APPLICATION OF DYNAMIC PROGRAMMING METHODS TO A PROBLEM OF IDENTIFICATION BY SEQUENTIAL EXPERIMENTATION

Thesis for the Degree of Ph. D.

MICHIGAN STATE UNIVERSITY
ROBERT CARL JUOLA

1968

This is to certify that the

thesis entitled

APPLICATION OF DYNAMIC PROGRAMMING
METHODS TO A PROBLEM OF IDENTIFICATION
BY SEQUENTIAL EXPERIMENTATION

presented by

Robert Carl Juola

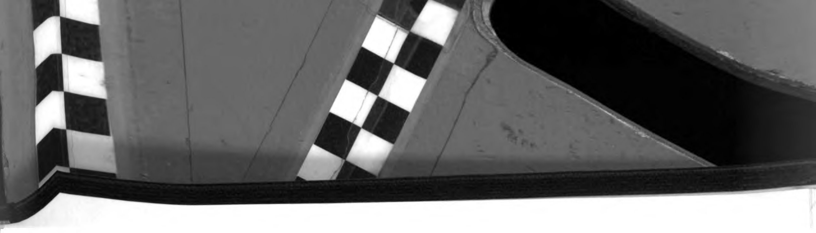
has been accepted towards fulfillment
of the requirements for

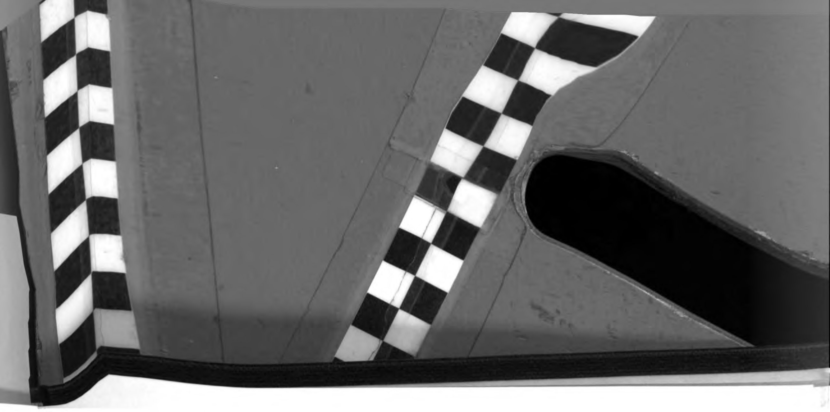
Ph.D. degree in Statistics

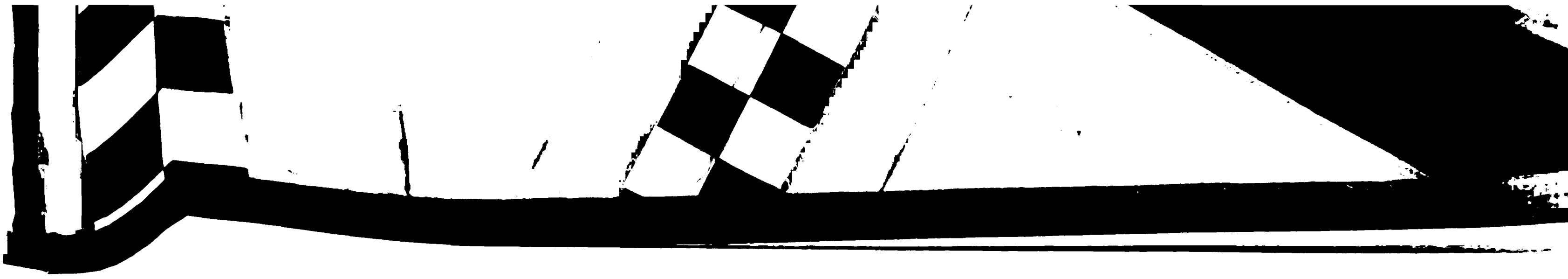
Leo Katz
Major professor

Date 23 Aug 68

O-169







ABSTRACT

APPLICATION OF DYNAMIC PROGRAMMING
METHODS TO A PROBLEM OF IDENTIFICATION
BY SEQUENTIAL EXPERIMENTATION

by Robert Carl Juola

Suppose we are given $n + 1$ urns each containing n balls of two types (to be specific we shall always refer to the two types as black balls and white balls), and further are given that the composition (the number of black balls and the number of white balls) of each of the urns is distinct from the composition of all of the other urns. That is, one of the urns contains n black balls and no white balls, a second contains $n - 1$ black balls and 1 white ball, a third contains $n - 2$ black balls and 2 white balls, and so on until the last contains n white balls and no black balls. However, we assume we are given no information about which urn contains any of the compositions.

Consider now the problem of determining with certainty the composition of all of the urns by drawing the balls randomly, without replacement, one at a time from the collection of urns. The draws are to be made from an urn of the drawer's choice, but the choice of the urn from which to draw is allowed to depend only on the numbers of black balls and white balls which have been previously drawn from each of the $n + 1$ urns.

The goal of these draws is to determine the composition of all of the urns with the fewest possible expected number of draws. The method of solution of this problem is a dynamic program.

A theorem giving upper and lower bounds for the smallest expected number of draws required to determine the composition of all of the urns is given for all n. An explicit solution is given for the smallest expected number of draws until the composition of all of the urns is determined for $n = 2$, $n = 3$, and $n = 4$. These smallest expected numbers are: 3.5 for $n = 2$, 7.528 for $n = 3$, and 13.136 for $n = 4$.

APPLICATION OF DYNAMIC PROGRAMMING METHODS TO A PROBLEM OF IDENTIFICATION BY SEQUENTIAL EXPERIMENTATION

By

Robert Carl Juola

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Statistics and Probability

1968

ACKNOWLEDGEMENTS

The author wishes to express his deep gratitude to Professor Leo Katz for suggesting the problem and for his supervision and guidance during the course of this investigation.

The work on this dissertation was carried out with the financial support of the National Science Foundation under contract no. GP 7362, and the author gratefully acknowledges this support.

