as one might wish to $P^* = 0$.  

Example 2.2

Consider the hypothesis testing problem on a Bernoulli observation space presented in (2.6):

\[ H_1: \ p = p_1 \ vs. \ H_2: \ p = p_2, \]

where \( 1 > p_1 > p_2 > 0 \). It is obvious from this restriction that \( X = H \) favors the hypothesis \( H_1 \) while the input \( X = T \) favors \( H_2 \). That is, the sets \( A_{l2}(e) \) and \( A_{21}(e) \), defined in (2.30) and (2.31), simply reduce to \( \{T\} \) and \( \{H\} \), respectively. Further, from (2.33) and (2.34)

\[ \gamma_1 = p_1/q_1, \]
\[ \gamma_2 = p_2/q_2, \]

where \( q_i = 1 - p_i, i = 1,2. \)

This results in

\[ \gamma_{l2} = p_1q_2 / p_2q_1 \]

For the case of equal prior probabilities

\[ p^* = \left[ 1 + (p_1q_2 / p_2q_1)^{l/2(m-1)} \right]^{-1}. \]

The automata in Figure 2.1 achieve the maximal spread \( \sigma_{l2}^2 = (\gamma_{l2}^{m-1}) \) and also achieve \( p^* \) as \( \delta \to 0 \) when \( k \) is set to its optimal value

\[ k^* = (\gamma_1 \gamma_2)^{l/2(m-1)} \]
\[ = (p_1p_2 / q_1q_2)^{l/2(m-1)}. \]
CHAPTER III

MULTIPLE HYPOTHESIS TESTING

3.1 Definition of the Problem of K-hypothesis Testing

Let \( X_1, X_2, \ldots \) be a sequence of independent, identically distributed random variables drawn according to a probability measure \( P \) defined on an arbitrary probability space \( (X, \mathcal{B}, P) \). Let \( P_1, P_2, \ldots, P_K \) be a finite collection of probability measures on this space and consider the \( K \) hypotheses

\[
H_i: P = P_i, \quad i = 1, 2, \ldots, K,
\]  

(3.1)

\( P_1 \) occurring with a prior probability \( \pi_i, \sum_{i=1}^{K} \pi_i = 1 \). We wish to associate with the observed sequence \( X_1, X_2, \ldots \), a sequence of decisions \( d_1, d_2, \ldots \) about the true hypothesis \( H_t \).

We will repeat below the finite memory model for completeness.

The transition and decision rules are given as:

\[
T_n = f(T_{n-1}, X_n), \quad T_n \in S = \{1, 2, \ldots, m\}, \quad m \geq K,
\]

(3.2)

\[
d_n = d(T_n), \quad d_n \in \{H_1, H_2, \ldots, H_K\},
\]

(3.3)

where \( X_n \) is the nth observation, and \( T_n \) is the state of the automaton at time \( n \). The transition function \( f \) and the decision function \( d \) may be randomized, but must be independent of \( n \) and the past data. A state \( T_0 \in S \) is designated the start state.
The goal is to minimize the asymptotic proportion of errors

\[ P(e) = E\left( \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} e_i \right) \]  

(3.4)

where \( e_i = 1 \) or 0 according as \( d_i \neq H_t \) or \( d_i = H_t \), \( H_t \) denoting the true hypothesis. In the case of ergodic automata, the expectation is taken only with respect to the prior distribution on the hypotheses. However, if the automata have absorbing states or classes of states, then the expectation is also taken with respect to the distribution of \( e_i \) induced by absorption. As we observed in Section 2.3.1, decision theoretic considerations show that the error probability cannot be lowered by randomization in \( d \) or \( T_0 \). Further, in the case in which \( f \) describes an aperiodic, ergodic process on \( \{1,2,\ldots,m\} \), \( T_0 \) need not even be specified and

\[ P(e) = \lim_{n \to \infty} \Pr\{d_n \neq H_t\}. \]  

(3.5)

Define \( P^* \) to be the greatest lower bound on \( P(e) \):

\[ P^* = \inf_{(f,d)} P(e). \]  

(3.6)

In the next few sections we discuss some issues in establishing \( P^* \).

3.2 A Lower Bound for \( P(e) \)

Let the probability measure \( P_i \) possess the density function \( h_i(x) \), \( i = 1,2,\ldots,K \). If not, the discussion below may be carried on in terms of Radon-Nikodym derivative density functions (Hellman and Cover, 1970). The random variable \( X \) belongs to the observation space \( X \). Let \( a,b\in\{1,2,\ldots,K\} \) in what follows. For the hypothesis pair \( (H_a,H_b) \), define the likelihood ratio
Let $\lambda_{ab} (x) = h_a (x)/h_b (x)$, $a \neq b$. \hfill (3.7)

Let $\bar{\lambda}_{ab}$ and $\underline{\lambda}_{ab}$ be the essential supremum and infimum of the likelihood ratio $\lambda_{ab} (x)$. Also let

$$\gamma_{ab} = \frac{\bar{\lambda}_{ab}}{\underline{\lambda}_{ab}}. \hfill (3.8)$$

As observed in Section 2.3.1, the parameter $\gamma_{ab}$ is a useful measure of the resolvability between the hypotheses $H_a$ and $H_b$. Clearly,

$$\bar{\lambda}_{ab} = \frac{1}{\underline{\lambda}_{ba}}, \hfill (3.9)$$

$$\underline{\lambda}_{ab} = \frac{1}{\bar{\lambda}_{ba}}, \hfill (3.10)$$

and

$$\gamma_{ab} = \gamma_{ba}. \hfill (3.11)$$

Some of the results of Hellman and Cover (1970) on 2-hypothesis testing can now be applied to the $\binom{K}{2}$ hypothesis pairs $(H_a, H_b)$, $a < b$. Section 2.3.1 provides all the necessary facts regarding the 2-hypothesis case that are needed for our development. Our approach to multiple hypothesis testing is also to derive a set of inequality constraints relating problem statistics and machine parameters, and use them to derive a lower bound on the probability of error. The following notations and lemmas closely follow Section 2.3.1.

Denote the probability of transition from state $i$ to state $j$ under hypothesis $H_a$, $a \in \{1, 2, \ldots, K\}$, by $p_{ij}^a$, and the stationary probability of state $i$ under this hypothesis by $\mu_i^a$. Further, define the state likelihood ratio under the hypothesis pair $(H_a, H_b)$ for state $i$

$$\lambda_{i}^{ab} = \frac{\mu_i^a}{\mu_i^b}, \quad i = 1, 2, \ldots, m; \quad a < b. \hfill (3.12)$$
Application of Lemmas 2.1 and 2.2 yield, for all $a, b \in \{1, 2, \ldots, K\}$ and $i, j \in \{1, 2, \ldots, m\}$, the following results regarding the state transition probabilities and the state likelihood ratios.

**Lemma 3.1**

\[
\lambda_{ab}^a \leq \frac{p_{ij}^a}{p_{ij}^b} \leq \lambda_{ab}^b, \ a < b. \tag{3.13}
\]

**Lemma 3.2**

For an irreducible automaton in which the states are numbered in nondecreasing order according to the state likelihood ratio $\lambda_i^{ab}$, the following relation holds:

\[
1 \leq \frac{\lambda_{i+1}^{ab}}{\lambda_i^{ab}} \leq \gamma_{ab}, \ i = 1, 2, \ldots, m-1; \ a < b. \tag{3.14}
\]

Note that there is no requirement that the numbering of the states according to the state likelihood ratios remain the same for all hypothesis pairs $(H_a, H_b), \ a < b$ for (3.14) to be valid. In other words, for a given numbering of states, their ordering according to the state likelihood ratios need not remain the same. In fact, it is an open problem to determine the precise conditions under which the ordering remains invariant for all hypothesis pairs.

Some more of the results of 2-hypothesis testing directly carry over to the multiple hypothesis case. For instance, attention can be restricted to irreducible automata, since they are at least one state superior to the reducible ones. In Section 2.3.1, $P^*$, the greatest lower bound on the probability of error applicable to $m$-state automata, was derived for the 2-hypothesis case. It was also shown that, in
general, there does not exist an automaton that achieves a probability of error equal to $P^*$. But an $\varepsilon$-optimal class of randomized automata was exhibited such that for any $\varepsilon > 0$ there exists an automaton in this class for which $P(\varepsilon) \leq P^* + \varepsilon$. Hence, for the $K$-hypothesis case also, if any greatest lower bound on the probability or error can be derived, it can only be $\varepsilon$-achieved and that the $\varepsilon$-optimal class of automata must involve randomization in the transition function $f$ at least in the case of discrete distributions. On the other hand, it is not clear if the $\varepsilon$-optimal class of automata, in the limit as $\varepsilon \to 0$, should concentrate the stationary state probabilities under any hypothesis into just $K$ states, as is the case for $K = 2$.

Further, the limitations on $m$-state automata can be viewed in terms of the spreads achievable for various hypothesis pairs. Again, for $a, b \in \{1, 2, \ldots, K\}$, defining

$$g^{ab} = \min_{i \in S} \lambda^{ab}_i, \quad a < b, \quad (3.15)$$

and the spread,

$$s^{ab} = \max_{i \in S} \lambda^{ab}_i / \min_{i \in S} \lambda^{ab}_i, \quad a < b, \quad (3.16)$$

the following result holds:

**Lemma 3.3**

For any $m$-state automaton,

$$s^{ab} \leq (s^{ab})^{m-1}. \quad (3.17)$$
Lemma 3.3 follows from Lemma 2.5. The following inequalities can be written down as a consequence of the above definitions, the normalization conditions on the stationary occupancy probabilities, i.e.,

\[ \sum_{i=1}^{m} \mu_{i}^{a} = 1, \quad a = 1, 2, \ldots, K, \]

and Lemma 3.3:

\[ 1 < \sigma_{ab}^{\text{ab}} \leq (\gamma_{ab})^{m-1}, \]

\[ \frac{1}{(\gamma_{ab})^{m-1}} < g_{ab}^{\text{ab}} \leq 1. \]

The quantities \( g_{ab}^{\text{ab}} \) and \( \sigma_{ab}^{\text{ab}} \), which are dependent on the machine structure, together define the minimum and maximum value for the state likelihood ratio \( \lambda_{1}^{\text{ab}} \) achieved by the machine for the hypothesis pair \((H_{a}, H_{b})\). Recalling the solution for the 2-hypothesis case, it would seem that the larger the ratio between these two extreme values, the more efficient the machine is in resolving between \( H_{a} \) and \( H_{b} \). In fact, for \( K = 2 \), the spread \( \sigma_{12}^{\text{ab}} \) turns out to be a central notion in the theory of \( \varepsilon \)-optimal class of automata. But, as we will show later, the definitions of these quantities \( g_{ab}^{\text{ab}} \) and \( \sigma_{ab}^{\text{ab}} \) need to be refined for the multiple hypothesis testing problem considered here.

Let the state space \( S \) be partitioned into \( K \) subsets \( S_{1}, S_{2}, \ldots, S_{K} \) such that the decision \( H_{i} \) is made in the subset of states \( S_{i} \), and let \( P_{ab} \) denote the probability of deciding \( H_{b} \), in steady state, when the true hypothesis is \( H_{a} \), i.e.,

\[ P_{ab} = \lim_{n \to \infty} \Pr(d_{n} = H_{b} \mid H_{t} = H_{a}). \quad (3.18) \]
Then
\[ P_{ab} = \sum_{i \in S_b} \mu_i^a, \quad (3.19) \]
and
\[ P(e) = \sum_{a \neq b} \pi_a P_{ab}. \quad (3.20) \]

Since \( g^{ab} \) is the minimum state likelihood ratio and \( \sigma^{ab} \) is the spread achieved for the hypothesis pair \( (H_a, H_b), \ a, b \in \{1, 2, \ldots, K\}, \) and \( a < b, \) the following relation holds:
\[ g^{ab} < \mu_i^a / \mu_i^b < g^{ab} \sigma^{ab}, \quad i = 1, 2, \ldots, m. \]

Using this fact and (3.19), which specifies the probabilities of errors, \( K(K-1) \) inequalities connecting these probabilities can be constructed as below.
\[ P_{12} = \sum_{i \in S_2} \mu_i^1 > g^{12} \sum_{i \in S_2} \mu_i^2 = g^{12} P_{22}. \]

Observe that as a consequence of (3.18)
\[ P_{22} = 1 - P_{21} - P_{23} \]
for a 3-hypothesis problem and hence,
\[ P_{12} > g^{12}(1 - P_{21} - P_{23}). \]

Similar analysis yields the other inequalities. Thus, for \( K = 2, \) we had
For $K \geq 3$, we have 6 inequalities as follows:

\begin{align*}
P_{12} &\geq g_{12}^{12}(1-P_{21}) , \\
and \quad P_{21} &\geq \frac{1}{g_{12}^{12}} (1-P_{12}) . \\
\end{align*}

Multiplying the inequalities in pairs to eliminate the $g_{ab}$ and replacing each $\sigma_{ab}$ by its maximal value $(\gamma_{ab})^{m-1}$, we arrive at $K(K-1)/2$ inequalities which for $K = 3$ are

\begin{align*}
P_{12}^2 P_{21} &\geq \gamma_{12}^{-(m-1)} (1-P_{12})(1-P_{13})(1-P_{21})(1-P_{23}), \\
P_{23}^2 P_{32} &\geq \gamma_{23}^{-(m-1)} (1-P_{21})(1-P_{23})(1-P_{31})(1-P_{32}), \\
and \quad P_{13}^2 P_{31} &\geq \gamma_{13}^{-(m-1)} (1-P_{12})(1-P_{13})(1-P_{31})(1-P_{32}). \\
\end{align*}
The above inequalities clearly hold for any m-state automaton. Let \( P^*_L \) denote the probability of error obtained by minimizing (3.20) subject to (3.23). The reader should note the conceptual difference between \( P^*_L \) and \( P^* \). \( P^* \) is, by definition, the lowest probability of error achievable for the problem. On the other hand, there is no guarantee that the inequalities leading to the derivation of \( P^*_L \) necessarily are tight or that they do capture all the constraints that might exist in any realizable automaton. Thus, while for \( K = 2 \), \( P^*_L \) and \( P^* \) turn out to be identical, for \( K > 2 \), \( P^* \) is generally greater than \( P^*_L \). In the next section, we consider a particular example to illustrate some of the reasons for this.

The derivation of \( P^*_L \) for the general K-hypothesis case involves the Lagrange minimization of (3.20), an expression involving \( K(K-1) \) variables, subject to \( K(K-1)/2 \) nonlinear inequality constraints of the form (3.23), and hence considerable algebraic manipulation. However, for the particular case of 3-hypothesis testing with equal prior probabilities for the hypotheses, it is somewhat manageable and we solve this case in Appendix A. The resulting expressions are complex, as can be expected. For the case of Bernoulli observations, the expressions are simpler and are presented in the next sub-section. On the other hand, the derivation of \( P^*_L \) is straightforward even for the K-hypothesis case, if the resolvability measures \( \gamma_{ab} \) are all equal. This is the case for the symmetric hypothesis testing problems considered in Section 3.5. Also, the fact that the derivation of \( P^*_L \) is straightforward if the quantities \( \gamma_{ab} \) are equal allows us to
present a further loose lower bound on \( P(e) \),

\[
P_{LL}^* = \left[ 1 + \frac{1}{K-1} \Gamma^{-1}\left(\frac{m-1}{m}\right) \right]^{-1},
\]

(3.24)

where

\[
\Gamma = \max_{a<b} \gamma_{ab},
\]

for any K-hypothesis testing with equal prior probabilities if we replace each \( \gamma_{ab} \) by \( \gamma^{-1}\left(\frac{m-1}{m}\right) \) before the minimization.

3.2.1 \( P_L^* \) for Bernoulli Observations

Consider a K-hypothesis testing problem on a Bernoulli observation space. Let \( X_1, X_2, \ldots \) be a sequence of independent, identically distributed random variables with possible values \( H \) and \( T \) such that \( \Pr(X_i = H) = p \) and \( \Pr(X_i = T) = 1-p = q \). Let the K hypotheses be of the form

\[
H_i: \quad p = p_i, \quad i = 1, 2, \ldots, K,
\]

(3.25)

where \( 1 > p_1 > p_2 > \ldots > p_K > 0 \). Let hypothesis \( H_i \) have prior probability \( \pi_i \), \( \sum_{i=1}^{K} \pi_i = 1 \).