TABLE OF CONTENTS

LIST OF TABLES

APPENDICES

I. INTRODUCTION

Statistical problems are mainly of two types, estimation or hypothesis testing. Here we are concerned with a third type, identification. In an hypothesis testing problem, one formulates an hypothesis and an alternative; and compares the distribution of the observed data to the distribution which the data would have if the hypothesis were true and to the distribution it would have if the alternative were true. One then rejects the hypothesis if the distribution of the observed data is not sufficiently close to the distribution if the hypothesis were true compared to the distribution if the alternative were true. In an estimation problem, a class of probability distributions which contains unknown parameters is postulated for the generation of the data, and those values of the parameters which best fit (in some pre-assigned sense) the observed data are found. In our identification problem the data are assumed to have been generated by one of a family of possible generating mechanisms and it is desired to know with certainty which of the mechanisms is the true generating mechanism for the observed data.

As stated above the problem of identification is not probabilistic at all. However if we consider the problem of amassing the data necessary to make this certain identification

sequentially, having to choose one of a number of possible experiments to obtain the next datum, the problem of finding an optimal decision strategy for obtaining the next datum is a dynamic programming problem. Further, if the experiments which are performed to obtain the next datum have random outcomes then this dynamic program has genuine stochastic elements.

We here propose a formal problem with finitely many possible generating mechanisms and ask which of these is the true mechanism. (This can be considered a special case of the multiple decision problem, in which we require the probability of making an error to be zero.)

Suppose that we are given n + 1 urns, each containing n balls of two types (to be specific we shall always refer to the two types as black balls and white balls), and further are given that the composition (the number of black balls and the number of white balls) of each of the urns is distinct from the composition of all of the other urns. That is, one of the urns contains n black balls and no white balls, a second contains n - 1 black balls and 1 white ball, a third contains n - 2 black balls and 2 white balls, and so on, to the last, containing n white balls and no black balls. Further, we assume we are given no information about which urn contains which composition.

Consider now the problem of collecting information on the composition of all of the urns by drawing the balls

randomly, without replacement, one at a time from the collection of urns. Each draw is to be made from an urn of the drawer's choice, which is allowed to depend on his information concerning the numbers of black balls and white balls which have been already drawn from each of the $n + 1$ urns.

The goal of these draws is to determine exactly the composition of all of the urns, with the smallest possible expected number of draws. The method of solution of this problem (which will be called an urn problem of order n) will be dynamic programming. This method is explained in chapter III and the solution of an urn problem of order n is presented as a dynamic program in chapter IV. A FORTRAN IV program is given in appendix 1 which utilizes the results of chapter IV in order to give an explicit solution for the urn problem of order n for small n $(n \leq 4)$.

II. NOTATION

In order to approach a solution of the problem presented above (which will be called an urn problem of order n), it is necessary to develop a rather large set of notations which will allow us to make the problem specific.

Since we will be recording the colors of the balls we have seen from each urn and must choose the urn from which to draw next, it is necessary to label the urns in some way. A convenient identification of the urns is arbitrarily to call one of them "0", one of them "1", another "2", and so on. Formally we have:

Definition II.1 The $n + 1$ urns in an urn problem of order n are indexed by $I_0^n = \{0,1,\ldots,n\}$.

The actual, but unknown, composition of the urn is characterized by the number b_j of black balls $(n - b_j$, white balls) contained in it. The reason for the choice of indexing now becomes clear; because of the assumption that the composition of each urn is distinct from that of all of the other urns, the collection $\{b_0,b_1,\ldots,b_n\}$ is a particular element of the permutation group of I_0^n. The corresponding composition of white balls is $\{n - b_0, n - b_1,\ldots, n - b_n\}$.

II. NOTATION

In order to approach a solution to the problem presented above (which will be called an urn problem of order n), it is necessary to develop a rather large set of notations which will allow us to make the problem specific.

Since we will be recording the colors of the balls we have seen from each urn and must choose the urn from which to draw next, it is necessary to label the urns in some way. A convenient identification of the urns is arbitrarily to call one of them "0", one of them "1", another "2", and so on. Formally we have:

<u>Definition II.1</u> The $n + 1$ urns in an urn problem of order n are indexed by $I_0^n = \{0, 1, \ldots, n\}$.

The actual, but unknown, composition of the urn is characterized by the number b_i of black balls ($n - b_i$ white balls) contained in it. The reason for the choice of indexing now becomes clear; because of the assumption that the composition of each urn is distinct from that of all of the other urns, the collection $\{b_0, b_1, \ldots, b_n\}$ is a particular element of the permutation group of I_0^n. The corresponding composition of white balls is

$\{n - b_0, n - b_1, \ldots, n - b_n\}$.

<u>Definition II.2</u> The set of all possible "true" compositions of the urns is S_{n+1}, the collection of all permutations of I_0^n. As is usual, we shall identify an element $\eta \in S_{n+1}$ as $\eta = (\eta_0, \eta_1, \ldots, \eta_n)$.

At any time, to summarize the information which has been obtained on past draws, we shall use a $2 \times (n+1)$ matrix M whose $(1,j)^{th}$ element is the total number of black balls previously seen from the j^{th} urn and whose $(2,j)^{th}$ element is the total number of white balls seen from the j^{th} urn. Two extreme examples of M matrices are:
1. $M = \begin{pmatrix} 0,0,\ldots,0 \\ 0,0,\ldots,0 \end{pmatrix}$, representing the information obtained prior to the first draw and 2. $M = \begin{pmatrix} n, n-1, \ldots, 0 \\ 0, \quad 1, \ldots, n \end{pmatrix}$, representing the information which could have resulted from drawing all $n(n+1)$ balls available in the urn problem of order n, if the urns had been labelled so that their true composition was indexed by $(n, n-1, n-2, \ldots, 0)$.

In order to formalize the definition of information matrices and to focus our attention on the particular subset of the set of all the $2 \times (n+1)$ matrices which we will call information matrices, we need the following definitions.

<u>Definition II.3</u> Let

$$\mathcal{M}_0 = \{M = [{}^{b}_{w}] \mid b \in \mathop{\mathrm{X}}_{i=0}^{n} I_0^n,\ w \in \mathop{\mathrm{X}}_{i=0}^{n} I_0^n, \text{ and}$$

$b + w \le e_{n+1} \cdot n\}$ *where* e_n *is the vector of* n 1's.

Definition II.4 For each $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}_0$, $H(M) = \{\eta \in S_{n+1} | b \le \eta \le n.e_{n+1} - w\}$ where $\eta = (\eta_0, \eta_1, \ldots, \eta_n)$ is considered as an $n+1$-vector. $H(M)$ will be called the M-admissible permutations.

With these definitions, $M \in \mathcal{M}_0$ is a $2 \times (n+1)$ matrix of non-negative integers whose columns total less than or equal to n and for every such M, the M-admissible permutations are the subset of S_{n+1} which corresponds to the true compositions from which it is possible to have observed M.

We shall now prove a characteristic theorem for $H(M)$, namely:

Theorem II.1 If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}_0$, and $\eta \in S_{n+1}$, then $\eta \in H(M) \Leftrightarrow \prod_{j=0}^{n} \binom{\eta_j}{b_j}\binom{n-\eta_j}{w_j} > 0$ where $\binom{a}{b}$ is the usual binomial coefficient.

Proof: $\eta \in H(M) \Leftrightarrow b \le \eta \le n.e_{n+1} - w$, where $\eta = (\eta_0, \eta_1, \eta_2, \ldots, \eta_n)$ is considered an $n+1$ vector.

$$\Leftrightarrow \eta_j \ge b_j \ge 0 \quad \text{and} \quad n - \eta_j \ge w_j \ge 0 \quad \text{for all } j$$

$$\Leftrightarrow \binom{\eta_j}{b_j} > 0 \quad \text{and} \quad \binom{n-\eta_j}{w_j} > 0 \quad \text{for all } j$$

$$\Leftrightarrow \prod_{j=0}^{n} \binom{\eta_j}{b_j}\binom{n-\eta_j}{w_j} > 0.$$

Among the matrices in $\mathcal{M}_0$, we shall restrict our attention to the subclass of all those M with the property that H(M) is non-empty. This is the set of matrices for which there exists at least one true composition for which it is possible to have observed, through draws from the urns, the b vector of black balls and the w vector of white balls.

Definition II.5 M is an information matrix if $H(M) \neq \phi$. The set of all information matrices will be denoted $\mathcal{M}$.

If $M \in \mathcal{M}$, we may now say that there exists at least one actual, but possibly unknown, composition from which it is possible to have observed the information matrix M and then to identify H(M) as the set of permutations which index true compositions that have not been ruled impossible by observing M.

The choice of a prior distribution whose support is the whole of S_{n+1} will allow us to identify H(M) as the support of the posterior distribution on S_{n+1} after observing M. We will choose a uniform prior distribution to reflect our initial assumption of complete uncertainty about the actual composition of any of the urns.

Definition II.6 $P_0(\eta) = \frac{1}{(n+1)!}$ for every $\eta \in S_{n+1}$, the uniform distribution on S_{n+1}.

With this choice of a prior distribution, it is now possible to give explicit formulae for the calculation of the posterior distributions.

<u>Theorem II.2</u> If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}$, and $\pi \in S_{n+1}$, then

$$P_M(\pi) = P(\pi|M) = \frac{\prod_{j=0}^{n} \binom{\pi_j}{b_j}\binom{n-\pi_j}{w_j}}{\sum_{\eta\in H(M)} \prod_{k=0}^{n} \binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}}$$

<u>Proof</u>:

$$P_M(\pi) = \frac{P(M|\pi)P_0(\pi)}{\sum_{\eta\in S_{n+1}} P(M|\eta)P_0(\eta)}$$

$$= \frac{\prod_{j=0}^{n} \dfrac{\binom{\pi_j}{b_j}\binom{n-\pi_j}{w_j}}{\binom{n}{b_j+w_j}(n+1)!}}{\sum_{\eta\in S_{n+1}}\left[\prod_{k=0}^{n} \dfrac{\binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}}{\binom{n}{b_k+w_k}(n+1)!}\right]}$$

$$= \frac{\prod_{j=0}^{n} \binom{\pi_j}{b_j}\binom{n-\pi_j}{w_j}}{\sum_{\eta\in S_{n+1}}\left[\prod_{k=0}^{n} \binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}\right]}$$

$$= \frac{\prod_{j=0}^{n} \binom{\pi_j}{b_j}\binom{n-\pi_j}{w_j}}{\sum_{\pi\in H(M)}\left[\prod_{k=0}^{n} \binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}\right]}$$

since $\pi \notin H(M) \Leftrightarrow \prod_{J=0}^{n} \binom{b_j}{\pi_j}\binom{w_j}{n-\pi_j} = 0.$

It is not necessary to consider all of the matrices in $\mathfrak{M}$. It often occurs that for a given matrix $M \in \mathfrak{M}$, it is possible to immediately conclude that more is known about the composition of the urns than is clearly shown by the matrix M. For example, consider $M = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$; it is known that urn 3 contains all white balls since urns 0, 1 and 2 all contain at least one black ball. Thus, the matrix $M' = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$ exhibits the same knowledge about the composition of all of the urns as does M, but M' exhibits this knowledge more clearly.

If it were not for this possibility of immediately concluding that more is known about the composition of the urns than is clearly shown by the matrix M, there would be no problem in minimizing the number of draws until the composition of all of the urns is known. We would be forced to draw all of the balls in the entire system. However, this is not the case; we know, in fact, that if the composition of n of the urns is known, then the composition of the remaining urn is also known. This causes an immediate reduction in the total number of draws until the composition of all the urns is known from $n(n+1)$ to at most n^2.

The fact that this ability on our part to be able to conclude more knowledge about the composition of the urns

can and does occur at many other times in the process of drawing the balls provides us with the basic tool to find the minimum expected number of draws until the composition of all of the urns is known.

The matrix $M = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$ exhibits the same knowledge about the composition of the urns as do the matrices $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$, $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$, and $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$, but it shows this knowledge more clearly. With this in mind we will call $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$, $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$, and $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$ reducible matrices and call $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$ an irreducible matrix; and finally, we shall call $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$ the reduction of $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$ and will denote this reduction by $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} = R\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$; similarly $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} = R\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = R\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$. Formally, we have the following definition.