We will derive \( P_L^* \) for the 3-hypothesis case with equal prior probabilities for the hypotheses. Observe that definition (3.8)

results in

\[
\gamma_{12} = \frac{p_1 q_2}{p_2 q_1},
\]

(3.26)

\[
\gamma_{23} = \frac{p_2 q_3}{p_3 q_2},
\]

(3.27)

and

\[
\gamma_{13} = \frac{p_1 q_3}{p_3 q_1} = \gamma_{12} \gamma_{23},
\]

(3.28)

where \( q_i = 1-p_i, \quad i = 1, 2, 3 \).
Further, from (3.18)

\[ P_{11} = 1 - P_{12} - P_{13}, \quad (3.29) \]
\[ P_{22} = 1 - P_{21} - P_{23}, \quad (3.30) \]
and \[ P_{33} = 1 - P_{31} - P_{32}. \quad (3.31) \]

Then the steady state probability of error will be

\[ P(e) = \frac{1}{3} \sum_{a \neq b} P_{ab}, \quad a, b \in \{1, 2, 3\}. \quad (3.32) \]

Also, any m-state automaton obeys the following constraints derived from (3.23),

\[ P_{12} P_{21} \leq \tau_{12}^{-1}(1-P_{12}-P_{13})(1-P_{21}-P_{23}), \quad (3.33) \]
\[ P_{23} P_{32} \leq \tau_{23}^{-1}(1-P_{21}-P_{23})(1-P_{31}-P_{32}), \quad (3.34) \]
\[ P_{13} P_{31} \leq \tau_{13}^{-1}(1-P_{12}-P_{13})(1-P_{31}-P_{32}), \quad (3.35) \]

where \( \tau_{12} = (\gamma_{12})^{m-1}, \tau_{23} = (\gamma_{23})^{m-1} \) and \( \tau_{13} = (\gamma_{13})^{m-1} \). (3.36)

A Lagrange minimization of (3.32) subject to (3.33) - (3.36) yields the optimal quantities \( \overline{P}_{ab}, a, b \in \{1, 2, 3\} \), such that

\[ \overline{P}_L = \frac{1}{3} \sum_{a \neq b} \overline{P}_{ab}. \quad (3.37) \]

The optimal values \( \overline{P}_{ab} \) do not correspond to the error probabilities achievable by any real machines, but merely stand for the results of the minimization operation. Appendix A presents the details of this optimization. The relationship \( \gamma_{13} = \gamma_{12} \gamma_{23} \) that obtains in the Bernoulli
case leads to simplification in the expressions for the quantities $\bar{P}_{ab}$. It can be checked by direct substitution that $\bar{P}_{ab}$ satisfy the following equations.

\[
\frac{\bar{P}_{12}}{\bar{P}_{11}} = \frac{\tau_{12} \tau_{23} - 1}{\tau_{12} (\tau_{23} + 1)} \quad (3.38)
\]

\[
\frac{\bar{P}_{13}}{\bar{P}_{11}} = \frac{\tau_{12} + 1}{\tau_{12} \tau_{23} + 1} \quad (3.39)
\]

\[
\frac{\bar{P}_{21}}{\bar{P}_{22}} = \frac{\tau_{23} + 1}{\tau_{12} \tau_{23} - 1} \quad (3.40)
\]

\[
\frac{\bar{P}_{23}}{\bar{P}_{22}} = \frac{\tau_{12} + 1}{\tau_{12} \tau_{23} - 1} \quad (3.41)
\]

\[
\frac{\bar{P}_{31}}{\bar{P}_{33}} = \frac{\tau_{23} + 1}{\tau_{23} \tau_{12} + 1} \quad (3.42)
\]

and

\[
\frac{\bar{P}_{32}}{\bar{P}_{33}} = \frac{\tau_{12} \tau_{23} - 1}{\tau_{23} \tau_{12} + 1} \quad (3.43)
\]

The quantities $\bar{P}_{ab}$ still satisfy (3.29) - (3.31). Observe that the optimal values are such that the inequalities (3.33) - (3.35) are satisfied with equality. In addition, the following relationship holds between them:
While $P_*$ is not a tight lower bound on $P(e)$, it is nevertheless not a trivial lower bound. This can be seen in Section 3.4, where we present a class of automata that is very close to optimal.

### 3.3 Issues in Determining a Tight Lower Bound

We remarked in the previous section that $P_*$, discussed earlier, is not in general equal to $P^*$, the greatest lower bound on the probability of error. In other words, there does not exist a class of automata that can achieve a probability of error arbitrarily close to $P_*$. We consider again the particular case of Bernoulli observations to discuss this issue. It must be remembered, however, that the issues raised do apply to any general $K$-hypothesis testing problem.

Consider a 3-hypothesis testing problem on a Bernoulli observation space of the form

$$H_i: p = p_i, \ i = 1,2,3, \quad (3.46)$$

where $1 > p_1 > p_2 > p_3 > 0$. Let the hypotheses have equal prior probabilities. We observed in Section 3.2.1 that the optimal values $\overline{P}_{ab}$ constituting $P_*^*$ are such that the constraints (3.33) - (3.35) are met with equality. Hence, in order to ε-achieve $P_*^*$ so that

$$\frac{P_{21}}{P_{22}} = \frac{P_{31}}{P_{32}} = \frac{1}{P_{12}}, \quad (3.44)$$

and

$$\frac{P_{13}}{P_{12}} = \frac{P_{33}}{P_{32}} = \frac{1}{P_{23}}. \quad (3.45)$$

While $P_*^*$ is not a tight lower bound on $P(e)$, it is nevertheless not a trivial lower bound. This can be seen in Section 3.4, where we present a class of automata that is very close to optimal.
the spreads $\sigma^{ab}$ achieved by some $\epsilon$-optimal class should, as $\epsilon \to 0$, approach, if not equal, $(\gamma_{ab})^{m-1}$, $a,b \in \{1,2,3\}$, $a < b$. In fact, there exists a class of automata depicted in Figure 3.1 for which

$$
\sigma^{12} = (\gamma_{12})^{m-1}, \sigma^{23} = (\gamma_{23})^{m-1} \text{ and } \sigma^{13} = (\gamma_{13})^{m-1},
$$

and yet $P_L^*$ is not $\epsilon$-achievable. In other words, even though the class of automata depicted in Figure 3.1 achieves maximal spreads for all hypothesis pairs, there does not exist an automaton in this class for which the probability of error is arbitrarily close to $P_L^*$.

The fact that the likelihood ratios $\ell_{12}(x)$, $\ell_{23}(x)$ and $\ell_{13}(x)$ all have their supremum and the infimum at $X = H$ and $X = T$, respectively, suggests an investigation of this class of automata. Self loops are deleted in Figure 3.1, and the class of automata shall be called "linear machines" for obvious reasons. Basically, $X = H$ results in transition towards higher numbered states and $X = T$ towards lower numbered states. The transitions away from states $1,m_1$ and $m$ involve artificial randomization. For example, the transition from state 1 to state 2 on input $X = H$ occurs only with a probability $\delta$.

Observe that for this class of automata the minimum and the maximum of the state likelihood ratios occur at states 1 and $m$, respectively, for all hypothesis pairs. The spreads achieved by a machine can be effectively used for minimizing the probability of error only if the state space can be partitioned into subsets $S_1,S_2$ and $S_3$ so that the state likelihood ratio $\lambda_{1}^{ab}$ has its minimum value in a state belonging to $S_b$ and its maximum value in a state belonging
Fig. 3.1 Linear Machines.
to $S_a$, for every pair $(a,b)$. Clearly, this is not the case for the automata in Figure 3.1, as seen by the above observation, and the $\epsilon$-achievability of $P_L^*$ is thus precluded for this class. In general, therefore, it is necessary to refine the definition of the quantities $a_{ab}$ and $g_{ab}$ to reflect the above arguments.

**Definition**

\[
a_{ab}^u = \frac{\max_{i \in S_a U S_b} \lambda_{i}^{ab}}{\min_{i \in S_a U S_b} \lambda_{i}^{ab}}, \quad a < b, \quad (3.48)
\]

\[
g_{ab}^u = \min_{i \in S_a U S_b} \lambda_{i}^{ab}, \quad a < b. \quad (3.49)
\]

The subscript $u$ is used to refer to the fact that they are "useful" parts of the original quantities. For the class of automata in Figure 3.1, with the decision rule shown there

\[
\sigma_{12}^u = (\gamma_{12})^{m-1}, \quad \sigma_{23}^u = (\gamma_{23})^{m-1}, \quad \text{and} \quad \sigma_{13}^u = (\gamma_{13})^{m-1}. \quad (3.50)
\]

Observe that as a consequence of the above definitions

\[
a_{ab}^u \leq a_{ab} \quad \text{and} \quad g_{ab}^u \geq g_{ab}. \quad (3.51)
\]

The quantities $a_{ab}^u$ and $g_{ab}^u$, in fact, should be used in the derivation of the inequalities (3.22), in order to derive tight constraints.

The set of inequalities (3.22) can be rewritten for the class of automata in Figure 3.1 by substituting the useful spreads in (3.50) and replacing $g_{ab}$ by $g_{ab}^u$. While these inequalities are intractable
enough, they still do not incorporate all the constraints which apply to realizable automata. Theorem 3.1, presented a little later, asserts the inability to achieve the maximal useful spreads simultaneously for all hypothesis pairs by any $m$-state automaton. Theorem 3.2 shows an additional constraint that exists among the quantities $q_u^{ab}$. In general, such complex relationships that might exist among $q_u^{ab}$ and $q_u^{ab}$ ought to be determined and included in the optimization constraints in deriving $P^\ast$. On the other hand, the derivation of $P_L^\ast$ simply ignored the notion of useful spreads and replaced each $\sigma^{ab}$ by its maximal value $(\gamma_{ab})^{m-1}$. Thus, the set of inequality constraints by themselves are not tight. Further, additional constraints relating the various $q_u^{ab}$ and $q_u^{ab}$ were ignored. As a result, in general, $P_L^\ast \neq P^\ast$, but in any case $P_L^\ast \leq P^\ast$.

Observe that $\sigma^{ab}$ and $g^{ab}$ achieved by a machine are dependent only on the transition function $f$, whereas $q_u^{ab}$ and $g_u^{ab}$ depend on both the functions $f$ and $d$. Also, for the 2-hypothesis case, i.e., $K = 2$, $q_u^{12} = \sigma^{12}$ and $q_u^{12} = g^{12}$. In fact, the only nontrivial condition for the $\varepsilon$-achievability of $P_L^\ast (= P^\ast)$, i.e., the ability to achieve for any arbitrary $\varepsilon > 0$ an error probability $P(e) \leq P_L^\ast + \varepsilon$, for $K = 2$, is the achievability of the maximum spread $\sigma^{12} = (\gamma_{12})^{m-1}$. Purely deterministic transition functions exist that achieve the maximum spread and then it is only necessary to suitably define the decision function and allow randomization in $f$ to $\varepsilon$-achieve $P_L^\ast$. However, for $K > 2$, introduction of $q_u^{ab}$ and $q_u^{ab}$ implies that a simultaneous optimization on both the functions $f$ and $d$ is needed to derive $P^\ast$. 
The difficulty of this optimization and the existence of additional constraints relating these parameters are the major stumbling blocks in determining $P^*$. That the characteristics mentioned above are not unique to the class of automata considered here follows from the theorems below.

**Theorem 3.1**

There exists no m-state automaton (specified by both f and d) that achieves the "useful" spreads $\sigma_{12}^u = (\gamma_{12})^{m-1}$, $\sigma_{23}^u = (\gamma_{23})^{m-1}$, and $\sigma_{13}^u = (\gamma_{13})^{m-1}$.

Proof. Suppose there exist automata that achieve the maximal useful spreads. The proof is by contradiction and follows from definitions (3.48), (3.49), and Lemmas 2.3 and 3.3. Lemmas 2.3 and 3.3 together imply that, for the hypothesis pair $(H_1, H_3)$, in order to achieve the maximal spread $\sigma_{13}^u = (\gamma_{13})^{m-1}$, the automata must involve transitions only to adjacent states with upward and downward transitions based on $X = H$ and $X = T$, respectively. Definition (3.48) and the assumption $\sigma_{13}^u = (\gamma_{13})^{m-1}$ imply that the decision rule must be such that the hypotheses $H_3$ and $H_1$ are the decisions made in states 1 and m, respectively. Similarly, the assumption $\sigma_{12}^u = (\gamma_{12})^{m-1}$ implies that the automata transition structure must be as above, but that decision in state 1 must involve hypothesis $H_2$. Thus, a contradiction develops. Hence the theorem. \(\square\)

Theorem 3.1 demonstrates certain limitations on m-state automata, and Theorem 3.2 warns about certain additional relationships, beside the inequalities, that might exist.
Theorem 3.2

For any m-state automaton that achieves the "useful" spreads

\[ g_{u}^{12} = (\gamma_{12})^{m-m_1}, \quad g_{u}^{23} = (\gamma_{23})^{m_1-1}, \quad \text{and} \quad g_{u}^{13} = (\gamma_{13})^{m-1} \]

for some \( 1 < m_1 < m \), the following relationship holds:

\[ g_{u}^{13} = g_{u}^{12} g_{u}^{23} \frac{1}{(\gamma_{12})^{m_1-1}}. \]  \hspace{1cm} (3.52)

Proof. As in the proof of Theorem 3.1, the automata must involve only transitions to adjacent states with transitions upwards on \( X = H \) and downwards on \( X = T \). Further, the form of useful spreads implies a decision rule involving the hypotheses \( H_3, H_2, \) and \( H_1 \) in states \( 1, m_1, \) and \( m \), respectively. Recall the notation \( \mu_{i}^{a}, a = 1, 2, 3, \) for the stationary occupancy probability for state \( i \) under the hypothesis \( H_a \). From the machine structure

\[ g_{u}^{13} = \frac{\mu_{1}^{1}}{\mu_{1}}, \]

\[ g_{u}^{23} = \frac{\mu_{2}^{1}}{\mu_{1}}, \]

\[ g_{u}^{12} = \frac{\mu_{1}^{1}}{\mu_{m_1}}, \]

and

\[ \frac{\mu_{m_1}^{1}}{\mu_{m_1}^{1}} = (\gamma_{12})^{m_1-1}. \]
Thus,

\[ q_{13} = \frac{1}{\mu_1^3} \]

\[ = \frac{\mu_1^1 \mu_2^1 \mu_{m1}^1 \mu_{m1}^2}{\mu_1^2 \mu_1^2 \mu_m^1 \mu_{m1}^1} \]

\[ = \frac{\mu_{m1}^1 \mu_1^2 \mu_{m1}^2}{\mu_{m1}^3 \mu_1^1 \mu_{m1}^2} \]

\[ = q_{12} q_{23} \frac{1}{(\gamma_{12})^{m1-1}}. \]

3.4 Close-to-Optimal Automata

Given the complexity of determining \( P^* \), the greatest lower bound on the probability of error, and a class of automata to \( \epsilon \)-achieve this bound for a \( K \)-hypothesis testing problem, one might be just interested in constructing sub-optimal or, in some sense, close-to-optimal automata. One way of measuring the closeness to optimality of different sub-optimal finite memory schemes is by comparing the additional memory requirement to match the performance of the optimal randomized \( m \)-state automaton. For a given \( K \)-hypothesis testing problem, let us suppose that the optimal randomized automaton with \( B \) bits of memory (or equivalently \( m \) states, \( m = 2^B \)) achieves a probability of error equal to \( P_m^* \). The subscript \( m \) in \( P_m^* \) is used to emphasize the fact that the optimal automaton uses only \( m \) states. Clearly, a
sub-optimal automaton using $B$ bits of memory will achieve an error probability $P(e) \geq P^*_m$. Presumably, if more memory is allowed for the sub-optimal scheme, it can achieve a probability of error equal to $P^*_m$. Let $b$ be the minimal number of bits of memory to be added to the sub-optimal automaton such that with $(B+b)$ bits it achieves an error probability equal to $P^*_m$. Then the quantity $b$, the additional memory requirement to match the performance of the optimal randomized $m$-state automaton can be used as a measure of closeness to optimality. Observe that $b$ can be a function of $m$. Further, in general, we are interested in sub-optimal automata for a class of $K$-hypothesis testing problems and not just any specific one. Thus, for any problem in this class there is an optimal automaton and a corresponding sub-optimal automaton specified by our design procedure. Over this class of problems, the quantity $b$, the additional memory required by our sub-optimal automaton to match the performance of the corresponding optimal $m$-state automaton, can now be a function of the particular problem parameters also.

To be specific, consider the class of 3-hypothesis testing problems on a Bernoulli observation space each one of the form (3.46). Since the difficulty in determining $P^*_m$ and a class of automata to $\epsilon$-achieve this bound has already been shown in Section 3.3, we are interested in sub-optimal automata. For a given 3-hypothesis problem and a sub-optimal scheme, the additional memory requirement is then a function of $m$ and the parameters $P_1, P_2$ and $P_3$. We note this dependency by expanding the notation for $b$ to $b(m, P_1, P_2, P_3)$. It is
clear that as long as the probability of error decreases monotonically and goes to zero as the memory size is increased for the sub-optimal scheme, \( b \) is finite for any \( m, p_1, p_2 \) and \( p_3 \). Also, Hellman (1972) has shown that \( P_m^* \) cannot go to zero faster than exponentially in \( m \). Hence, if the probability of error for the sub-optimal automaton goes to zero exponentially in the number of states then

\[
b_1(p_1, p_2, p_3) = \sup_m b(m, p_1, p_2, p_3)
\]

is also finite. In other words, for any 3-hypothesis testing problem there exists a finite quantity \( b_1 \) such that for any \( B \), the sub-optimal automaton with \( (B + b_1) \) bits in memory has an error probability equal to that of the optimal randomized rule with \( B \) bits. Thus, for large memory size \( B \), the additional memory requirement \( b_1 \), being finite, becomes negligible. Observe that as \( B \to \infty \), \( (b_1/B) \to 0 \). Such sub-optimal automata were called asymptotically optimal by Hellman (1972). He displayed several deterministic asymptotically optimal automata. However, \( b_1(p_1, p_2, p_3) \) is still a function of the problem parameters and can become arbitrarily large for some problems. In fact, for the class of deterministic automata specified by Hellman

\[
b^* = \sup_{(p_1, p_2, p_3)} b_1(p_1, p_2, p_3) = \infty.
\]

If, on the other hand, \( b^* \) is also finite, then the class of automata can be called close-to-optimal for the class of 3-hypothesis testing problems. We will demonstrate such automata for the case of Bernoulli observations in the next sub-section. Thus, the noteworthy point is
that the number of bits of memory to be added is not only finite and
independent of \( m \) but is also independent of problem parameters.