<u>Definition II.7</u> An information matrix $M = \begin{bmatrix} b \\ w \end{bmatrix}$ is reducible if there exists $j \in I_0^n$ such that for every permutation η in $H(M)$, η_j is equal to some fixed integer k, but $b_j + w_j < n$. If M is not reducible, it will be called irreducible. The class of irreducible matrices will be denoted $\mathcal{M}^*$.

To illustrate this definition, let us consider the following two examples:

<u>Example I</u>. $M = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$; $H(M) = \{(3,2,1,0), (3,1,2,0), (2,3,1,0), (2,1,3,0), (1,3,2,0), (1,2,3,0)\}$. Each of the permutations in $H(M)$ has the last coordinate equal to zero. But $b_3 + w_3 < n$, and so M is reducible. Similarly,

$M' = \binom{1\ 1\ 1\ 0}{0\ 0\ 0\ 1}$, $M'' = \binom{1\ 1\ 1\ 0}{0\ 0\ 0\ 2}$ are *reducible. But* $M''' = \binom{1\ 1\ 1\ 0}{0\ 0\ 0\ 3}$ is irreducible.

Example II. $M' = \begin{bmatrix} b' \\ w' \end{bmatrix} = \binom{3\ 0\ 1\ 0}{0\ 3\ 1\ 0}$ and $M'' = \begin{bmatrix} b'' \\ w'' \end{bmatrix} = \binom{3\ 0\ 0\ 0}{0\ 3\ 0\ 0}$; $H(M') = H(M'') = \{(3,0,2,1), (3,0,1,2)\}$. If $\eta \in H(M')$, then $\eta_0 = 3$ and $\eta_1 = 0$, and $b'_0 + w'_0 = b'_1 + w'_1 = n$, and $b''_0 + w''_0 = b''_1 + w''_1 = n$, so that both M' and M'' are irreducible, although they are different and induce the same posterior distribution on S_{n+1}, namely:

$$P_{m'}((3,0,2,1)) = P_{m''}((3,0,2,1)) = P_{m'}((3,0,1,2)) = P_{m''}((3,0,1,2)) = 1/2.$$

Every reducible matrix M, has corresponding to it an irreducible matrix, $R(M)$, which exhibits all of the knowledge about the composition of the urns that M does. This reduction of a reducible matrix to an irreducible matrix preserves the posterior distribution on S_{n+1} induced by M, and therefore, the conditional probability of drawing a black (or white) ball from any of the urns whose composition is not already known. This will be proven in the theorems following the formal definition of the reduction of an information matrix.

Definition II.8 If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}$, then $R[M] = \begin{bmatrix} b' \\ w' \end{bmatrix}$, where:

1. If M is irreducible, $b' = b$, and $w' = w$.
2. If M is reducible, and for every $\eta \in H(M)$

there exist non-negative integers $i_0, i_1, \ldots, i_J$ and $k_0, k_1, \ldots, k_J$ for $0 \leq J \leq n$ such that

$\eta_{i_j} = k_j$ for $j = 1,2,\ldots,J$,

$b'_\ell = b_\ell$ and $w'_\ell = w_\ell$ for $\ell \neq i_1, i_2, \ldots, i_J$ and

$b'_{i_j} = k_j$ and $w'_{i_j} = n - k_j$ for $j = 1,2,\ldots,J$.

The justification for considering only irreducible matrices is that if we have an information matrix M which has the property that every permutation in H(M) has the same j^{th} coordinate (say π_j) then no matter what the true composition of all of the urns is, the j^{th} urn is known to have π_j black balls and $n - \pi_j$ white balls. Since we know the composition of the j^{th} urn, there is nothing to be lost by letting the information matrix reflect this additional knowledge more completely.

Theorem II.3 M is an irreducible information matrix if and only if $M = R(M)$.

Proof: Direct verification of definitions II.7 and II.8.

The following two theorems will show that for any $M \in \mathcal{M}$, the corresponding $R(M)$ induces the same posterior distribution on S_{n+1} as M does.

Theorem II.4 If $M = \left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right] \in \mathcal{M}$ then $H(R(M)) = H(M)$.

Theorem II.5 If $M = \left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right] \in \mathcal{M}$, then $P_M(\eta) = P_{R(M)}(\eta)$ for every $\eta \in S_{n+1}$.

there exist non-negative integers [illegible]

and $k_0, k_1, \ldots, k_j$ for [illegible] such that

$\eta_{i_j} = k_j$ for [illegible]

$b'_\ell = b_\ell$ and w'_ℓ [illegible]

$b'_{i_j} = k_j$ and w'_{i_j} = [illegible]

The justification for [illegible]

matrices is that if we have an information matrix M which

has the property that every [illegible] has the

same j^{th} coordinate (say η_j) then no matter what the true

composition of all of the urns is, the j^{th} urn is known to

have η_j black balls and $n - \eta_j$ white balls. Since we

known the composition of the j^{th} urn, there is nothing to be

lost by letting the information matrix reflect this additional

knowledge more completely.

<u>Theorem II.3</u> M is an irreducible information matrix if and

only if $M = R(M)$.

<u>Proof:</u> Direct verification of definitions II.7 and II.8.

The following two theorems will show that for any

$M \in \mathcal{M}$, the corresponding $R(M)$ induces the same posterior

distribution on S_{n+1} as M does.

<u>Theorem II.4</u> If $M = \begin{bmatrix} a \\ b \end{bmatrix} \in \mathcal{M}$ then $H(R(M)) = H(M)$.

<u>Theorem II.5</u> If $M = \begin{bmatrix} a \\ b \end{bmatrix} \in \mathcal{M}$, then $P_M(\eta) = P_{R(M)}(\eta)$ for

every $\eta \in S_{n+1}$.

The proofs of these two theorems will be indirect as there is a considerable overlapping if we prove both of them directly. The proof will consist of three steps:

A. $H(M) \supseteq H(R(M))$, B. Theorem II.5, and finally

C. $H(M) \subseteq H(R(M))$.

Proof of A: 1. If $M = R(M)$ the theorem is true.

2. If $M = [{b \atop w}] \neq R[M] = [{b' \atop w'}]$ then $b' \geq b$ and $w' \geq w$ and $\eta \in H(R(M)) \Rightarrow$

$b' \geq \eta \geq n \cdot e_{n+1} - w'$ where η is considered as an $n+1$ vector.