3.4.1 Close-to-Optimal Automata for Bernoulli Observations,

In this section we will demonstrate a class of automata for
3-hypothesis testing on Bernoulli observations, which requires at most
one extra bit of memory to match the performance of an optimal \( m \)-state
automaton, independent of \( m, p_1, p_2 \) and \( p_3 \). We will also remark on
the extensibility of our result to the general case, i.e., \( K > 3 \).

Figure 3.2 specifies a class of \((2m-1)\)-state automata for the
3-hypothesis problem. Again, the transitions upwards are all on \( H \)
while those downwards are on \( T \). The transitions away from states \( 1, m \)
and \((2m-1)\) involve artificial randomization. It can be checked that
for any \( k_1 \) and \( k_2 \), the randomization coefficients as in Figure 3.2,
the following equations are satisfied, as \( \delta \to 0 \).

\[
\begin{align*}
P_{12}P_{21} &= \tau_{12}^{-1} (1-P_{12}P_{13})(1-P_{21}P_{23})^2, \quad (3.53) \\
P_{23}P_{32} &= \tau_{23}^{-1} (1-P_{21}P_{23})(1-P_{31}P_{32})^2, \quad (3.54) \\
P_{13}P_{31} &= \tau_{13}^{-2} (1-P_{12}P_{13})(1-P_{31}P_{32})^2, \quad (3.55)
\end{align*}
\]

where the \( \tau_{ab} \) are as defined in (3.36).

A few cautionary remarks to avoid notational confusion might be
in order. The \( P_{ab} \) in (3.53) - (3.55) are the error probabilities for
the class of \((2m-1)\)-state machines in Figure 3.2. On the other hand,
the quantities \( P_{ab} \) in Section 3.2.1 and in what follows refer to the
values obtained by optimizing over constraints applicable to \( m \)-state
Fig. 3.2 A class of close-to-optimal automata for Bernoulli 3-hypothesis testing.
machines. What we shall show is that by suitably choosing $k_1$ and $k_2$ in Figure 3.2 and letting $\delta \to 0$, we can achieve an error probability strictly smaller than the lower bound $P_L^*$ for $m$-state machines, given by (3.37). In what follows, all the derivations correspond to letting $\delta \to 0$.

It is straightforward to derive that, for machines in Figure 3.2,

$$
\frac{P_{12}}{P_{11}} = k_1 \cdot \left(\frac{q_1}{P_1}\right)^{m-1},
$$

and

$$
\frac{P_{23}}{P_{22}} = k_2 \cdot \left(\frac{q_2}{P_2}\right)^{m-1}.
$$

Choose $k_1$ and $k_2$ such that

$$
\frac{P_{12}}{P_{11}} = \bar{P}_{12}/\bar{P}_{11},
$$

and

$$
\frac{P_{23}}{P_{22}} = \bar{P}_{23}/\bar{P}_{22}.
$$

From (3.33), (3.53) and (3.58)

$$
\frac{P_{21}}{P_{22}} = \bar{P}_{21}/\bar{P}_{22}.
$$

Similarly, from (3.34), (3.54) and (3.59)

$$
\frac{P_{32}}{P_{33}} = \bar{P}_{32}/\bar{P}_{33}.
$$

Further, it can be checked that for the class of machines under consideration,

$$
\frac{P_{13}}{P_{12}} = k_2 \cdot \left(\frac{q_1}{P_1}\right)^{m-1},
$$

which leads to

$$
\frac{P_{13}}{P_{11}} = \left(\frac{P_{13}}{P_{12}}\right)\left(\frac{P_{12}}{P_{11}}\right) = k_2 \cdot \left(\frac{q_1}{P_1}\right)^{m-1}\left(\frac{P_{12}}{P_{11}}\right).
$$
On the other hand, from (3.44), (3.57) - (3.59)

\[ \frac{P_{13}}{P_{11}} = \left( \frac{P_{13}}{P_{12}} \right) \left( \frac{P_{12}}{P_{11}} \right) = \left( \frac{P_{12}}{P_{11}} \right)^2 (q_2/p_2)^{m-1} \left( \frac{P_{12}}{P_{11}} \right). \]  

(3.63)

From (3.62), (3.63) and the fact that \( p_1 > p_2 \), we can infer

\[ \frac{P_{13}}{P_{11}} < \frac{P_{13}}{1 - P_{11}}. \]  

(3.64)

By similar analysis

\[ \frac{P_{31}}{P_{33}} < \frac{P_{31}}{1 - P_{33}}. \]  

(3.65)

From (3.58) and (3.64), (3.59) and (3.60), and (3.61) and (3.65) respectively

\[ P_{11} > \frac{P_{11}}{P_{12}} \quad P_{22} = \frac{P_{22}}{P_{21}} \quad P_{33} > \frac{P_{33}}{P_{32}}. \]  

(3.66)

Thus, as \( \delta \to 0 \)

\[ P(e) < \frac{P_L}{1 - P_L} \leq P^*, \]  

(3.67)

where \( P_L^* \) and \( P^* \) refer to \( m \)-state machines and \( P(e) \) is the error probability of the \((2m-1)\)-state machine under consideration. Thus the additional memory required by the sub-optimal automaton is at most one bit, independent of \( m, p_1, p_2 \) and \( p_3 \). It should be added that (3.56) - (3.59) combined with (3.38) and (3.41) lead to

\[ k_1 = \left( \frac{\tau_{12}^{\frac{1}{2}}}{\tau_{23}^{\frac{1}{2}}} - 1 \right) / \tau_{12} \left( 1 + \tau_{23}^{\frac{1}{2}} \right) \left( p_1/q_1 \right)^{m-1}, \]  

(3.68)

and

\[ k_2 = \left( \frac{\tau_{12}^{\frac{1}{2}} + 1}{\tau_{12}^{\frac{1}{2}} - 1} \right) \left( p_2/q_2 \right)^{m-1}. \]  

(3.69)
which are positive quantities, assuring realizability.

The case of unequal prior probabilities for the hypotheses admits of direct extension. Equations (3.44) and (3.45) will remain valid, but the values of $\overline{P}_{ab}$ given by (3.38) - (3.43) will be different, leading to different values for $k_1$ and $k_2$. Hence, in this case also for suitable $k_1$ and $k_2$, the automata are close to optimal within one bit of memory. Notice that to determine $k_1$ and $k_2$ we need to solve for $P_L^*$ which, as we have observed repeatedly, is fairly intractable. Also, we have tacitly assumed that the degenerate situation where the prior probabilities provide more information than can be gathered by any $m$-state automaton for resolving the hypotheses does not arise. Unlike the 2-hypothesis case, degenerate situations can arise in a variety of ways in 3-hypothesis testing. However, they can always be easily detected by testing the values $\overline{P}_{ab}$ of the error probabilities for the implicit conditions $0 < \overline{P}_{ab} < 1$. Under a degenerate situation, optimal decision rules require that one or more of the subset of states $S_1$, $S_2$, $S_3$ be null. Hence, the derivation of the inequality constraints and $P_L^*$ do not really hold and the values $\overline{P}_{ab}$ derived by Lagrange minimization violate the conditions $0 < \overline{P}_{ab} < 1$. Nondegeneracy will assure that the coefficients $k_1$ and $k_2$ for automata in Figure 3.2 are positive and hence realizable. Recall that for the case of equal prior probabilities, $k_1$ and $k_2$ are always positive and thus realizable.
3.4.2 Extension to K-Hypothesis Testing \( (K > 3) \)

We believe that the results of the previous Section 3.4.1 can be extended for \( K > 3 \). That is, a class of \( ((K-1)(m-1) + 1) \)-state automata can be constructed to achieve a probability of error smaller than the optimal \( m \)-state automaton for the K-hypothesis testing problem. A rigorous proof of the statement would require us to write down \( K(K-1)/2 \) inequality constraints and derive an expression for \( P_L^* \), as in (3.37). Clearly, this involves considerable algebraic manipulation, which we will avoid here. We will, however, consider an approximate solution and argue the extensibility of the result at least for large memory sizes.

Consider again the 3-hypothesis testing problem with equal prior probabilities. For large values of \( m \), the error probabilities are very small compared to unity and hence the inequalities (3.33) - (3.35) can be reduced to

\[
\begin{align*}
P_{12} P_{21} &\geq \tau_{12}^{-1} , \quad (3.70) \\
P_{23} P_{32} &\geq \tau_{23}^{-1} , \quad (3.71) \\
\text{and} \quad P_{13} P_{31} &\geq \tau_{13}^{-1} \quad (3.72)
\end{align*}
\]

resulting in

\[
P_L^* = \frac{2}{3} \left[ \tau_{12}^{-1} + \tau_{23}^{-1} + \tau_{13}^{-1} \right] . \quad (3.73)
\]
Now consider automata depicted in Figure 3.2 and set

\[ k_1 = \left( \frac{p_1 p_2}{q_1 q_2} \right)^{\frac{1}{2}} (m-1), \]  

and \[ k_2 = \left( \frac{p_2 p_3}{q_2 q_3} \right)^{\frac{1}{2}} (m-1). \]  

Notice that (3.73) - (3.75) are also obtainable from the corresponding expressions in previous sections by treating \( \tau_{ab} \) large compared to unity. Clearly, as \( \delta \to 0 \), \( P(e) \) is less than \( P^*_L \) given by (3.73), which, for large memory sizes, is less than \( P^* \). Observe that the transition and the decision functions in states \( 1 \) to \( m \), and \( m \) to \((2m-1)\) now correspond to the \( \epsilon \)-optimal automata for resolving \( H_3 \) vs. \( H_2 \) and \( H_2 \) vs. \( H_1 \), respectively.

More generally, for the \( K \)-hypothesis problem \( K(K-1)/2 \) inequality constraints can be written down, resulting in, for large memory,

\[ P^*_L = \frac{2}{K} \sum_{a<b} \tau_{ab}^{-1}, \]  

where

\[ \tau_{ab} = (\gamma_{ab})^{m-1} = \left( \frac{p_a q_b}{p_b q_a} \right)^{m-1}. \]

The class of \(((K-1)(m-1)+1)\)-state automata for \( K \)-hypothesis testing will also involve transitions upward and downward only to adjacent states on \( X = H \) and \( X = T \), respectively. Further, the \((K-1)\) randomization coefficients required are given by

\[ k_a = \left( \frac{p_a p_{a+1}}{q_a q_{a+1}} \right)^{\frac{1}{2}} (m-1), \quad a = 1, 2, \ldots, K-1. \]

Then for large memory, as \( \delta \to 0 \), \( P(e) < P^*_L < P^* \), thus demonstrating
the extension of the result for \( K > 3 \), at least for large \( m \). For small values of \( m \) and unequal prior probabilities the required randomization coefficients will be different from those given by (3.78).

The success in constructing close-to-optimal automata for the Bernoulli case, even in the absence of complete knowledge of \( P^* \), can be attributed to the fact that the hypothesis pairs \( (H_a, H_b) \), \( a, b \in \{1, 2, \ldots, K\}, a < b, \) all have the supremum and the infimum of their likelihood ratios occurring at \( X = H \) and \( X = T \), respectively. Thus all transitions of automata in Figure 3.2, for example, occur on high likelihood ratio events. For the general \( K \)-hypothesis testing problem, however, the construction of neither the optimal nor close-to-optimal automata is known. In the next section, we will consider the symmetric hypothesis testing problem and demonstrate that the symmetric nature of the hypotheses allow us the construction of close-to-optimal automata.

3.5 Symmetric Hypothesis Testing

The class of symmetric hypothesis testing problems, as presented below, was originally introduced in the finite memory context by Sagalowicz (1970). This involves situations where the pair-wise hypothesis testing problems \( (H_a, H_b), a, b \in \{1, 2, \ldots, K\}, a \neq b, \) exhibit certain symmetry. Sagalowicz only attempted to construct the best \( K \)-state symmetric automata for \( K \)-hypothesis testing. Our interest here will be to exploit the symmetric nature of the problem to present close-to-optimal automata in the sense developed in Section 3.4.
Consider the following K-hypothesis testing problem. $X$ is a $d$-dimensional random vector, $d \geq K-1$, distributed according to a probability measure $P$. The hypothesis $H_i$ is $P = P_i^*$, $i = 1, 2, \ldots, K$, where $P_i^*$ are spherically symmetric around the points $M_i$ located at the vertices of a regular simplex. The hypotheses have equal prior probabilities.

Due to the symmetry of the problem, the quantities $\gamma_{ab}$, $a < b$, defined in (3.8) turn out to be equal and thus can be set to a constant $\gamma$. Further, it can be shown that

$$\overline{L}_{ab} = \frac{1}{L_{ab}} = \gamma^4,$$

for $a, b \in \{1, 2, \ldots, K\}$, $a < b$. The minimization of (3.20) subject to (3.23), in this symmetric problem, results in the lower bound

$$P_L^* = \left[ 1 + \frac{1}{K-1} \gamma^{4(m-1)} \right]^{-1}.$$

(3.79)

Note that for this problem $P_L^*$ is the same as $P_{LL}^*$ as defined in (3.24).

We will demonstrate a class of automata that requires

$$\binom{K}{2}m - K^2 + 2K$$

states to $\epsilon$-achieve this bound. Since $P_L^*$ is not even a tight lower bound, this class of automata loses at most

$$\log_2 \left( \binom{K}{2} \right)$$

bits compared to the optimal randomized automaton, independent of $m$ or the problem parameter $\gamma$. Thus this class of automata is close-to-optimal within $\log_2 \left( \binom{K}{2} \right)$ bits of memory.

For simplicity, let $K = 3$ in what follows. Further, let $i, j, v, w, a, b \in \{1, 2, 3\}$, $i \neq j$, $v \neq w$, $i \neq a$, $j \neq a$, $v \neq b$ and $w \neq b$. Let $A_{ij}$ ($c$) be sets such that
Further, for each $\epsilon$, the sets $A_{ij}(\epsilon)$ should satisfy the following relationships:

$$\lim_{\epsilon \to 0} \frac{Pr(A_{ij}(\epsilon)|H_j)}{Pr(A_{ij}(\epsilon)|H_i)} = \gamma.$$  

This can always be done.

Sagalowicz (1970) considered a class of 3-state automata, as shown in Figure 3.3, where the automata make a transition from state $i$ to state $j$ if $X(A_{ij}(\epsilon))$, and stay in the same state otherwise. He showed that this class of automata achieves, as $\epsilon \to 0$,

$$P(\epsilon) = \left[ 1 + \frac{1}{2} \gamma^2 \right]^{-1}. \tag{3.84}$$

He further showed that there exists no 3-state symmetric automaton that achieves a probability of error smaller than that given in (3.84).

Consider a class of symmetric automata that requires $(3m-3)$ states as in Figure 3.4, which, again, omits self loops. Note that the transition scheme resembles that of the automata considered by Sagalowicz. Further, note the randomization in the transitions away from the states labeled $1,m$ and $(2m-1)$ where the decision made corresponds to the hypotheses $H_1$, $H_2$ and $H_3$, respectively. Due to the fact that as the randomization parameter $\delta \to 0$ the asymptotic state probabilities...
Fig. 3.3 A class of automata considered by Sagalowicz.
Fig. 3.4 A class of sub-optimal automata for symmetric hypothesis testing.
are concentrated in the states \(1, m\) and \((2m-1)\), it is not critical what decision function is used on the other states.

In the limit that \(\delta\) and \(\varepsilon\) approach 0, it can be seen that

\[
P(\varepsilon) = \left[1 + \frac{1}{2} \gamma^{2(m-1)}\right]^{-1}.
\]

(3.85)

More generally, for the symmetric K-hypothesis problem a class of automata that requires \(\binom{K}{2}m - K^2 + 2K\) states can be constructed to yield an error probability which in the limit \(\delta \to 0\) and \(\varepsilon \to 0\) is

\[
P(\varepsilon) = \left[1 + \frac{1}{K-1} \gamma^{2(m-1)}\right]^{-1}.
\]

(3.86)

The reader may have noticed that we needed only the fulfillment of equations (3.80) - (3.83) in order to construct the sub-optimal automata of interest. In fact, it can be shown that for such automata to be constructed, it is sufficient that sets satisfying (3.80) - (3.83) can be defined for the K-hypothesis problem. Consider, for example, a discrete random variable \(X\) with possible values \(a, b,\) and \(c\). Let \(Pr(X=a|H_1) = Pr(X=b|H_2) = Pr(X=c|H_3) = 0.6\), and \(Pr(X=b|H_1) = Pr(X=c|H_1) = Pr(X=b|H_2) = Pr(X=c|H_2) = Pr(X=a|H_3) = Pr(X=b|H_3) = 0.2\). Now define the input sets as follows.

\[
A_{21}(\varepsilon) = \{a\} = A_{31}(\varepsilon),
\]

\[
A_{12}(\varepsilon) = \{b\} = A_{32}(\varepsilon),
\]

and \(A_{13}(\varepsilon) = \{c\} = A_{23}(\varepsilon)\).
Clearly, the conditions given by the equations (3.80) - (3.83) are satisfied, the probability of error given by (3.85) holds, and hence Figure 3.4 describes the desired automata.