$\Rightarrow b \leq \eta \leq n \cdot e_{n+1} - w$ where η is considered as an $n+1$ vector.

$\Rightarrow \eta \in H(M)$

Proof of B: 1. If $M = R(M)$ the theorem is true.

2. If $M \neq R(M)$ and $\eta \notin H(M)$, then $\eta \notin H(R(M))$ and hence $P_M(\eta) = P_{R(M)}(\eta) = 0$

3. If $M = [{b \atop w}] \neq R(M)$ and $M \in \mathcal{M}$, then there exists $i \in I_0^n$ and $j \in I_0^n$ such that $\eta_i = j$ for every $\eta \in H(M)$, then for every $\eta \in H(M)$

$$P_M(\eta) = \frac{\prod_{\ell=0}^{n} \binom{\eta_\ell}{b_\ell} \binom{n-\eta_\ell}{w_\ell}}{\sum_{\pi \in H(M)} \left[\prod_{k=0}^{n} \binom{\eta_k}{b_k} \binom{n-\eta_k}{w_k} \right]}$$

$$= \frac{\left\{\prod_{\ell \neq i} \binom{\eta_\ell}{b_\ell}\binom{n-\eta_\ell}{w_\ell}\right\}\binom{\eta_i}{b_i}\binom{n-\eta_i}{w_i}}{\left\{\sum_{\pi \in H(M)} \prod_{k \neq i} \binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}\right\}\binom{\eta_i}{b_i}\binom{n-\eta_i}{w_i}}$$

$$= \frac{\left\{\prod_{\ell \neq i} \binom{\eta_\ell}{b_\ell}\binom{n-\eta_\ell}{w_\ell}\right\}\binom{j}{j}\binom{n-j}{n-j}}{\left\{\sum_{\pi \in H(M)} \prod_{k \neq i} \binom{\eta_k}{b_k}\binom{n-\eta_k}{w_k}\right\}\binom{j}{j}\binom{n-j}{n-j}}$$

The proof of B is now complete since we may repeat the above calculation for any other pair (ℓ, p) such that $\eta_\ell = p$ for all $\eta \in H(M)$ and obtain the result

$$P_M(\eta) = P_{R(M)}(\eta)$$

<u>Proof of C</u>: If $M \in \mathcal{M}^*$, then $\eta \in H(M) \Rightarrow P_M(\eta) > 0 \Rightarrow$

$$P_{R(M)}(\eta) > 0 \Rightarrow \eta \in H(R(M)).$$

The proof of both theorems is now complete.

Since we shall have a choice of the urn from which to draw next, after observing the results of previous draws, we shall denote by $D(M,j)$ the act of drawing from the j^{th} urn when the past information is M. We shall denote the random variable resulting from $D(M,j)$ by $D_j[M]$. $D(M,j)$ must result in one of two results, namely, either a black ball is drawn or a white ball is drawn. For completeness of definitions, if $D(M,j)$ is chosen and if n balls have

The proof [illegible]

the above calculation [illegible]

$T_k = p$ for all [illegible]

[illegible]

Proof of C: [illegible]

[illegible]

The proof of both [illegible]

Since we shall have a choice of the urn from which

to draw next, after observing the [illegible] of previous draws,

we shall denote by $D(M,j)$ [illegible]

urn when the past information is M. [illegible]

random variable resulting from $D(M,j)$ [illegible]

must result in one of two results, [illegible]

ball is drawn or a white ball is drawn. For completeness of

definitions, if $D(M,j)$ is chosen and if n balls have previously been drawn from the j^{th} urn we will say $D_j(M) = M$.

Definition II.9 Let $\alpha_j = \begin{bmatrix} \delta_{j,0}, \delta_{j,1}, \ldots, \delta_{j,n} \\ 0 \quad ,0 \quad ,\ldots,0 \end{bmatrix}$ and

$\beta_j = \begin{bmatrix} 0 \quad ,0 \quad ,\ldots,0 \\ \delta_{j,0}, \delta_{j,1}, \ldots, \delta_{j,n} \end{bmatrix}$, where $\delta_{i,j} = 0$ if $i \neq j$

and $\delta_{i,i} = 1$.

Definition II.10 If $M \in \mathcal{M}^*$ and $0 \le b_j + w_j < n$, then $M_j^+ = R[M + \alpha_j]$ and $M_j^- = R[M + \beta_j]$.

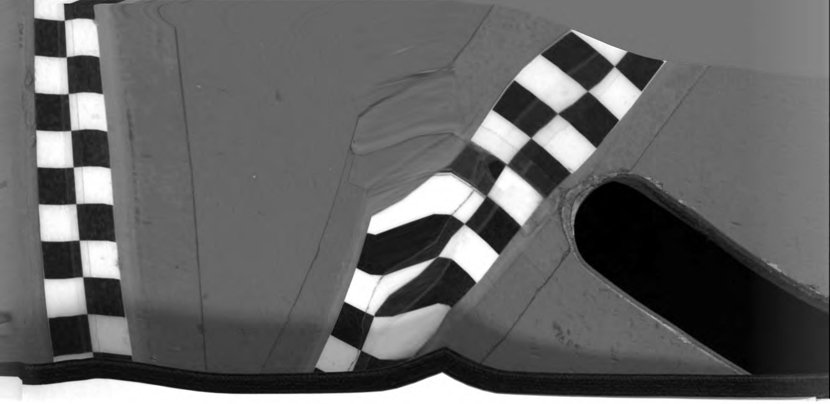
Theorem II.6 If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}^*$ and $b_j + w_j < n$, let $P_j(B|M)$ denote the conditional probability of drawing a black ball from the j^{th} urn, after the M matrix of black balls and white balls have been previously drawn. Then

$$P_j(B|M) = \frac{\sum_{\eta \in H(M)} \frac{\eta_j - b_j}{n - b_j - w_j} \prod_{k=0}^{n} \binom{\eta_k}{b_k} \binom{n-\eta_k}{w_k}}{\sum_{\pi \in H(M)} \prod_{\ell=0}^{n} \binom{\pi_\ell}{b_\ell} \binom{n-\pi_\ell}{w_\ell}} .$$

Proof:
$$P_j(B|M) = \sum_{\eta \in S_{n+1}} \frac{\eta_j - b_j}{n - b_j - w_j} P_M(\eta)$$

and apply theorem II.2.