3.6 Discussion

The design of optimal finite memory procedures for the general K-hypothesis problem has resisted a solution since 1970 when Hellman and Cover solved the 2-hypothesis case. The results of this chapter imply that the reason the techniques of the 2-hypothesis case resist extension to multiple hypothesis case is related to the multiplicity of constraints in realizable automata for the latter case, and the difficulty of incorporating all of them in a set of inequalities satisfied by the error probabilities. We introduced the notion of close-to-optimality and demonstrated its utility.

We believe that this notion of close-to-optimality has the potential of making finite memory theory applicable to many problems which are currently intractable. Cover (1970) considered the finite memory estimation problem and showed that it can be broken down into two parts, one involving multiple hypothesis testing and the other a quantization problem. Hence, the results of this chapter suggest that one might look for only close-to-optimal solutions for the estimation problem also. Chapters IV and V deal with compound hypothesis testing and the two-armed bandit problem, respectively. Several close-to-optimal deterministic automata are presented there. Problems which are currently intractable and for which the notion of close-to-optimality can be applied to obtain reasonable sub-optimal solutions include multiple
compound hypothesis testing and the many-armed bandit problem. These are basically extensions of the problems considered in the next two chapters.
CHAPTER IV

COMPOUND HYPOTHESIS TESTING

The hypothesis testing problems studied in Chapters II and III involved simple hypotheses, i.e., each hypothesis specified one parameter or distribution. On the other hand, in compound hypothesis testing, the hypotheses are allowed to specify a set of possible parameter values or distributions. In this chapter, we study compound hypothesis testing on a Bernoulli observation space, with the added constraint of finite memory. We will see that the basic issue to be resolved is the optimality principle to be used: the Bayesian approach will be found to be both complex and unnatural, and the minimax principle will need to be supplemented to avoid the inclusion of trivial machines as optimal solutions. In the next section, we start with a formal definition of the compound hypothesis testing problem.

4.1 Definition of the Problem

Let \( X_1, X_2, \ldots \) be a sequence of independent, identically distributed Bernoulli random variables with possible values \( H \) and \( T \) such that \( \Pr(X_i = H) = p \) and \( \Pr(X_i = T) = 1-p = q \). In Section 2.3 we considered the hypothesis testing problem of the form,

\[
H_1: \ p = p_1 \ \text{vs.} \ \ H_2: \ p = p_2,
\]

where \( 1 > p_1 > p_2 > 0 \). Observe that under either hypothesis the
parameter \( p \) can assume only one possible value. In other words, the Bernoulli distribution from which we draw the random sample is completely specified under either hypothesis. Such hypotheses are called simple hypotheses. Further, it is assumed that \( p_1 \) and \( p_2 \) are the only two possible values for the parameter \( p \). Consider, on the other hand, a hypothesis testing problem of the form,

\[
\begin{align*}
H_1: & \quad p \geq p_1 \quad \text{vs.} \quad H_2: \quad p \leq p_2, \\
\end{align*}
\]

where \( 1 \geq p_1 \geq p_2 > 0 \), and it is assumed that \( p \) cannot lie in the open interval \((p_2, p_1)\). Each hypothesis in (4.1) specifies a (nonsingleton) set of possible values for the parameter \( p \). Such hypotheses are called compound hypotheses and (4.1) is an example of the class of problems called compound hypothesis testing problems.

As in earlier chapters, we are interested in designing finite memory decision devices for this problem. For completeness, we repeat the notation: the decision maker is a finite-state stochastic automaton \( \alpha \), with inputs \( X_n \), outputs \( d_n \) and state space \( S = \{1, 2, \ldots, m\} \) such that \( T_n \), the state of the automaton at time \( n \), changes according to the rule,

\[
T_n = f(T_{n-1}, X_n), \ T_n \in S, \quad (4.2)
\]

and decision \( d_n \) of the automaton at time \( n \) is

\[
d_n = d(T_n), \ d_n \in \{H_1, H_2\}, \quad (4.3)
\]

where functions \( f \) and \( d \) are allowed to be stochastic. A state
$T_0$ is designated the initial state, and $f$ and $d$ describe the transition and output functions, respectively. The number of states, $m$, is a measure of the memory size of the automaton. It corresponds to $\log_2 m$ bits. Again, we will not allow the transition and decision function to be dependent on the time instant $n$ or the past data $X_1, X_2, \ldots, X_{n-1}$ except for what is summarized through the statistic $T_{n-1}$. That is, the functions $f$ and $d$ will be both time- and data-invariant.

For any given $p$ and an automaton $a$ the probability of error is defined to be

$$P_e(p,a) = E\left(\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} e_i \right), \quad (4.4)$$

where $e_i = 0$ or $1$ according as $d_i$ is the correct decision or not.

(The above expanded and slightly modified notation for the probability of error emphasizes its dependency on the parameter value $p$ and the automaton $a$. Such a change is necessary to aid a proper understanding of the results to be presented.) Recall from the statement of the problem that $p$ cannot assume a value in the open interval $(p_2, p_1)$. Hence, for $p \geq p_1$ ($p \leq p_2$) a decision $d_i = H_2(d_i = H_1)$ at time instant $i$ is in error. The expectation in (4.4) needs to be taken only if the automaton has absorbing states or classes of states. In what follows, if we restrict attention to irreducible automata the expectation operation will not be performed. Again, we need not even specify $T_0$, the start state for such automata.
A Bayesian approach will require us to treat the unknown \( p \) as a random variable with a known prior distribution and choose \( f \) and \( d \), the transition and decision functions, so as to minimize the average probability of error. On the other hand, if one adopts the minimax principle, the objective is to choose \( f \) and \( d \) so that the error probability satisfies

\[
P^* = \inf_A \sup_P P_e(p, a). \tag{4.5}
\]

The minimaxing operation in (4.5) can be viewed as finding first, for each automaton,

\[
P^*_C(a) = \sup_P P_e(p, a), \tag{4.6}
\]

and then choosing the infimum of \( P^*_C(a) \) over all \( m \)-state automata.

In this chapter, we use the minimax principle throughout. It is worth emphasizing that in the supremum operation in (4.5) and (4.6), \( p \) cannot take values in the interval \((p_2, p_1)\). We will hereafter assume this qualification without explicit restatement.

The quantity \( P^*_C \), defined in (4.5), is called the minimax probability of error for this problem. As will be seen later, \( P^*_C \) is nonzero for this problem, i.e.,

\[
\inf_A \sup_P P_e(p, a) \neq 0.
\]

In contrast,

\[
\sup_P \inf_A P_e(p, a) = 0.
\]
This is because, for any given value $p'$ of the parameter $p$, there always exists at least one automaton that achieves a zero error probability. For example, if $p' > p_1$, the trivial automaton $\alpha$, that always decides hypothesis $H_1$, achieves a zero error probability. However, these automata do not show such a good performance if the value of $p$ is varied from $p'$.

At this point we want to pause and make some observations on the choice of the minimax principle here. If there exists an automaton that achieves a zero probability of error for every value of the parameter $p$, then there is no need for using the Bayesian or minimax principle. While Cover (1969) has demonstrated such time-varying automata, a time-invariant automaton that achieves a zero error probability for every value of $p$ does not exist. Also, among all $m$-state automata, there does not exist an automaton that achieves the smallest probability of error for every value of $p$. Hence, it is necessary to adopt the Bayesian or minimax or some other suitable approach to defining optimality. We recall that the operating characteristic, relating the probabilities of error of the two kinds, for the 2-hypothesis case—(2.27) in Section 2.3.1—was derived completely independent of any Bayesian or minimax formulation. However, the greatest lower bound on the probability of error was derived using a Bayesian approach. That is, we assumed prior probabilities for the hypotheses and minimized the expected asymptotic proportion of errors. In Chapter III also we assumed a Bayesian approach in studying the multiple hypothesis testing problem. Such an approach to this compound hypothesis testing problem
will require us to assume a prior distribution on \( p \) and not just prior probabilities on the hypotheses. But the minimization of expected asymptotic proportion of errors leads to a complicated optimization problem. Further, it is seldom that enough information about the unknown parameter is available to assume a prior distribution and lacking such knowledge any arbitrary assumption appears to be unreasonable. Hence, we will adopt the minimax principle in studying the compound hypothesis testing problem. Basically, the objective would be to design the best machine for the worst case. The minimax principle is not, however, without its own set of difficulties. We will observe in Section 4.3 that there exist compound hypothesis testing problems for which the minimax principle is too inclusive, and trivial automata can be exhibited that are optimal in the minimax sense. This situation arises if the worst case is so hard that no improvement is possible over a random machine, which thus becomes minimax optimal. Hence, in order to disqualify trivial solutions, it is necessary to demonstrate certain additional optimality properties besides the minimax optimality.

Our approach to compound hypothesis testing is to view the problem in terms of a collection of simple hypothesis testing problems. This viewpoint is made precise in Section 4.3. It is thus necessary to first study the simple hypothesis testing problem in the context of the minimax principle.
4.2 Simple Hypothesis Testing in a Minimax Formulation

In what follows we adapt the results of Section 2.3.1 for the case of Bernoulli observations using the minimax principle. Consider a simple 2-hypothesis testing problem of the form

\[ H_1: p = p_1 \ vs. \ H_2: p = p_2, \quad (4.7) \]

where \( 1 > p_1 > p_2 > 0 \). Let \( \alpha \) and \( \beta \) denote the probabilities of error of the two kinds achieved by the automaton \( a \), i.e.,

\[ \alpha = P_e(p_1, a) , \quad (4.8) \]

\[ \beta = P_e(p_2, a) . \quad (4.9) \]

Observe that the definition of \( P_e(p, a) \) in (4.5) was originally in the context of the compound hypothesis testing problem (4.1). Yet, the error probabilities, \( \alpha \) and \( \beta \), for the simple hypothesis case can be expressed as above in terms of \( P_e(p, a) \). This is because the hypothesis \( H_1(H_2) \) is true in the compound problem (4.1), if the hypothesis \( H_1(H_2) \) is true in the simple hypothesis testing problem considered here and the automaton \( a \) can be viewed as solving the compound problem (4.1).

As observed earlier, the operating characteristic for any \( m \)-state automaton solving a 2-hypothesis testing problem was derived in Section 2.3.1, and this derivation was independent of the optimality principle used. Hence, it applies equally well for the minimax formulation adopted here. Rewriting (2.27) in terms of the notations of this chapter,
where
\[ \gamma(p_1, p_2) = \frac{q_2}{p_2q_1} \]
and
\[ q_1 = 1 - p_1 \text{ and } q_2 = 1 - p_2. \]

Now define
\[ \mathbb{P}_S(a) = \max (\alpha, \beta). \]  

The subscript \( S \) in \( \mathbb{P}_S(a) \) is used to refer to the fact that we are dealing with a simple hypothesis testing problem. Using the minimax principle leads to the objective of minimizing \( \mathbb{P}_S(a) \) over all \( m \)-state automata. Minimizing (4.12) subject to (4.10) yields a lower bound on \( \mathbb{P}_S(a) \) as
\[ \left[ 1 + \gamma(p_1, p_2)^{\frac{1}{4}}(m-1) \right]^{-1}. \]
There exist automata, as will be presently seen, that achieve \( \mathbb{P}_S(a) \) arbitrarily close to this lower bound. Hence, denoting the infimum over all \( m \)-state automata by
\[ \mathbb{P}_S^* = \inf_a \mathbb{P}_S(a), \]
for any automaton \( a, \mathbb{P}_S(a) \geq \mathbb{P}_S^* \) where
\[ \mathbb{P}_S^* = \left[ 1 + \gamma(p_1, p_2)^{\frac{1}{4}}(m-1) \right]^{-1}. \]

Arguments similar to those in Section 2.3.1 can be presented to show that there exists no automaton that achieves \( \mathbb{P}_S^* \). An automaton \( a \) is said to be minimax optimal if \( \mathbb{P}_S(a) = \mathbb{P}_S^* \). Thus, minimax optimal automata do not exist for this problem. An \( \varepsilon \)-minimax optimal (or \( \varepsilon \)-minimax for short) class of automata can, however, be constructed.
That is, for any $\epsilon > 0$, there exists an automaton $a_\epsilon$ in this class for which $P_S(a_\epsilon) \leq P_S^* + \epsilon$. Figure 4.1 depicts one such optimal class, where $k$ should be set equal to

$$k^* = \left(\frac{p_1p_2}{q_1q_2}\right)^{1/(m-1)}.$$  \hspace{1cm} (4.15)

Basically, the transitions are between adjacent states, input $X = H$ leading towards the higher numbered states and $X = T$ leading towards lower numbered states. The decision function is as follows: decide the hypothesis $H_2$ in state 1 and the hypothesis $H_1$ in state $m$, deterministically; decide hypothesis $H_1$ or $H_2$ with equal probability in states 2 to $(m-1)$. The transitions away from the extreme states 1 and $m$ involve artificial randomization. If in state 1, the machine transits to state 2 on input $X = H$ with a small probability $0 < \delta < 1$. If in state $m$, it transits to state $m-1$, on $X = T$ with probability $k\delta$. The other transitions do not involve any randomization.

When $k$ is set to $k^*$ given by (4.15), and as $\delta \to 0$, the probability of error approaches $P_S^*$. As $\delta \to 0$, the stationary occupancy probabilities are concentrated in the extreme states, 1 and $m$, under either hypothesis. Thus, the actual decision rule in states 2 to $(m-1)$ is not crucial for proving the $\epsilon$-achievability of $P_S^*$. However, this particular randomized decision scheme is chosen to facilitate the proof of a result in Section 4.3.

It is worth recalling that in Chapter II, instead of minimizing (4.12), we minimized
In states 2 to (m-l) decision H stands for $H_1$ or $H_2$ with equal probability.

Fig. 4.1 An ε-minimax class of automata. Self transitions are deleted.
where \( \pi_1 \) and \( \pi_2 \) were the prior probabilities of the two hypotheses, i.e., we adopted the Bayesian approach. The structure of one class of optimal machines derived there was the same as that of Figure 4.1, but the value for the optimal randomization coefficient \( k^* \) would be different for different values of the prior probabilities. For the particular case \( \pi_1 = \pi_2 = \frac{1}{2} \), the expression for the optimal randomization coefficient \( k^* \) coincides with that in (4.15). Hence, we observe that the optimal minimax automaton is identical to the optimal Bayes automaton for the above prior distribution. This result can also be shown by a different approach. The following result, applicable to \( m \)-state automata solving a 2-hypothesis testing problem, has been shown by Shubert (1974b):

\[
\inf_{\pi} \sup_a P_{\text{Bayes}}(a) = \sup_{\pi_1, \pi_2} \inf_a P_{\text{Bayes}}(a). \tag{4.17}
\]

Notice that

\[
\sup_{\pi_1, \pi_2} P_{\text{Bayes}}(a) = \max(\alpha, \beta), \tag{4.18}
\]

thus making the left hand side of (4.17) equal to \( P_S^* \). By considering the expression for the probability of error for the Bayes approach given in (2.28) it can be readily shown that the least favorable prior distribution for the right hand side of (4.17), i.e., the prior probabilities for the hypotheses that maximize the infimum on \( a \) of \( P_{\text{Bayes}}(a) \), is given by \( \pi_1 = \pi_2 = 1/2 \). Thus, the optimal minimax
automaton is identical to the optimal Bayes automaton for this prior
distribution.

4.3 Compound Hypothesis Testing

We are now ready to consider compound hypothesis testing of
the form (4.1), i.e.,

\[ H_1: \ p \geq p_1 \quad \text{vs.} \quad H_2: \ p \leq p_2, \]

where \( 1 > p_1 > p_2 > 0 \). For notational simplicity, let

\[ \alpha(p) = P_e(p, a), \quad \text{if} \quad p \geq p_0 \tag{4.19} \]

and

\[ \beta(p) = P_e(p, a), \quad \text{if} \quad p \leq p_1. \tag{4.20} \]

Further, let

\[ \bar{\alpha} = \sup_{p \geq p_0} \alpha(p) \tag{4.21} \]

and

\[ \bar{\beta} = \sup_{p \leq p_1} \beta(p). \tag{4.22} \]

From (4.6),

\[ P_c(a) = \sup_p P_e(p, a) = \max (\bar{\alpha}, \bar{\beta}). \tag{4.23} \]

The objective is to determine an m-state automaton, i.e., a pair of
transition and decision functions, so as to minimize \( P_c(a) \). The
approach we take is to view the compound hypothesis testing problem in
terms of a suitable collection of simple hypothesis testing problems
and thus express \( P_c(a) \) in terms of the various \( P_s(a) \), the
probabilities of error applicable to the simple hypothesis testing problems in that collection.

Now, consider a collection of simple hypothesis testing problems, each member in the collection of the form

\[ H_1: \ p = p_1' \ vs. \ H_2: \ p = p_2' , \]  

(4.24)

where \( p_1' \geq p_1 \) and \( p_2' \leq p_2 \). In order to emphasize the dependence of \( P_S(a) \) on \( p_1' \) and \( p_2' \) in a problem of the form (4.24), we expand the notation to \( P_S(a; p_1', p_2') \).

The following lemma is straightforward.

**Lemma 4.1.**

\( P_C(a) \) given by (4.23) is equal to the supremum of \( P_S(a; p_1', p_2') \) over the collection of simple hypothesis testing problems of the form (4.24), i.e.,

\[ P_C(a) = \sup_{(p_1', p_2')} P_S(a; p_1', p_2') . \]  

(4.25)

This lemma thus allows us to view the compound hypothesis testing problem in terms of a collection of simple hypothesis testing problems.

From Section 4.2,

\[ P_S(a; p_1', p_2') \geq \left[ 1 + \gamma(p_1', p_2') \right]^{(m-1)} - 1 . \]

This leads to

\[ \sup_{(p_1', p_2')} P_S(a; p_1', p_2') \geq \left[ 1 + \gamma(p_1', p_2') \right]^{(m-1)} - 1 , \]
where $\gamma(p_1, p_2)$ is the resolvability measure given in (4.11).

Consideration of the class of automata in Figure 4.1 with

$$k = k^* = (p_1p_2/q_1q_2)^{1/(m-1)}$$

and in the limit $\delta \to 0$, yields,

$$P_c^* = \inf_a \sup_{(p_1', p_2')} P_S(a; p_1', p_2') = \left[1 + \gamma(p_1, p_2)\right]^{-1}. \quad (4.26)$$

$P_c^*$ is the minimax value for the compound hypothesis testing problem under consideration, i.e., for the problem

$$H_1: p \geq p_1 \text{ vs. } H_2: p \leq p_2.$$

Observe that $P_c^*$ for this problem is precisely the same as $P_S^*$ derived in Section 4.2 for the simple hypothesis testing problem

$$H_1: p = p_1 \text{ vs. } H_2: p = p_2.$$

Further, the above simple hypothesis testing problem belongs to the collection of simple hypothesis testing problems generated by the compound problem. We also observed in Section 4.2 that there exists no automaton that achieves $P_S^*$ and that only an $\epsilon$-minimax class of automata can be constructed. As a result, $P_c^*$, the greatest lower bound on $P_c(a)$, cannot be achieved by any automaton. An automaton $a$ is minimax if $P_c(a) = P_c^*$. Thus, minimax automata do not exist for this problem. However, a class of $\epsilon$-minimax automata can be constructed, as in Figure 4.1. That is, for every $\epsilon > 0$, there exists
an automaton \( a_e \) in this class for which \( P_C(a_e) < P_C^* + \epsilon \).

The following minimax theorem, satisfied by the collection of simple hypothesis testing problems, will be of use in later development.

**Theorem 4.1**

\[
\inf_a \sup_{(p'_1, p'_2)} P_S(a; p'_1, p'_2) = \sup_{(p'_1, p'_2)} \inf_a P_S(a; p'_1, p'_2). \tag{4.27}
\]

**Proof.** From Section 4.2,

\[
\inf_a P_S(a; p'_1, p'_2) = \left[ 1 + \gamma(p'_1, p'_2)^{1/(m-1)} \right]^{-1}.
\]

Also

\[
\inf_{(p'_1, p'_2)} \gamma(p'_1, p'_2) = \gamma(p_1, p_2). \tag{4.28}
\]

Hence

\[
\sup_{(p'_1, p'_2)} \inf_a P_S(a; p'_1, p'_2) = \left[ 1 + \gamma(p_1, p_2)^{1/(m-1)} \right]^{-1}. \tag{4.29}
\]

The theorem follows by comparing (4.26) and (4.29).

Notice that, if problem (4.1) is modified as follows,

\[
H_1: p > p_1 \quad \text{vs.} \quad H_2: p < p_2,
\tag{4.30}
\]

where \( 1 > p_1 > p_2 > 0 \), \( P_C^* \) remains the same as in (4.26), and the automata in Figure 4.1, with \( k \) as given by (4.15) are still \( \epsilon \)-minimax.

However, since the simple hypothesis testing problem,

\[
H_1: p = p_1 \quad \text{vs.} \quad H_2: p = p_2,
\]

does not belong to the collection of simple problems generated by the
compound problem (4.30), we cannot assert that there does not exist a minimax optimal automaton. In fact, for the particular case of (4.30) corresponding to $p_1 = p_2 = p_0$, it is fairly straightforward to exhibit minimax optimal automata.

Also, the hypothesis testing problem

$$H_1: p > p_0 \quad \text{vs.} \quad H_2: p < p_0,$$

(4.31)

where $1 > p_0 > 0$, exemplifies an important issue regarding the usage of the minimax principle. This is considered below. The minimax value for this problem, as seen from (4.26), is $P_C^* = 1/2$. Hence, an automaton $a$ is minimax optimal if $P_C(a) = 1/2$, i.e., $P_e(p,a) \leq 1/2$, for all possible values of $p$, $p = p_0$ being excluded. An automaton $a$ is said to be $\varepsilon$-minimax if $P_C(a) \leq (1/2) + \varepsilon$. Consider an automaton, say $a_R$, which chooses randomly between the hypotheses. Clearly

$$P_e(p,a_R) = 1/2, \text{ for all } p,$$

and hence $a_R$ is minimax optimal. On the other hand, the automata depicted in Figure 4.1, with $k$ set to $(p_0/q_0)^{m-1}$, are not only minimax optimal but are also superior to $a_R$, for all possible values of $p$, and hence are preferable over $a_R$. (See Appendix B for the proof of the fact that automata in Figure 4.1 are minimax optimal for the problem (4.31), and not just $\varepsilon$-minimax.) Thus, for problem (4.31), the minimax principle is too inclusive—even the trivial automaton $a_R$ is minimax optimal. Hence, it is necessary to demonstrate certain additional optimality properties for the automata we
exhibit, besides the minimax optimality. Theorem 4.2 makes this idea precise, derives the lower bound on the probability of error achievable for any \( p \) by automata that are minimax optimal and shows that the error probabilities achieved by our minimax automata in Figure 4.1 can be made arbitrarily close to this bound.

**Theorem 4.2**

For any automaton that is minimax optimal,\(^{107}\)

\[
\alpha(p) \geq \left[1 + \left(\frac{pq_0}{p_0q}\right)^{m-1}\right]^{-1}, \tag{4.32}
\]

\[
\beta(p) \geq \left[1 + \left(\frac{qp_0}{pq_0}\right)^{m-1}\right]^{-1}. \tag{4.33}
\]

Further, the error probabilities of the minimax automata in Figure 4.1 with \( k = (p_0/q_0)^{m-1} \) approach these lower bounds for every value of \( p \) as \( \delta \to 0 \).

**Proof.** For any value of \( p > p_0 \) and every value of \( p' < p_0 \), from the operating characteristic (4.10),

\[
\alpha(p)\beta(p') \geq \{\gamma(p, p')\}^{(m-1)} (1-\alpha(p)) (1-\beta(p')). \tag{4.34}
\]

By the assumption of minimax optimality, \( \beta(p') \leq 1/2 \) and hence \( \beta(p')/(1-\beta(p')) \leq 1 \). Therefore, for every value of \( p' < p_0 \),

\[
\alpha(p) \geq \left[1 + \left(\frac{pq'}{p'q}\right)^{m-1}\right]^{-1},
\]

i.e.,

\[
\alpha(p) \geq \left[1 + \left(\frac{pq_0}{p_0q}\right)^{m-1}\right]^{-1}.
\]

Bound (4.33) can be derived by a similar analysis.
The fact that the automata in Figure 4.1 ε-achieve these bounds, in the sense that the bounds are attained as $\delta \to 0$, can be checked by a straightforward analysis. Denote by $\mu_i(p)$ the stationary probability of state $i$, for any $p$. Clearly, for any $\delta$,

$$\frac{\mu_1(p)}{\mu_m(p)} = k_1 \left( \frac{q/p}{m-1} \right) = \left( \frac{q_0/pq_0}{m-1} \right),$$

and as $\delta \to 0$, $\mu_1(p) + \mu_m(p) = 1$.

Further, for any $p > p_0$, and as $\delta \to 0$

$$\alpha(p) = \frac{\mu_1(p)}{\mu_1(p) + \mu_m(p)} = \left[ 1 + \left( \frac{\mu_m(p)}{\mu_1(p)} \right) \right]^{-1},$$

and

$$= \left[ 1 + \left( \frac{pq_0/pq_0}{m-1} \right) \right]^{-1}.$$ 

Again, a similar analysis shows the ε-achievability of bound (4.33).

It must be stressed again that while there exist minimax automata for this problem, there does not exist an automaton that will achieve the lower bounds for all values of $p$. The bounds can only be ε-achieved, in the sense that automata can be presented that achieve error probabilities arbitrarily close to the bound, for any value of $p$. Further, similar bounds can be derived for the general case (4.1) or (4.30).

We conclude this section with a discussion of deterministic machines for compound hypothesis testing. Sagalowicz (1970) considered the problem of the form (4.31). He constructed a particular class of automata that are generalizations of the saturable counter structure.
mentioned in Section 2.3. A saturable counter allows only transitions to adjacent states. The automata considered by Sagalowicz, however, allowed transitions that are not necessarily restricted to adjacent states. A complete analysis of the performance could not be made by Sagalowicz, who therefore only conjectured that the class of automata achieves a zero probability of error as the number of states tends to infinity. In what follows, we construct a class of deterministic automata for problem (4.30) (of which problem (4.31) is a special case),

\[ H_1: p > p_1 \quad \text{vs.} \quad H_2: p < p_2, \]

and show that these automata achieve zero error probabilities as \( m \to \infty \).

The transition and decision functions of these automata are given in Figure 4.2, with self-loops deleted. A simple analysis yields

\[
\alpha(p) = \left[1 + p^{m'} / q^{m-m'}\right]^{-1},
\]

and

\[
\beta(p) = \left[1 + q^{m-m'} / p^{m'}\right]^{-1}.
\]

This machine achieves zero probability of error under either hypothesis as \( m \to \infty \) if \( m' \) is adjusted such that

\[
\left[1 + \ln p_2 / \ln q_2\right]^{-1} \leq m' / m \leq \left[1 + \ln p_1 / \ln q_1\right]^{-1}.
\]

The optimal value of \( m' \) is given by

\[
m'/m = \left[1 + \ln(p_1 p_2) / \ln(q_1 q_2)\right]^{-1}.
\]
Fig. 4.2 A class of asymptotically optimal deterministic automata.
Also, the probability of error goes to zero exponentially as \( m \to \infty \). This fact guarantees that there exists a finite \( b^* \), given \( p_1 \) and \( p_2 \), such that the deterministic machine with the addition of \( b^* \) extra bits of memory, has a lower error probability, for all \( p \), than the optimal \( m \)-state randomized automaton. In this case, while \( b^* \) is independent of \( m \), it is dependent on \( p_1, p_2 \) and can become arbitrarily large for some problems. Thus, the automata are only asymptotically optimal and not close-to-optimal, in the sense developed in Section 3.4.

4.4 Some Variations on the Theme

In Section 4.3, we derived the minimax class of automata for problem (4.31),

\[ H_1: p > p_0 \quad \text{vs.} \quad H_2: p < p_0, \quad 0 < p_0 < 1. \]

Consider the following variation of the problem,

\[ H_1: p > \Pr(Z_1 = 1) \quad \text{vs.} \quad H_2: p < \Pr(Z_1 = 1), \quad (4.39) \]

where \( Z_1, Z_2, \ldots \) is a given random binary string such that the \( Z_i \) are independent, identically distributed random variables with

\[ \Pr(Z_1 = 1) = p_0 \quad \text{and} \quad \Pr(Z_1 = 0) = (1-p_0) = q_0. \]

That is, \( p_0 \) is known only implicitly by means of the random string \( Z_1, Z_2, \ldots \). The design of the optimal automaton for this case seems difficult, since the decision maker must, in some sense, estimate \( p_0 \) and at the same time attempt to resolve the hypotheses. However, we will demonstrate a randomized \( m \)-state machine for this case that precisely matches the performance of the minimax machine derived for problem (4.31), i.e.,
when $p_0$ is known explicitly. Recall from Section 2.3 that Shubert (1974a) studied a similar variation of simple hypothesis testing.

**Theorem 4.3**

The class of automata in Figure 4.3 $\varepsilon$-achieves the lower bounds on $\alpha(p)$ and $\beta(p)$ given in (4.32) and (4.33) for the modified problem (4.39).

**Remark:** The notation in Figure 4.3 is to be interpreted as follows. Event causing transitions to the right is given by: $X = H$ and $Z = 0$, denoted by $(H, 0)$. Similarly, the transitions to the left occur on $(T, 1)$. The other two possible inputs cause self-transitions which are deleted from the diagram for clarity. Also the transitions away from the extreme states 1 and $m$ and the decisions of states 2 to $(m - 1)$ involve randomization. Hypothesis $H_1$ or $H_2$ is decided with equal probability in states 2 to $(m - 1)$.

**Proof.** Follows from a straightforward analysis of the machine. □

Suppose we interpret the $X$ and $Z$ of problem (4.39) as the outcomes of tossing two coins $A$ and $B$, respectively. Then problem (4.39) is equivalent to

$$H_1: p_A > p_B \text{ vs. } H_2: p_A < p_B ,$$

(4.40)

where $p_A$ and $p_B$ are the biases towards heads of the two coins.

Further, since $p_A > p_B (p_A < p_B)$ if and only if $p_A + q_B > 1(p_A + q_B < 1)$, interchanging the labels $H$ and $T$ of the outcome of tossing coin $B$
In states 2 to (m-1) decision H stands for $H_1$ or $H_2$ with equal probability.

Fig. 4.3 A class of automata for problem (4.39).
results in the equivalent problem,

\[ H_1: P_A + P_B > 1 \quad \text{vs.} \quad H_2: P_A + P_B < 1. \]  \hspace{1cm} (4.41)

We define \( \alpha(p_A, p_B) \) as the probability of error achieved by the machine given coins \( A \) and \( B \) and if \( H_1 \) is true; \( \beta(p_A, p_B) \) is similarly defined. The automata in Figure 4.3, \textit{mutatis mutandis}, remain minimax for problems (4.40) and (4.41). (The reader is reminded that the minimax probability of error is 1/2 for these problems.) They also \( \varepsilon \)-achieve the lower bounds on \( \alpha(p_A, p_B) \) and \( \beta(p_A, p_B) \) achievable by any minimax optimal machine, i.e., they achieve error probabilities arbitrarily close to the lower bound applicable to minimax automata, for any value of \( p_A \) and \( p_B \). The lower bounds can be obtained from (4.32) and (4.33) through appropriate substitutions. However, for convenience in later reference, we explicitly set down these lower bounds as follows:

\[
\inf_{\alpha} \alpha(p_A, p_B) = \left[ 1 + \left( \frac{p_A q_B}{q_A p_B} \right)^{m-1} \right]^{-1} \quad \text{for problem (4.40)},
\]

\[
\alpha(p_A, p_B) = \left[ 1 + \left( \frac{p_A p_B}{q_A q_B} \right)^{m-1} \right]^{-1} \quad \text{for problem (4.41)}. \hspace{1cm} (4.42)
\]

\[
\inf_{\beta} \beta(p_A, p_B) = \left[ 1 + \left( \frac{q_A p_B}{p_A q_B} \right)^{m-1} \right]^{-1} \quad \text{for problem (4.40)},
\]

\[
\beta(p_A, p_B) = \left[ 1 + \left( \frac{q_A q_B}{p_A p_B} \right)^{m-1} \right]^{-1} \quad \text{for problem (4.41)}. \hspace{1cm} (4.43)
\]
In the foregoing, we have assumed that the two coins are tossed simultaneously. Cover and Hellman (1970) considered problem (4.40) under the constraint that only one of the coins is to be tossed at each instant. The following bounds can be derived from their work for this case, (again restricting the infimum to minimax optimal machines),

\[
\inf_a \alpha(p_A, p_B) = \left[ 1 + \left( \max \left( \frac{p_A}{p_B}, \frac{q_B}{q_A} \right) \right)^{m-1} \right]^{-1} \text{ for problem (4.40)},
\]

\[
\inf_a \beta(p_A, p_B) = \left[ 1 + \left( \max \left( \frac{q_A}{q_B}, \frac{p_B}{p_A} \right) \right)^{m-1} \right]^{-1} \text{ for problem (4.41)}. \tag{4.44}
\]

The following points are worth making at this juncture. First, while the machines of Figure 4.3 ε-achieve the bounds (4.42) - (4.43), there exists no single class of machines that ε-achieves (4.44) - (4.45). In fact, Cover and Hellman showed that the choice of the ε-optimal class for problem (4.40) depends upon knowing whether \( p_A + p_B > 1 \) or \( p_A + p_B < 1 \), i.e., on having correctly resolved problem (4.41). In view of the equivalence established earlier, it can be seen that, similarly, the ε-optimal class for problem (4.41) depends on having correctly resolved problem (4.40). Thus problems (4.40) and (4.41) are related to each other. It is interesting that the solution of one of these problems requires having solved the other, of equal difficulty. Second, though there is clearly an advantage in memory conferred by allowing simultaneous tossing, as far as infima (4.42) - (4.45) are concerned, this
advantage is less than one bit. (This can be easily checked by replacing \( m \) by \( 2m \) in the bounds given by (4.44) and (4.45), i.e., by allowing one additional bit of memory for nonsimultaneous tossing situations, and comparing it against the bounds (4.42) and (4.43), applicable to \( m \)-state automata.) Also, if only one coin is to be tossed at each instant, then with the addition of two bits of memory, we can use the scheme in Figure 4.3 and do better than the bounds (4.44) - (4.45). (The additional memory is to be used to remember the result of the previous toss and to input the pair of results every other instant to the machine in Figure 4.3).