Definition II.11 A choice $D(\cdot,\cdot)$ is a random mapping from $\mathcal{M}^* \times I_0^n$ into $\mathcal{M}^*$ given by:



1. If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}^*$ and $b_j + w_j < n$, then

$$D_j(M) = \begin{cases} M_j^+ & \text{with probability } P_j[B|M] \\ M_j^- & \text{with probability } 1 - P_j[B|M], \end{cases}$$

or 2. If $M = \begin{bmatrix} b \\ w \end{bmatrix} \in \mathcal{M}^*$ and $b_j + w_j = n$, then $D_j(M) = M$ with probability one.

This chapter has provided us with the tools and conventions which will now be used in order to present a technique for solving an urn problem of order n. This proposed technique is dynamic programming and is discussed in chapter III. Chapter IV then presents a specific dynamic program for the urn problem of order n, which is used to solve the several examples of an urn problem of order n, namely $n = 2$, $n = 3$, and $n = 4$.

III. DYNAMIC PROGRAMMING

Dynamic programming is a mathematical technique which is often useful for making a series of interrelated decisions. When it is applicable to a problem, it provides a systematic procedure for determining the combination of decisions which will maximize the overall effectiveness of those decisions in striving for some fixed goal.

There are a number of problem types for which a dynamic programming formulation of the problem is useful. We will state for our problem a specific set of conditions under which a dynamic program is possible and at the same time introduce the standard terminology of dynamic programming.

Following Bellman [1], these conditions may be stated for the urn problem as follows:

1. The decision problem may be divided into stages, each of which can be characterized by a "small" set of parameters called state variables. In the urn problem we are concerned with, the information matrices M are the state variables.

2. At each stage, the statistician has the choice of a number of possible experiments. In our case, he has the choice of drawing from one of the $n+1$ urns; the act

of drawing from urn j is denoted $D(\cdot,j)$ for $j = 0,1,\ldots,n$.

3. The effect of a chosen experiment is a transformation of the state variables. The act $D(M,j)$ results in a transformation of the state variables to either M_j^+ or M_j^-.

4. The past history of the system has no importance in determining the future decisions, except through the present.

5. The purpose of the decision process is to minimize some fixed function of the state variables. In our case, the goal is to minimize the expected number of experiments made until a terminal state is reached.

In finite decision problems, those with only finitely many possible decisions until termination, we have the additional parameter of time. This parameter manifests itself in the form of the number of permissible decisions which remain to be made in the problem. In our problem, time is the number of decisions to be made until a terminal state is reached. It is usually helpful to separate the time parameter from the state variables as time usually plays a role in the problem which is quite different from the state variables. In our problem it is the parameter upon which the objective function depends.

Standard terminology in dynamic programming is: a policy is a rule for choosing, for each value of the state

variables and the time parameter, an experiment.

An optimal policy is a policy which minimizes a pre-assigned function of the state variables. This pre-assigned function of the final state variables will be called the criterion function and will be denoted G.

In the case that the decision results in a probability distribution on the transformations, it is not possible to minimize with certainty the criterion function after N steps. Rather, the quantity which we will seek to minimize is the expected value of the criterion function.

With the structure on the problem as above, the possibility of constructing an optimal policy rests on the following principle due to Bellman [1].

Bellman's Principle of Optimality: An optimal policy has the property that whatever the initial values of the state variables and the initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision, treated as new initial values of the state variables.

Implementation of an optimal policy repeatedly utilizes the principle of optimality to consider a series of decisions backwards, evaluating each decision on the basis of proceeding optimally from whatever state has resulted from previous decisions. This works as follows: for each possible state of nature prior to the final decision, we answer the question "What is the best decision to make if we are forced to stop now or to stop at whatever state results from our decision?" This can be called the final-step decision problem. Knowing the answer to the final-step decision problem for each state of nature, we now ask "What is the best decision to make at each stage from every state if we are permitted at most two more decisions?", and so on backward until we evaluate the best decision to make if we are permitted N decisions. In the urn problem of order n, with only $n(n+1)$ balls available in the system, and with each decision to draw a ball removing one ball from the system, it is obvious that $N = n(n+1)$ will be a sufficiently large number to solve the problem, since it exhausts the system.

IV. THE URN PROBLEM AS A DYNAMIC PROGRAM

The goal of determining the composition of all of the urns can be expressed as an hypothesis testing problem as follows: given the $(n+1)!$ a priori equally likely hypothesis find a policy, i.e., a function from all information matrices M^* into $\{0,1,\ldots,n\}$, to test the $(n+1)!$ hypotheses at size zero and power one. The goal of testing these hypotheses with minimum expected sample size is now: among all policies which test the hypotheses at size zero and power one find a policy whose expected sample size to termination is the minimum.

In the following, all matrices M will be elements of $\mathcal{M}^*$ and if M is not an element of $\mathcal{M}^*$, we shall identify M with $R(M)$. This should cause no ambiguity or confusion since it will not affect the posterior distribution or the probabilities of drawing a black ball from any of the urns whose composition is not already known.

<u>Definition IV.1</u> M is a terminal information matrix if $H(M)$ is a singleton.

<u>Definition IV.2</u> If P is any policy, let $E_P(\nu_n)$ be the expected number of draws until a terminal information matrix is reached in an urn problem of order n.

IV. THE UR[illegible]

The goal of determ[illegible] urns can be expres[illegible] follows: given the [illegible] find a policy, i.e. a [illegible] M^* into [0,1, [illegible] zero and power one [illegible] with minimum expected [illegible] which test the hypotheses [illegible] is the policy whose expected sample [illegible] minimum.

In the following, all matri[illegible] identify of $\mathcal{M}^*$ and if M is not an ele[illegible] M with $\mathcal{E}(M)$. This should cause no ambi[illegible] since it will not affect the posterior distribut[illegible] the probabilities of drawing a black ball from any of the urns whose composition is not already known.

<u>Definition IV.1</u> M is a terminal information matrix if $H(M)$ is a singleton.

<u>Definition IV.2</u> If P is any policy, let $\mathcal{E}_P(v_n)$ be the expected number of draws until a terminal information matrix is reached in an urn problem of order n.

Definition IV.3 O_n - $\mathrm{Min}_{\mathcal{P}} E_P(\nu_n)$ where $\mathcal{P}$ is the class of all possible policies.

We can find an upper and a lower bound on O_n, both of the bounds being of the order of n^2.

Theorem IV.1 $\binom{n+1}{2} \le O_n \le n^2$.

Proof: The upper bound is immediate since if we know the composition of n of the urns, we know the composition of the remaining urn. The composition of n of the urns can be found by simply drawing all of the balls in these urns. This policy requires drawing n^2 balls. To establish the lower bound, we need a lemma.

Lemma IV.2 If X,Y are n+1 vectors of integers between O and n inclusive and is there exists one and only one permutation π satisfying the inequalities $e_{n+1}0 \le X \le \pi \le Y \le e_{n+1}n$, where π is considered as a n+1 vector, then

$$\sum_{i=0}^{n} \cdot (Y_i - X_i) \le \frac{n(n+1)}{2}.$$

Proof of Lemma: Without loss of generality, $\pi_j = j$ since if it is not, we can reorder the X's and Y's, and $\sum_{i=0}^{n} (Y_i - X_i)$ is independent of the order of the subscripts on the X's and Y's.

Now, if $i < j$, either $Y_i < j$ or $X_j > i$, for if

not, it is possible to interchange the i^{th} and j^{th} coordinates of the permutation and preserve the inequalities of the Lemma.

Consider Y_0, $0 \le Y_0 \Rightarrow X_1, X_2, \ldots, X_{Y_0}$ are all greater than 0, which may be a vacuous relationship.

Now, consider Y_1, $1 \le Y_1 \Rightarrow X_2, X_3, \ldots, X_{Y_1}$ are all greater than 1. Similarly, $j \le Y_j \Rightarrow X_{j+1}, X_{j+2}, \ldots, X_{Y_J}$ are all greater than j. Thus a block of $Y_j - j$ of the X's are greater than or equal to $j+1$ for $j = 0,1,\ldots,n-1$.

We then have

$$\begin{aligned}
\sum_{j=0}^{n} X_i &= \sum_{k=0}^{n} k[\#X\text{'s} = k] \\
&= \sum_{k=0}^{n} k[\#X\text{'s} \ge k] - \sum_{k=0}^{n-1} k[\#X\text{'s} \ge k+1] \\
&= \sum_{k=0}^{n} k[\#X\text{'s} \ge k] - \sum_{k=0}^{n-1} (k+1)[\#X\text{'s} \ge k+1] \\
&\qquad + \sum_{k=0}^{n-1} [\#X\text{'s} \ge k+1] \\
&= \sum_{k=0}^{n-1} [\#X\text{'s} \ge k+1] \\
&\ge \sum_{k=0}^{n-1} [Y_j - j] = \sum_{k=0}^{n} [Y_j - j], \text{ since } Y_n = n,
\end{aligned}$$

hence $\sum_{j=0}^{n} (Y_j - X_j) \le \sum_{j=0}^{n} (Y_j - Y_j + j) = \sum_{j=0}^{n} j = \binom{n+1}{2}$.

This completes the proof of the Lemma.

We shall be using the contrapositive of the Lemma in the remainder of the proof.

<u>Corollary IV.3</u> If X and Y are n+1 vectors of integers 0 to n inclusive such that $\sum_{i=0}^{n} (Y_i - X_i) > \binom{n+1}{2}$ then the set of permutations $\pi \in S_{n+1}$ satisfying $X \leq \pi \leq Y$ where π is considered as an n+1 vector is either empty or has at least two elements.

Suppose $M = \begin{bmatrix} b \\ w \end{bmatrix}$ is a matrix such that $\sum_{j=0}^{n} (b_j + w_j) < \binom{n+1}{2}$ then $\sum_{j=0}^{n} (n - w_j - b_j) > \binom{n+1}{2}$ which implies by the corollary that either $H(M) = \phi$ or $H(M)$ has at least two elements; in either case M is not terminal.

The proof of the theorem is now complete since if fewer than $\binom{n+1}{2}$ balls have been observed, the resulting information matrix cannot be terminal.

We can now express the solution of the urn porblem of order n as a dynamic program. For every information matrix $M \in \mathcal{M}^*$, define the function $g(M) = 0$ if M is a terminal matrix; and $g(M) = 1$ if M is not a terminal matrix.

We would like to minimize for every $M_0 \in \mathcal{M}^*$ the criterion function $G_N(M_0) = \sum_{j=0}^{N} g(M_i)$, for $N \geq n(n+1)$, where M_i is the state resulting from the i^{th} act on the state M_{i-1}. This is impossible since the state resulting from the i^{th} act is a random variable whose value depends on the entire sequence of decisions which have been made prior to the i^{th} decision and also on the randomness in the occurrence of a black ball or white ball in drawing from any urn.

We shall adopt the convention of using $\hat{M}_j$ to denote the random variable resulting from the j^{th} decision on state M_{j-1}. With this notation we shall minimize the following criterion function

$$f_N(M_0) = g(M_0) + E(\sum_1^N g(\hat{M}_i)) \qquad (1)$$

The principle of optimality requires that every optimal policy satisfy the following recursion relation:

$$f_N(M) = g(M) + \min_q [f_{N-1}(M_q^+)P_q(B|M) + f_{N-1}(M_q^-)(1-P_q(B|M)] \qquad (2)$$

where q ranges from 0 to n inclusive and $P_q(B|M)$ is given by Theorem II.6.

<u>Theorem IV.4</u> If P is an optimal policy which chooses the act $D_j(M)$ for an $M = [{b \atop w}]$ such that $b_j + w_j = n$, then M is terminal.

<u>Proof</u>: Consider $f_k(M)$ for $k > n(n+1)$. The optimality of P requires that $f_k(M) = g(M) + f_{k-1}(M)$ but k is sufficiently large so that $f_k(M) = f_{k-1}(M)$ which implies $g(M) = 0$. But $g(M) = 0$ if and only if M is terminal.