For purposes of comparison, we present in Figure 4.4a and 4.4b the Cover-Hellman class of machines for problem (4.40). The machine in Figure 4.4a \( \varepsilon \)-achieves the lower bound if \( P_A + P_B < 1 \), and Figure 4.4b is the \( \varepsilon \)-optimal class for \( P_A + P_B > 1 \). Appropriate modifications in the labels of the input stream in Figure 4.4a and Figure 4.4b result in the corresponding \( \varepsilon \)-optimal classes for problem (4.41) for the cases \( P_A > P_B \) and \( P_A < P_B \), respectively. The notation used in these figures is as follows. At each state coin \( A \) or \( B \) is chosen with equal probability. The arrow "\( O(A) = H \)" means: transit if the outcome of tossing \( A \) is \( H \). Other transitions are similarly explained. As usual self-loops are omitted. Hypothesis \( H_1 \) or \( H_2 \) is decided with equal probability in states 2 to \( (m-1) \).

To recapitulate briefly the proceedings so far in this section: we have considered four problem situations, viz., problems (4.40) and (4.41), each with and without simultaneous tossing permitted. For each case, we have derived the lower bounds on error probabilities
In states 2 to \((m-1)\) decision \(H\) stands for \(H_1\) or \(H_2\) with equal probability.

Fig. 4.4 Cover–Hellman automata: (a) for \(p_A + p_B < 1\), and (b) for \(p_A + p_B > 1\).
achievable by minimax automata, and have displayed machines which
\( \varepsilon \)-achieve these bounds. All these machines require artificial random-
ization. In general, the amount of memory, measured in bits, to be
added to deterministic machines to achieve an error probability lower
than that for the corresponding optimal \( m \)-state randomized machine,
while finite, may be dependent on problem parameters and thus may be
unbounded, as in the case discussed at the end of Section 4.3. However,
for all problems considered in this section, the number of extra bits
to be added is bounded, independent of \( m, p_A \) and \( p_B \).

The deterministic machine represented by Figure 4.5 with \( 2m \)
states achieves a lower error probability, for all \( p_A \) and \( p_B \),
than the corresponding \( m \)-state minimax randomized class for problem (4.40) with
simultaneous tossing allowed. That is, the additional memory required is
at most 1 bit. Transitions to the right occur on \((H,T)\), i.e., a \( H \) on
coin \( A \) and a \( T \) on \( B \). Transitions to the left occur on \((T,H)\). The
other inputs cause self-transitions which are deleted in the diagram.

An analysis shows that

\[
\mu_m = \mu_1 \cdot (p_A q_B / p_B q_A)^{m+1},
\]

and

\[
(\mu_1 / p_B q_A) + (\mu_m / p_A q_B) = 1,
\]

where \( \mu_1 \) is the stationary probability for state \( i \).
Fig. 4.5 A deterministic automaton for problem (4.40) with simultaneous tossing.
Hence,
\[
\alpha(p_A, p_B) = \sum_{i=1}^{m} u_i = u_1/p_B q_A
\]
\[
= \left[ 1 + (p_A q_B / q_A p_B)^m \right]^{-1},
\]
which is clearly superior to the bound provided in (4.42). A similar analysis yields the required result for \( \beta(p_A, p_B) \).

Problem (4.41), with simultaneous tossing, can be handled in a similar manner. Moreover, problems (4.40) and (4.41) with nonsimultaneous tossing can be handled by this deterministic solution, with less than two extra bits in the manner indicated earlier in this section. Thus, again, the additional memory requirement is a finite number of bits, independent of \( m, p_A \) and \( p_B \).

4.5 Discussion

The general approach to compound hypothesis testing taken in this chapter can be summarized as follows: view the compound problem as a collection of simple hypothesis testing problems, choose the "worst" problem and design the best machine for that problem. (A simple hypothesis testing problem can be considered the "worst" among a collection, if \( p_s^* \), the greatest lower bound on the probability of error, associated with this problem is the largest.) Clearly, the relevant question is: What are the conditions under which the above approach will yield the minimax solution? It can be seen that it is sufficient that the compound problem satisfies the following condition:
\[
\inf_{a} \sup_{P} P_{S}(a;P) = \sup_{P} \inf_{a} P_{S}(a;P), \quad (4.46)
\]

where \( P \) is an arbitrary simple hypothesis testing problem in the collection, and \( P_{S}(a;P) \) is the probability of error, \( \max(\alpha(a;P), \beta(a;P)) \), of the machine \( a \), on problem \( P \). A direct way to see if (4.46) applies is to design the best machine \( a^{*} \) for the "worst" problem \( P_{w} \) and check that

\[
P_{S}(a^{*}_{w};P) \leq P_{S}(a^{*}_{w};P_{w}) \quad (4.47)
\]

is valid for all \( P \neq P_{w} \). That (4.47) is necessary and sufficient for (4.46) follows from the game-theoretic notion of "saddle-points." All the compound problems considered so far in this paper satisfy the minimax condition (4.46), and Theorem 4.1 is an explicit statement of this for problem (4.1). As further examples, the following problems can be handled by the techniques of this chapter:

\[
H_{1}: p = p_{1} \quad \text{vs.} \quad H_{2}: p \leq p_{2}, \quad 1 > p_{1} > p_{2} > 0, \quad (4.48)
\]

and

\[
H_{1}: p \geq p_{1} \quad \text{vs.} \quad H_{2}: p = p_{2}, \quad 1 > p_{1} > p_{2} > 0. \quad (4.49)
\]

However, the following examples fail to satisfy (4.46),

\[
H_{1}: p = p_{0} \quad \text{vs.} \quad H_{2}: p \neq p_{0}, \quad 1 > p_{0} > 0, \quad (4.50)
\]

and

\[
H_{1}: p \in \{0.7, 0.2\} \quad \text{vs.} \quad H_{2}: p \in \{0.8, 0.1\}. \quad (4.51)
\]

Intuitively, problems (4.50) and (4.51) fail to satisfy the minimax condition (4.46), because the values allowed for the parameter \( p \) under hypothesis \( H_{2} \) surround some or all of the allowed values of \( p \) under \( H_{1} \).
The design of minimax automata for compound problems failing to satisfy (4.46) is an open problem. On the other hand, problems need not be restricted to Bernoulli variables for the condition (4.46) to be satisfied, and minimax machines along the lines suggested in the beginning of this section to be available.
CHAPTER V

THE TWO-ARMED BANDIT PROBLEM

5.1 Definition of the Problem

Given two coins labeled A and B, with respective biases towards heads $p_A$ and $p_B$, it is desired to conduct an infinite sequence of tosses, selecting at each trial one of the coins, in such a way as to maximize the long-run proportion of heads. The values of $p_A$ and $p_B$ are not known. The problem, then, is to determine the optimal strategy for the selection of the coin to be tossed at a given instant. This problem, originally posed by Robbins (1956), is generally known as the two-armed bandit problem (TABP).

The choice of coins in the problem above is similar to that of a gambler who selects at each instant one of the two levers of a slot machine. The machine rewards each pull with $1, with probabilities $p_A$ and $p_B$ respectively, for the two arms. The objective of the gambler, obviously, is to maximize his winnings, but he does not know the values of $p_A$ and $p_B$. Hence the name two-armed bandit for the problem.

This chapter is concerned with the TABP under a finite memory constraint. In other words, our objective is to design a finite-state machine that learns to select the coin with the larger bias more often.
Clearly, the finite-state automaton has to perform the compound hypothesis test

\[ H_1: P_A > P_B \text{ vs. } H_2: P_A < P_B , \]  

studied in Chapter IV. A solution for TABP, however, has an additional burden of maximizing the proportion of heads at the same time as it attempts to resolve the compound hypothesis test. The decision maker (the finite-state automaton) has two aims in selecting the coin for each toss: to maximize the proportion of heads (by selecting the coin known to have a larger bias, on the basis of available information) and to increase the knowledge of the coin biases (by selecting the coin about which least is known). The aims could conflict. Thus, the problem models the conflict between data gathering and control that arises in many fields of application (Chernoff, 1975).

Denote by \( p(a; p_A, p_B) \), the asymptotic proportion of heads obtained, given the two coins \( A \) and \( B \) and the automaton \( a \). Our objective is to maximize \( p(a; p_A, p_B) \). Also, it is clear that for any given automaton \( a \), \( \min(p_A, p_B) \leq p(a; p_A, p_B) \leq \max(p_A, p_B) \). Further, for any given \( p_A \) and \( p_B \), there exists an automaton \( a \) such that

\[ p(a; p_A, p_B) = \max(p_A, p_B). \]

For example, if \( p_A > p_B \), the automaton \( a_A \) that always chooses coin \( A \) achieves \( p(a_A; p_A, p_B) = p_A = \max(p_A, p_B) \).

But this automaton \( a_A \) will achieve \( p(a_A; p_A, p_B) = p_A = \min(p_A, p_B) \), if instead \( p_A < p_B \). Thus, the automaton \( a_A \) does not maximize \( p(a; p_A, p_B) \) for all values of \( p_A \) and \( p_B \). Consider another automaton, say \( a_R \), that chooses one of the two coins randomly at any instant. For this random automaton, \( p(a_R; p_A, p_B) = (p_A + p_B)/2 \), for all values
of $p_A$ and $p_B$. Clearly, this automaton $a_R$ does not maximize $p(a; p_A, p_B)$ for any value of $p_A$ and $p_B$, except for the degenerate case of $p_A = p_B$. In fact, there exists no automaton that maximizes $p(a; p_A, p_B)$ for all values of $p_A$ and $p_B$. Thus other approaches are needed. One possibility is a Bayesian formulation of the problem in which the average value of the quantity $p$ is maximized. However, as in the compound hypothesis testing problem considered in the last chapter, simply assigning prior probabilities to the hypotheses in (5.1) is not adequate for this; one needs prior distributions on $p_A$ and $p_B$. Again, for reasons set forth in the last chapter, we shall assume no such prior distributions, and hence will not attempt the Bayesian formulation. An alternative approach is the maximin principle on $p$, i.e., choose an automaton $a$ that maximizes $\{\inf p(a; p_A, p_B)\}$, where the infimum is taken over $\{(p_A, p_B)\}$. However, this approach can lead to some unsatisfactory solutions. Consider, for example, a particular case of the two-armed bandit problem, in which $p_B$, the bias of coin $B$, is precisely known. We name this particular case TABPO. For any automaton $a$

$$p(a; p_A, p_B) \geq p_B \quad \text{if } p_A > p_B,$$

$$\leq p_B \quad \text{if } p_A < p_B.$$  

Since the bias $p_B$ is known for this problem, an automaton that maximizes $\{\inf p(a; p_A, p_B)\}$ simply chooses coin $B$ at every instant and thus achieves $p(a; p_A, p_B) = p_B$, for all values of $p_A$ and $p_B$. While this performance is the best possible if $p_A < p_B$, it is inferior
to that of even the random automaton $d_R$ for $p_A > p_B$. This we consider unsatisfactory, and hence the maxmin approach is unsuitable for this problem. We take a different approach in which attention is restricted to a subclass called expedient machines (Tsetlin, 1973). These machines have the property that $p(a; p_A, p_B) > (p_A + p_B)/2$ as long as $p_A \neq p_B$, i.e., they perform better than the random automaton for all values of $p_A$ and $p_B$. Several expedient automata have been previously exhibited (Fu and Li, 1969; Tsetlin, 1973) and our objective will be to determine the optimal automaton among the expedient ones.

Denote by $r(a; p_A, p_B)$, the asymptotic proportion of the choice of the coin with larger bias, given the two coins $A$ and $B$ and the automaton $a$. Since the long-run proportion of heads is maximized if and only if $r(a; p_A, p_B)$ is maximized, it is just as good a criterion for comparing automata. The quantity $r$ is such that, for any automaton $a$, $0 < r(a; p_A, p_B) < 1$, and it is convenient to carry on the discussion in terms of $r$ rather than $p$. Again, in terms of $r$, the expedient automata have the property that $r > 1/2$ as long as $p_A \neq p_B$. We restrict attention to expedient automata and attempt to maximize $r(a; p_A, p_B)$ and thus equivalently $p(a; p_A, p_B)$.

To recapitulate: Since there are no uniformly best machines for all values of $p_A$ and $p_B$, and since machines which perform very well in one range of values of $p_A$ and $p_B$ tend to perform inferior to even random machines in another range of values, we first delineate a class of machines which have a prima facie reasonable quality for all values of $p_A$ and $p_B$, viz., machines in this class perform better than
random machines for the entire range of values of $p_A$ and $p_B$. We then seek to find the best machine among this class, called the expedient class.

There is a slightly different way of looking at the situation, which will unify the approach with that of Chapter IV. If we adopted the maxmin approach on $r$, since

$$\sup_a \inf_{(p_A,p_B)} r(a; p_A, p_B) = 1/2,$$

(5.2)
even the trivial automaton $a_R$, which chooses one of the two coins randomly at any instant, is maxmin, i.e., the maxmin principle is too inclusive. The expedient automata above are also maxmin, by definition. Thus, our approach is equivalent to adopting the maxmin principle on the quantity $r$, and demanding additional optimality properties for the automata, besides the maxmin optimality, in order to disqualify trivial solutions. This viewpoint parallels our approach to compound hypothesis testing. Also, observe that the maxmin optimal solution derived for TABPO, a particular case of TABP, always chooses coin B and hence fails to be maxmin optimal, if the criterion is $r$ instead of $p$.

Thus, even though the asymptotic proportion of heads, $p(a; p_A, p_B)$, is maximized if and only if $r(a; p_A, p_B)$ is maximized, maximizing $\{\inf p(a; p_A, p_B)\}$ is not necessarily equivalent to maximizing $\{\inf r(a; p_A, p_B)\}$, where the infimum is taken over $\{(p_A, p_B)\}$.

Our objective is to design a finite-state automaton $a$, with inputs $X_n$, outputs $E_n$, and state space $S = \{1, 2, \ldots, m\}$ such that $T_n$, the state of the automaton at time $n$, is varied according to
the rule,

\[ T_n = f(T_{n-1}, X_n), \quad T_n \in S, \quad (5.3) \]

where \( X_n \), the input at time \( n \), consists of a pair \( (e_n, Y_n) \), the coin tossed at time \( n \) and the outcome of the coin toss—head (H) or tail (T). The choice of the coin is made according to the rule

\[ e_n = e(T_{n-1}), \quad e_n \in \{A, B\}. \quad (5.4) \]

Both the transition function \( f \) and the output functions \( e \) are allowed to be stochastic, but must be data- and time-independent. That is, the transition and output functions do not change with time or observed data. This is in agreement with the finite-memory model in the previous chapters. Let \( \alpha_i \) denote the probability of choosing coin \( A \) in state \( i \) at any instant, i.e.,

\[ \alpha_i = \Pr(e_n = A | T_{n-1} = i). \quad (5.5) \]

Thus, coin \( B \) is chosen in state \( i \) with probability \( 1 - \alpha_i \). A state \( T_0 \in S \) is designated as the start state. Let \( s_n = 1 \) or 0 according as the coin chosen at time \( n \) has a larger bias or not. Given the two coins \( A \) and \( B \), with respective biases towards heads \( p_A \) and \( p_B \), and the automaton \( a \), define

\[ r(a; p_A, p_B) = E(\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} s_i). \quad (5.6) \]

As in previous chapters, the expectation operation in (5.6) needs to be performed only if the automaton \( a \) has absorbing states, and the
specifications of \( T_0 \) is immaterial if attention is restricted to irreducible automata.

We now proceed to the definition of expedient automata.

**Definition**

An automaton \( a \) is called expedient if, for all \( p_A \) and \( p_B \),
\[
r(a; p_A, p_B) \geq 1/2,
\]
with the equality holding true only if \( p_A = p_B \).

An expedient automaton, therefore, does better than an automaton that chooses one of the two coins randomly at any instant. Henceforth, we will restrict attention to expedient automata to determine an upper bound on \( r(a; p_A, p_B) \), for all \( p_A \) and \( p_B \). We will demonstrate that these bounds are least upper bounds by displaying sequences of expedient automata which achieve these bounds to arbitrary closeness. We will also present several close-to-optimal expedient machines.

5.2 A Particular Case of TABP—TABPO

In contrast to the TABP where both \( p_A \) and \( p_B \) are unknown, we assume, in this section, that \( p_B \), the bias of coin B, is precisely known. We give the name TABPO to this version of TABP. Chernoff (1975) uses this formulation in connection with the problems of sequential testing in clinical trials and sequential sampling inspection. The solution to TABPO is not obtainable as a special case of the solution for TABP, and is thus worth consideration in its own right.
Let \( p_1 \) and \( p_2 \) be such that \( 1 > p_1 > p_B > p_2 > 0 \). Denote, for notational simplicity

\[
x_1 = r(a; p_A = p_1, p_B),
\]

and

\[
x_2 = r(a; p_A = p_2, p_B).
\]

That is, the quantities \( r_1 \) are the asymptotic proportion of the choice of the correct coin when \( p_A \), the bias towards heads of coin \( A \), equals \( p_1 \). Observe that when \( p_A = p_1 (p_A = p_2) \), coin \( A(B) \) has larger bias and hence \( r_1(r_2) \) measures the asymptotic proportion of the choice of the coin \( A(B) \). Consider the TABPO situation where the bias \( p_A \) can take only one of the two values \( p_1 \) and \( p_2 \), i.e., the hypothesis test to be resolved is:

\[
H_1: p_A = p_1 \quad \text{vs.} \quad H_2: p_A = p_2.
\]

The theory developed in Section 2.3.1 allows us to write down the following operating characteristic:

\[
x_1 x_2 \leq \gamma^{(m-1)} (1-x_1)(1-x_2),
\]

where

\[
\gamma = \frac{p_1 q_2}{p_2 q_1},
\]

\[
q_1 = 1-p_1, \quad \text{and} \quad q_2 = 1-p_2.
\]

The following theorem establishes, for all \( p_A \) and \( p_B \), an upper bound on \( r(a; p_A, p_B) \).
Theorem 5.1. Restricting attention to expedient automata,

\[ r(a; p_A, p_B) \leq \frac{(p_A q_B / p_B q_A)^{m-1}}{1 + (p_A q_B / p_B q_A)^{m-1}} \text{ for } p_A > p_B, \]

\[ \leq \frac{(p_B q_A / p_A q_B)^{m-1}}{1 + (p_B q_A / p_A q_B)^{m-1}} \text{ for } p_A < p_B. \] (5.12)

Proof. Consider two possible bias values \( p_1, p_2 \) for coin \( A \). Observe that if \( p_1 \) and \( p_2 \) are both greater than \( p_B \) or both less than \( p_B \), trivial instances of TABPO result, and they can be ignored. Let

\( 1 > p_1 > p_B > p_2 > 0, \) and use the notation of (5.7) and (5.8). Since we restrict attention to expedient automata, \( r_2 > 1/2, \) and \( r_2/(1-r_2) > 1. \) This implies, from the operating characteristic (5.10),

\[ r_1/(1-r_1) \leq \gamma^{(m-1)}, \] (5.13)

i.e.,

\[ r_1 \leq \gamma^{(m-1)}/(1+\gamma^{(m-1)}). \] (5.14)

Since (5.14) is true for any value of \( p_2 < p_B \), due to expediency required at all such \( p_2 \), the tightest such bound results as \( p_2 + p_B \), i.e.,

\[ r_1 = r(a; p_A = p_1, p_B) \leq \frac{(p_1 q_B / p_B q_1)^{m-1}}{1 + (p_1 q_B / p_B q_1)^{m-1}} \text{ for } p_1 > p_B. \] (5.15)

A similar analysis yields an upper bound for \( r_2 \). Hence the theorem. \( \square \)
There exists no automaton that achieves the bounds for all values of $p_A$ and $p_B$. (See, for example, a parallel result in compound hypothesis testing, Section 4.3.) However, the fact that these bounds are indeed the least upper bounds follows from consideration of the automaton in Figure 5.1. (To be precise, Figure 5.1 represents a class of automata, the class being generated by varying $\delta$. But, once $\delta$ is fixed, Figure 5.1 represents only one automaton.) The transitions to the right and left occur on the input pairs $(A,H)$ and $(A,T)$, respectively. $a_i$ denotes the probability of choosing coin $A$ for tossing in state $i$. Note, further, that the transition away from the state $m$ occurs with probability $k\delta$ on input $(A,T)$, where $k = (p_B/q_B)^{m-1}$. For $0 < \delta < 1/2$, the machine is expedient and as $\delta \to 0$ achieves the bound given by (5.12), for any $p_A$ and $p_B$.

Observe that the transitions in this automaton all occur as a result of tossing coin $A$. This is intuitively pleasing, because it is coin $A$ whose bias we attempt to learn. Further, as $\delta \to 0$, the stationary occupancy probabilities are concentrated in the extreme states, 1 and $m$. Thus, the coin tossing probabilities, $a_i$, in states 2 to $(m-1)$ can be simply set to 1, without changing the performance obtained as $\delta \to 0$. However, the particular decision rule in Figure 5.1 facilitates the proof of the expediency of the automaton. The proof is very similar to that presented in Appendix B.
Fig. 5.1 Optimal automaton for the TABPO. Self transitions are deleted.
5.3 The Two-Armed Bandit Problem

Cover and Hellman (1970) considered a particular version of TABP, in which one of the coins has bias $p_1$ and the other has bias $p_2$ towards heads ($p_1$ and $p_2$ are known values), but it is not known which coin has which bias. That is, the hypotheses to be resolved are:

$$H_1: p_A = p_1 \text{ and } p_B = p_2 \text{ vs. } H_2: p_A = p_2 \text{ and } p_B = p_1,$$ (5.16)

as opposed to the hypotheses in the general Robbins version,

$$H_1: p_A > p_B \text{ vs. } H_2: p_A < p_B.$$

It turns out that the construction of the optimal randomized $m$-state machine requires only the knowledge of whether $p_1 + p_2$ is greater or less than 1, if the prior probabilities of the two hypotheses in (5.16) are equal. Also, if one adopts a minimax principle over the two hypotheses in (5.16) instead of a Bayesian approach with equal prior probabilities, the same solution results. The following theorem can be inferred from their results.

**Theorem 5.2** Restricting attention to expedient automata,

$$r(a; p_A, p_B) \leq \frac{(\max(p_A/p_B, q_B/q_A))^{m-1}}{1+\max(p_A/p_B, q_B/q_A)^{m-1}} \text{ for } p_A > p_B,$$

$$\leq \frac{(\max(p_B/p_A, q_A/q_B))^{m-1}}{1+\max(p_B/p_A, q_A/q_B)^{m-1}} \text{ for } p_A < p_B.$$

(5.17)
The choice of the optimal automaton, in this case, depends on the knowledge of $p_A + p_B$. Figures 5.2a and 5.2b depict the optimal automata for $p_A + p_B < 1$ and $p_A + p_B > 1$, respectively. For $p_A + p_B = 1$, either one is optimal. Again, in the limit as $\delta \to 0$, the bounds given by (5.17) are achieved. As in Section 5.2, these machines are expedient for $\delta < 1/2$.

It is clear that use of one of the automata exclusively can lead to inferior performance in the other range of $p_A + p_B$. Further, it has been shown in Section 4.4 that the compound hypothesis testing of the form,

$$H_1: p_A + p_B > 1 \quad \text{vs.} \quad H_2: p_A + p_B < 1,$$

and (5.1) are such that the construction of the optimal automaton for one of these problems requires having solved the other, both being of equal complexity. Hence it would be useful to have an automaton that is close to optimal for all values of $p_A + p_B$ for use in the absence of any knowledge about $p_A + p_B$. Consider the automaton in Figure 5.3, obtained by suitably modifying Milyutin's solution for the problem on the behavior of automata in random media (Milyutin, 1965). A simple analysis yields, in the limit as $\delta \to 0$,

$$r(a; p_A, p_B) = \frac{(q_B/q_A)(p_A q_B/p_B q_A)^{m-1}}{1 + (q_B/q_A)(p_A q_B/p_B q_A)^{m-1}} \quad \text{for} \quad p_A > p_B,$$

$$= \frac{(q_A/q_B)(p_B q_A/p_A q_B)^{m-1}}{1 + (q_A/q_B)(p_B q_A/p_A q_B)^{m-1}} \quad \text{for} \quad p_A < p_B.$$
Fig. 5.2 Cover-Hellman automata: (a) for $p_A + p_B < 1$, and (b) for $p_A + p_B > 1$. 
Fig. 5.3 A close-to-optimal automaton for the TABP. Self transitions are deleted.
Hence, the automaton in Figure 5.3 requires at most 2m states to yield a performance better than the bounds of (5.17). That is, for this sub-optimal scheme that requires no knowledge of \( p_A + p_B \), we require at most 1 extra bit of memory, independent of \( m \), to match the performance of the optimal automaton, for all \( p_A \) and \( p_B \). In this sense the automaton in Figure 5.3 is close to optimal for the TABP. Also note that this 2m-state automaton shows a better performance than the bounds (5.12) for the TABPO. In other words, the memory saved by providing an exact knowledge of the bias of one of the coins is less than 1 bit.

We observe, further, that in all these cases the automata presented require artificial randomization, either in the transition or in the coin selection. We now show that if two extra bits of memory are available, we can construct a deterministic automaton with no loss of performance and thereby eliminate the need for artificial randomization. The automaton described below requires \((4m-3)\) states and has two absorbing states, 1 and \((4m-3)\), where coins B and A, respectively, are tossed exclusively. The coin selection is made deterministically as follows:

\[
e(i) = \begin{cases} 
B, & i = 1, 2, \ldots, m, \\
= A, & i = m+1, \ldots, 3m-3, \\
= B, & i = 3m-2, \ldots, 4m-4, \\
= A, & i = 4m-3. 
\end{cases}
\quad (5.20)
\]

Since the states 1 and 4m-3 are absorbing, the transition rule is described below only for the other states.

\[
\begin{align*}
f(i,(B,H)) &= i-1, \ i = 2,3,\ldots,m, \\
f(i,(B,T)) &= 2m-1, \ i = 2,3,\ldots,m; \\
f(i,(A,T)) &= i-1, \ i = m+1,\ldots,2m-1, \\
f(i,(A,H)) &= 2m-1, \ i = m+1,\ldots,2m-2, \\
f(i,(A,T)) &= 2m-1, \ i = 2m,\ldots,3m-3, \\
f(i,(A,H)) &= i+1, \ i = 2m-1,\ldots,3m-3; \\
f(i,(B,T)) &= i+1, \ i = 3m-2,\ldots,4m-4, \\
f(i,(B,H)) &= 2m-1, \ i = 3m-2,\ldots,4m-4. \quad (5.21)
\end{align*}
\]

The state (2m-1) is designated the start state. Figure 5.4 depicts a typical portion of the automaton. A straightforward analysis shows that for all \( p_A \) and \( p_B \) such that \( p_A \neq p_B \), \( r \) for this machine is identical to the quantities on the right side of (5.12) and thus is strictly greater than the least upper bound given in (5.17) for m-state machines.

5.4 Discussion

This chapter has studied two bandit problems, viz., the TABPO and the TABP, and optimal randomized m-state automata were displayed. It was shown that the memory saved by providing an exact knowledge of one of the coin biases, over the TABP situation, is at most 1 bit, independent of \( m \). Further, a close-to-optimal deterministic automaton that eliminates the need for artificial randomization was also provided. This automaton is close to optimal in the sense that it requires at
Fig. 5.4 A close-to-optimal deterministic automaton for the TABP.
most 2 extra bits of memory, independent of \( m \), to match the performance of the optimal randomized \( m \)-state automaton for all \( p_A \) and \( p_B \).

Both the problems studied here, however, involve only 2 coins. Extensions of the results of this paper to situations where more than 2 coins are involved still remains open. Some ad hoc expedient automata are available in the literature (Fu and Li, 1969; Tsetlin, 1973). Before an optimal solution to the many-armed bandit problem is possible, the problem of multiple hypothesis testing with finite memory needs to be solved.

Further, finite time, finite memory solutions to these problems are of interest. Vasilev (1967) and Witten (1973) studied the finite time behavior of some solutions to the TABP. No optimal solution, however, is available. Some recent progress has been reported by Cover et al. (1976).
CHAPTER VI

CONCLUDING REMARKS

6.1 Summary

The concerns of this dissertation lie in the interface between theoretical computer science and statistical decision theory. The capabilities of finite-state automata have earlier been investigated with respect to classes of functions computed and languages accepted or generated. In the preceding chapters of this dissertation we have explored the power of finite-state machines as decision makers in a variety of basic statistical decision problems. The results can also be viewed as a study of memory complexity in algorithms implementing statistical decision rules. The particular problems considered are: multiple simple hypothesis testing, compound hypothesis testing, and the two-armed bandit problem.

The finite memory restriction is added to these decision theory problems by constraining the statistic on which the decisions are based to assume only a finite number of possible values. Then, the statistic can be stored in a finite number of bits of computer memory. Equivalently, the decision maker may be viewed as a time-invariant, finite-state stochastic automaton. The transition and decision rules themselves do not change with time or observed data, but they can be randomized. This model of finite memory was introduced by Hellman and
Cover (1970). Our objectives in this dissertation have been to identify several problem situations and develop corresponding finite memory theory for them. A summary of specific technical accomplishments follows.

The multiple hypothesis testing problem under the time-invariant finite memory constraint has resisted a solution since 1970 when Hellman and Cover solved the 2-hypothesis case. Even though particular cases involving "unbounded likelihood ratios" had been solved, the problem of determining $P^*$, the greatest lower bound achievable by m-state automata, and a class of automata that achieves an error probability arbitrarily close to $P^*$ has remained open for the general K-hypothesis testing. We derived a set of inequality constraints relating problem statistics and machine performance parameters for m-state automata. Further, we introduced a notion called the "useful spread" and demonstrated the necessity of considering it in deriving a tight lower bound. In general, the relationships that exist among the various useful spreads are not known, and the set of inequalities mentioned earlier do not incorporate all possible constraints on realizable automata. Consequently, the bound on $P^*$ that we derived is necessarily loose. Nevertheless, the bound is not a trivial one, and it leads to the construction of automata that are close to optimal. These automata, being sub-optimal, require some additional memory to match the performance of the optimal m-state automaton. What is noteworthy is that the number of bits of required additional memory is independent of m and the problem parameters, and is only a function of the number of hypotheses.
Problems for which close to optimal solutions were constructed include a class of Bernoulli hypothesis testing and another particular class called "symmetric hypothesis testing."

Compound hypothesis testing with finite memory was considered in detail. No arbitrary prior distribution was assumed over the problem parameters, and thus a Bayesian approach was not followed. Instead, the minimax principle was adopted throughout. The greatest lower bound on the probability of error achievable by m-state automata was derived using the minimax principle for problems on Bernoulli observation space. Minimax automata may or may not exist. Instances of compound hypothesis testing problems corresponding to these cases were presented. The automata we displayed also possess some additional optimality properties which become important for problems with minimax error probability $1/2$. Even the automaton that chooses one of the two hypotheses randomly at any instant is minimax optimal for these problems, and hence the need for additional optimality properties in order to disqualify such trivial solutions. Certain hypothesis testing problems involving the sum and the difference of the biases of two coins were also considered. The greatest lower bound achievable by m-state minimax automata was determined and a class of automata achieving these bounds to arbitrary closeness was exhibited. In all these cases the optimal automata involved randomization. If randomization requires additional memory, these schemes are far from optimal. Hence, it is useful to consider purely deterministic automata. We constructed close-to-optimal deterministic automata which, in most cases above, required at most
three extra bits of memory to match the performance of the optimal randomized m-state automaton. Again, the number of extra bits of memory needed was independent of m and the problem parameters.

The results on compound hypothesis testing were then applied to the two-armed bandit problem (TABP). Given two coins with unknown biases, the objective was to conduct an infinite sequence of tosses so as to maximize the long-run proportion of heads. This problem models the conflict between learning and control that arises in many fields of application. Again, the Bayesian approach was not followed, since it would have required the assumption of an arbitrary prior distribution on the problem parameters. The specification of an appropriate criterion for this problem was found to involve some very subtle notions. There are no uniformly best automata for all values of the problem parameters, i.e., coin biases. Automata which are optimal in one range of values of the parameters perform, in the other range, worse than even random schemes, i.e., those which choose one of the two coins randomly at any instant. We delineated a class called the "expedient" class, all machines in which are superior to random schemes. We then sought, from among the expedient class, a machine which maximized the long-run proportion of heads. The least upper bound on the asymptotic proportion of the choice of the coin with larger bias was determined and automata that achieve these bounds to arbitrary closeness were exhibited. It was further shown that memory saved by providing the complete information on one of the coin biases is at most one bit. A close-to-optimal deterministic automaton that requires at
most two extra bits of memory to eliminate the need for randomization was also presented.

The following two notions which have been contributed by this dissertation deserve emphasis.

1. Close-to-optimality: The primary objective of finite memory theory is to devise decision schemes that consume a specified amount of memory in implementation and yield results that are optimal for this memory. That is, according to the finite memory model and the criterion we have employed, the solution must achieve the smallest probability of error for any given memory size, the memory size being measured in the number of states of the finite-state automaton or the corresponding number of bits. Equivalently, the finite-state automaton should require as few states (or bits) of memory as possible to achieve a specified probability of error. However, the optimization involved is, in general, intractable for most problems. It would be useful to have, for such problems, finite memory sub-optimal algorithms which are known to be only a finite number of bits away from optimality, independent of specific problem parameters. The principle of close-to-optimality developed in this dissertation requires that sub-optimal automata be applicable over a class of problems. Further, for any specific problem in this class, the additional number of bits of memory required by the sup-optimal automaton to match the performance of the corresponding optimal m-state randomized automaton should be bounded not only independent of m, but also the specific problem parameters. Thus, the notion of close-to-optimality is robust and seems to have the potential of
making finite memory theory applicable to a larger class of problems which are currently intractable.

2. Non-Bayesian formulation of optimality: Previous workers in finite memory theory, except for Shubert (1974b), have all developed optimal solutions only in a Bayesian formulation. That is, they assumed prior probabilities and attempted to minimize the expected asymptotic proportion of errors. Such an attempt for the compound hypothesis testing problems would require us to assume arbitrary prior distributions on the problem parameters, which we consider unnatural, and hence have avoided here. We have employed the familiar minimax principle. The minimax principle, simply stated, attempts to minimize the worst case probability of error. However, there exist compound hypothesis testing problems for which the minimax principle is too inclusive, allowing even trivial automata to qualify as minimax optimal. Hence, it has been necessary to demonstrate additional optimality properties for the decision schemes we displayed, besides the minimax optimality. Similar problems arise in solving the two-armed bandit problem also. In our approach we attempt to choose the best machine among the class of expedient machines which have a prima facie "reasonable" behavior, i.e., they are always better than random schemes (except when the coin biases are equal). This approach is analogous to restricting to unbiased estimators in determining the uniformly best estimate (Ferguson, 1967). Thus, we have shown that non-Bayesian formulation of optimality is both possible and useful in many problems.
We have also thrown some light on the relative powers of randomized and deterministic automata in decision making. For most problems with discrete observation space, randomized machines are superior to deterministic ones, if randomization is implemented at no cost in memory. Hellman and Cover (1971) showed that given any finite $b$, there exist simple hypothesis testing problems for which the optimal randomized schemes save at least $b$ bits in memory over the optimal deterministic ones. This has left the general impression that deterministic schemes are considerably inferior to randomized schemes. We have presented several cases where the performance of deterministic automata is not very far from that of randomized ones. In situations where the implementation of randomization would in fact consume memory, such relatively efficient deterministic automata could be used to advantage. Again, the robustness and the intuitively reasonable structure of deterministic machines could make finite memory theory have a practical impact.

Another aspect of our investigation concerns the amount of memory saved by providing partial information in a decision making problem. For instance, in the TABPO problem, a particular case of TABP, the bias of one of the coins is known exactly. It turns out that this information saves at most one bit of memory. Similarly, in the TABP the optimal automaton can be chosen only if it is known whether the sum of the coin biases is greater than or less than unity. Again we show that this knowledge of the bias-sum is worth only one bit, i.e., automata can be constructed which, with this additional bit of memory, but without the knowledge of the sum of the coin biases, can do better than the optimal automata for this problem.
6.2 Suggestion for Further Research

Several open problems exist for further research in this area, many of which have already been indicated in the various chapters. We believe that the two notions, close-to-optimality and non-Bayesian formulation of optimality, contributed by this dissertation will provide a basis for dealing with currently intractable problems in finite memory decision theory. Consider, for example, a multiple compound hypothesis testing problem whose hypotheses are of the form

\[ H_1: p > p_1, \quad H_2: p_1 > p > p_2, \quad H_3: p < p_2, \]

with \( 1 > p_1 > p_2 > 0 \), where \( p \) is the parameter of a Bernoulli random variable. Clearly, the complexity of deriving optimal finite memory solutions even for the multiple simple hypothesis case suggests that one might look for only close-to-optimal solutions for this problem.

Again, the results of Chapter IV, where we studied compound hypothesis testing involving only two hypotheses, suggest that any Bayesian formulation assuming arbitrary prior distributions is unnatural and will lead to fairly intractable optimization problems. Hence, some sort of non-Bayesian formulation needs to be adopted. Similar arguments hold for the estimation problem and the many-armed bandit problem (where more than two coins are given), both of which involve multiple compound hypothesis testing as a component.

The finite memory problems studied in this dissertation have all assumed the availability of an arbitrarily large number of observations. Allowing the sample size to go to infinity, we have been able
to employ the results on the stationary probability distributions of Markov chains to prove the results of interest to us. But finite-sample problems remain intractable and no optimal solution has been found for any of the problems considered in this area. It is also of interest to study the rate of convergence of the probability of error with sample size for these finite memory solutions.

We believe that the following problems deserve special attention.

1. Our approach to determining \( P^* \), the greatest lower bound on the probability of error, for the multiple hypothesis testing problem has been based on determining the constraints on machine parameters applicable to m-state automata. But in a more fundamental sense, restriction to finite memory for decision making implies a restriction on the information that can be gathered and stored from the previous samples. Application of information theoretic notions to decision theory is well known (Kullback, 1968; Blahut, 1974). Whether and how the information theoretic notions can be applied to finite memory decision making, to provide an alternative way of deriving \( P^* \), is an open problem.

2. We observed that in most cases optimal finite memory solutions involve randomization. Viewed another way, randomized finite memory solutions require fewer numbers of states or bits of memory to achieve a given probability of error. However, the role played by randomization is not very clear. For example, whether a pseudo-random number generator, which is characterized by a finite period, can provide the random input for our infinite sample problems without degrading the performance.
appreciably is an open question. Further, Rabin (1976) and Pearl (1976) have recently observed that by providing randomization in their algorithms for certain non-statistical problems a saving of computational time can be realized. The relationship of our results to this deserves further study.

3. In this dissertation, we modeled the finite memory restriction by restraining the decision maker to be a time-invariant finite-state automaton. That is, the transition and decision schemes themselves do not change with time. Hence, there is no need for a clock that keeps track of the time instant. In contrast, the time-varying schemes reviewed in Section 2.2 require a clock of unbounded memory size for implementation. There appears to be a middle ground. Periodically time-varying schemes require only a clock of finite size. For example, an m-state automaton whose transition and decision rules depend on whether the time instant is odd or even requires a two-state clock for its implementation, besides the m-state memory for data storage. Recently, Mullis (1973) has shown that periodically time-varying deterministic schemes can show better performances than optimal time-invariant randomized schemes. He considered a Bernoulli symmetric hypothesis testing problem—see Section 2.3—of the form,

\[ H_1: p = p_1 \quad \text{vs.} \quad H_2: p = p_2, \]

where \( 1 > p_1 > p_2 > 0 \) and \( p_1 = 1 - p_2 \). It is easily seen that, for two-state automata, \( P^* \), the greatest lower bound on the probability of error, equals \( p_2 \). The structure of the optimal randomized
two-state automata can be derived from Figure 2.1. A transition from state 1 to state 2 occurs on input $X = H$ and a transition from state 2 to state 1 occurs on $X = T$. The randomization coefficient $k^*$ equals unity, and in fact no randomization is needed for optimal schemes. $P^*$ is achievable for this problem. Mullis demonstrated a two-state deterministic automaton whose transition rule changes according to whether the time instant is odd or even. His automaton has the same transition structure as the optimal time-invariant one mentioned above, except that only self transitions can occur at odd time instants on input $X = T$ and at even time instants on input $X = H$. Because of the periodic nature of the transition scheme, the stationary occupancy probabilities are also periodic. A simple analysis can show that the probability of error is less than $p_2$. Thus, this periodic time-varying two-state scheme, which also requires a two-state clock, can perform better than a two-state optimal time-invariant randomized scheme. It must be pointed out that this periodically time-varying scheme has used four states of memory, if the memory needed for implementing the clock is also taken into consideration. However, this periodically time-varying scheme is not necessarily optimal. More generally, whether an $m$-state periodically time-varying automaton performing in association with a $k$-state clock can show a better performance than a $km$-state time-invariant randomized automaton is an open question. If it does, then it is beneficial to divide the memory into two parts—one to implement the clock and the other to store the statistic. This area deserves further study.
The statistical inference problems studied in this dissertation arise in many practical computer science and engineering applications. It is well known that the problem of electrical signal detection under uncertainty introduced by noise can be posed as a hypothesis testing problem. Statistical pattern classification, again, is basically hypothesis testing. Reliability engineering and quality control are examples of situations where compound hypothesis testing and estimation problems arise. If the inference algorithms are to be implemented on a small computer and if an arbitrarily large number of observations are available, the computer memory can become a major limitation. Hence, it is necessary to develop finite memory solutions for these fundamental problems in statistical inference. We believe that this dissertation has contributed answers to many questions in that direction.
APPENDIX A

PL* FOR 3-HYPOTHESIS TESTING

Consider the Lagrange minimization of

\[ P(e) = \frac{1}{3} \sum_{a \neq b} P_{ab}, \quad a, b \in \{1, 2, 3\}, \quad (A.1) \]

subject to

\[ P_{12}P_{21} \geq \tau_{12}^{-1} (1-P_{12}P_{13})(1-P_{21}P_{23}), \quad (A.2) \]
\[ P_{23}P_{32} \geq \tau_{23}^{-1} (1-P_{21}P_{23})(1-P_{31}P_{32}), \quad (A.3) \]
\[ P_{13}P_{31} \geq \tau_{13}^{-1} (1-P_{12}P_{13})(1-P_{31}P_{32}). \quad (A.4) \]

Further, we require that

\[ P_{11} = 1-P_{12}P_{13}, \quad (A.5) \]
\[ P_{22} = 1-P_{21}P_{23}, \quad (A.6) \]
\[ P_{33} = 1-P_{31}P_{32}, \quad (A.7) \]

and \( 1 \geq P_{ab} \geq 0, \quad a, b \in \{1, 2, 3\} \).

Let

\[ P_{12}/P_{11} = 1/b, \quad P_{13}/P_{11} = a/b; \quad (A.9) \]
\[ P_{21}/P_{22} = 1/d, \quad P_{23}/P_{22} = c/d; \quad (A.10) \]
\[ P_{31}/P_{33} = 1/f, \quad P_{32}/P_{33} = e/f; \quad (A.11) \]

and \( \kappa_{ab} = \tau_{ab}^{-1}, \quad a, b \in \{1, 2, 3\}, \quad a < b. \quad (A.12) \)

A change of variables in the Lagrange minimization, using (A.9)—(A.12), allows considerable ease in algebraic manipulations. (A.2)—(A.4) now reduce to
The expression for $P(e)$, then, becomes

$$P(e) = \frac{1}{3} \left[ \frac{1+a}{1+a+b} + \frac{1+c}{1+c+d} + \frac{1+e}{1+e+f} \right].$$

(A.16)

The Lagrangian function,

$$F = P(e) + L_1(1-k_{12}bd) + L_2(ce-k_{23}df) + L_3(a-k_{13}bf),$$

(A.17)

where $L_1$, $L_2$ and $L_3$ are the Lagrangian multipliers, is differentiated with respect to the six variables $a - f$, and set equal to 0. This results in the following equations:

$$\frac{b}{(1+a+b)^2} + L_3 = 0 ,$$

(A.18)

$$\frac{-1-a}{(1+a+b)^2} - L_1 k_{12}d - L_3 k_{13}f = 0 ,$$

(A.19)

$$\frac{d}{(1+c+d)^2} + L_2 e = 0 ,$$

(A.20)

$$\frac{-1-c}{(1+c+d)^2} - L_1 k_{12}b - L_2 k_{23}f = 0 ,$$

(A.21)

$$\frac{f}{(1+e+f)^2} + L_2 c = 0 ,$$

(A.22)

and

$$\frac{-1-e}{(1+e+f)^2} - L_2 k_{23}d - L_3 k_{13}b = 0 .$$

(A.23)
The optimal values $\bar{a} - \bar{f}$ of the six variables $a - f$ satisfy (A.18)—(A.23). Further, these equalities show that the Lagrangian multipliers are all non-zero, implying that the constraints (A.13)—(A.15) must be satisfied with equality. Considerable algebraic manipulation of the equations yield the optimal values,

$$\bar{e} = \frac{k_{23}^{\frac{1}{2}}(k_{13}^{\frac{1}{2}} - 1)(1 - k_{12}^{\frac{1}{2}}k_{23}^{\frac{1}{2}} + k_{13}^{\frac{1}{2}})}{k_{13}^{\frac{1}{2}}(k_{23}^{\frac{1}{2}} - 1)(1 - k_{12}^{\frac{1}{2}} + k_{23}^{\frac{1}{2}} - k_{13}^{\frac{1}{2}})}, \quad (A.24)$$

$$\bar{f} = \frac{(1 - k_{23}^{\frac{1}{2}})(k_{13}^{\frac{1}{2}} - 1) - (k_{12}^{\frac{1}{2}}k_{23}^{\frac{1}{2}})(k_{13}^{\frac{1}{2}} - k_{12}^{\frac{1}{2}})}{k_{13}^{\frac{1}{2}}(k_{23}^{\frac{1}{2}} - 1)(1 - k_{12}^{\frac{1}{2}} + k_{23}^{\frac{1}{2}} - k_{13}^{\frac{1}{2}})}, \quad (A.25)$$

$$\bar{c} = (k_{13}k_{23}/k_{12})^{\frac{1}{2}} \bar{f}, \quad (A.26)$$

$$\bar{a} = (k_{13}k_{23}/k_{12})^{\frac{1}{2}} \bar{f}/\bar{e}, \quad (A.27)$$

$$\bar{d} = (k_{13}/k_{12}k_{23})^{\frac{1}{2}} \bar{e}, \quad (A.28)$$

and

$$\bar{b} = (k_{23}/k_{12}k_{13})^{\frac{1}{2}} (1/\bar{e}). \quad (A.29)$$

For the particular case of Bernoulli random variables

$$\tau_{13} = \tau_{12} \tau_{23}, \quad (A.30)$$

and hence,

$$k_{13} = k_{12} k_{23}. \quad (A.31)$$
This leads to simplification in the expressions (A.24)—(A.29):

\[
\bar{e} = \frac{(1-(k_{12}k_{23})^{\frac{1}{2}})}{k_{12}^{\frac{1}{2}}(1+k_{23}^{\frac{1}{2}})},
\]

(A.32)

\[
\bar{f} = \frac{(1+k_{12}^{\frac{1}{2}})}{k_{12}^{\frac{1}{2}}k_{23}^{\frac{1}{2}}(1+k_{23}^{\frac{1}{2}})},
\]

(A.33)

\[
\bar{c} = k_{23} \bar{f},
\]

(A.34)

\[
\bar{a} = k_{23}(\bar{f}/\bar{e}),
\]

(A.35)

\[
\bar{d} = \bar{e},
\]

(A.36)

and

\[
\bar{b} = \frac{1}{k_{12}^{\frac{1}{2}}\bar{e}}.
\]

(A.37)

(A.32)—(A.37) along with (A.9)—(A.12) yield the expressions presented in Section 3.2.1.
APPENDIX B

MINIMAX AUTOMATA FOR PROBLEM (4.31)

We argued in Section 4.3 that while there exist only $\varepsilon$-minimax automata for the compound hypothesis problem of the form

$$H_1: p \geq p_1 \quad \text{vs.} \quad H_2: p \leq p_2, \quad 1 > p_1 > p_2 > 0,$$

there exist minimax automata for the problem

$$H_1: p > p_0 \quad \text{vs.} \quad H_2: p < p_0, \quad 1 > p_0 > 0,$$

and that the automata in Figure 4.1 with the coefficient $k$ set to $(p_0/q_0)^{m-1}$ are, in fact, minimax. Since $p_C = 1/2$ for this problem, we need to show that for any $p \neq p_0$, and $\delta > 0$, the automata in this class are such that $P_e(p,\delta) \leq 1/2$.

Consider an automaton that belongs to the class depicted in Figure 4.1 with $k$ set to $(p_0/q_0)^{m-1}$ and a non-zero $\delta$. Denote by $\mu_i(p)$ the stationary probability of state $i$, for any $p$. Clearly

$$\mu_i(p) \geq 0, \quad \sum_{i=1}^{m} \mu_i(p) = 1,$$

and

$$\mu_m(p) = \mu_1(p) \cdot (pq_0/p_0q)^{m-1}.$$
Further, for any $p > p_0$,

$$P_e(p, a) = \alpha(p) = \mu_1(p) + \frac{1}{2} \sum_{i=2}^{m-1} \mu_i(p)$$

$$= \frac{1}{2}(\mu_1(p) - \mu_m(p)) + \frac{1}{2} \sum_{i=1}^{m} \mu_i(p)$$

$$= \frac{1}{2} + \frac{1}{2} \nu_1(p)(1 - (pq_0/pq_0q)^{m-1})$$

$$\leq \frac{1}{2}, \text{ since } p > p_0 \text{ and } q < q_0.$$

Similar analysis shows that $P_e(p, a) \leq \frac{1}{2}$ for any $p < p_0$. This proves the assertion that the automata in Figure 4.1 are minimax optimal and not just $\varepsilon$-minimax.
REFERENCES