This theorem allows us to reduce the number of acts which have to be examined in order to find an optimal policy. We need not look at those acts which would have us draw from an urn whose contents are already known.

In order to implement the dynamic program on a digital computer, it was found convenient to further restrict the class $\mathcal{M}^*$ of all irreducible information matrices and to impose an ordering on this restriction of $\mathcal{M}^*$. This ordering and restriction are of no value in deriving the dynamic program but they do serve to reduce the amount of calculation and the amount of core storage required to implement the dynamic program on a computer.

Convention IV.1 For every information matrix $M = [{b \atop w}] \in \mathcal{M}^*$ relabel the urns until the columns of M satisfy the following conditions:

1. $b_i + w_i \geq b_{i+1} + w_{i+1}$ for $i = 0,1,2,\ldots,n-1$ and
2. if $b_i + w_i = b_{i+1} + w_{i+1}$ then $b_i \geq b_{i+1}$ for $i = 0,1,2,\ldots,n-1$.

In the remainder, all information matrices will be written in this way.

Definition IV.4 Let $\gg$ be the ordering on $\mathcal{M}^*$ given by:

$$M^* = [{b^* \atop w^*}] \gg M = [{b \atop w}]$$

if for some integer $k \leq n$ either

1. $b_j + w_j = b_j^* + w_j^*$ for $j \leq k-1$ and $b_k + w_k < b_k^* + w_k^*$

or 2. $b_j + w_j = b_j^* + w_j^*$ for $j = 0,1,\ldots,n$ and

$b_j = b_j^*$ for $0 \le j \le k-1$ and

$b_k < b_k^*$.

Let us use the case $n = 2$ as an example to illustrate this ordering.

$$M_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad M_7 = \begin{pmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \end{pmatrix}$$

$$M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad M_8 = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$M_3 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad M_9 = \begin{pmatrix} 0 & 1 & 0 \\ 2 & 0 & 0 \end{pmatrix}$$

$$M_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad M_{10} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

$$M_5 = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad M_{11} = \begin{pmatrix} 0 & 1 & 1 \\ 2 & 0 & 0 \end{pmatrix}$$

$$M_6 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \qquad M_{12} = \begin{pmatrix} 2 & 1 & 0 \\ 0 & 1 & 2 \end{pmatrix}$$

The matrix $\begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ does not appear in the ordering since by previous conventions it is identified with $R\begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ which is $\begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$ which is reordered to $\begin{pmatrix} 0 & 1 & 1 \\ 2 & 0 & 0 \end{pmatrix}$ by convention IV.1 and now can be found in the ordering as M_{11}.

The value of these conventions in reducing the number of matrices which must be considered can be demonstrated with a few simple calculations.

The set $\mathcal{M}_0$ of all $2 \times (n+1)$ matrices of non-negative integers whose columns total less than or equal to n has $(2^{n+1} - 1)^{n+1}$ elements. This can be seen by observing that the

number of ways that two non-negative integers whose sum is less than or equal to n can be chosen in $\sum_{k=0}^{n} \sum_{j=0}^{k} \binom{k}{j}$ ways and $\sum_{k=0}^{n} \sum_{j=0}^{k} \binom{k}{j} = \sum_{k=0}^{n} 2^k = 2^{n+1} - 1$. Since each of the columns of a matrix in $\mathcal{M}_0$ can be chosen in $2^{n+1} - 1$ ways the total number of elements in $\mathcal{M}_0$ is $(2^{n+1} - 1)^{n+1}$.

The set $\mathcal{M} = \{M \in \mathcal{M}_0 | H(M) \neq \phi\}$ has at least $((n+1)!)^2$ elements. This is seen by observing that for every permutation $\eta = (\eta_0, \ldots, \eta_n) \in S_{n+1}$ the total number of ways that non-negative integers $\{b_i\}_{i=0}^{n}$ and $\{w_i\}_{i=0}^{n}$ satisfying $0 \leq b_i \leq \eta_i$ and $0 \leq n - \eta_i$ for $i = 0,1,\ldots,n$, can be chosen is $\prod_{j=0}^{n} (\eta_j + 1)(n - \eta_j + 1)$ ways.

The definition of $\mathcal{M}$ does not consider either the reducibility of an M matrix or the relabelling of the columns of the M matrix to eliminate redundancies caused by the initial labelling of the urns. The consideration of either of these alone will result in fewer matrices to consider. It was found to be difficult to calculate the magnitude of this reduction caused by either of these considerations alone. The imposition of both reduction and relabelling leads us to the class $\mathcal{M}^*$ of definition II.7.

The number of elements in $\mathcal{M}^* \subset \mathcal{M}_0$ has been found empirically for $n = 2,3,4$, by enumerating them. No general formula for the number of elements in $\mathcal{M}^*$ is known.

The above calculations are summarized in the following table.

	n=2	n=3	n=4	n=k
Number of elements in $\mathcal{M}_0$	334	47,875	$(31)^5$	$(2^{k+1}-1)^{k+1}$
Number of elements in $\mathcal{M}$	>36	>576	>14,400	$>((k+1)!)^2$
Number of elements in $\mathcal{M}^*$	12	122	1746	?

The dynamic program is now implemented by considering first, candidates for the last M. It is a terminal position so that $f_0(M) = 0$. Now, consider the next to the last M. The only possible image under any optimal policy is the last M so that $f_1(M) = 1$.

Since the ordering was chosen in such a way that for any $M \in \mathcal{M}^*$ all of the M_j^+ and M_j^-, for $j = 0,1,\ldots,n$, are higher in the ordering than was M, $f_{k+1}(M)$ is calculable as $\min_{j \in I_0^n} [1 + P_j(B|M) \cdot f_k(M_j^+) + (1 - P_j(B|M)) \cdot f_k(M_j^-)]$. When we have calculated $f_\ell\begin{pmatrix}0 & 0 & 0\\ 0 & 0 & 0\end{pmatrix}$, for some $\ell \geq n^2$, we have calculated the minimum expected number of draws to reach the terminal position.

Appendix 1 contains a FORTRAN IV computer program to utilize the above method to determine the minimum expected number of draws to the terminal information matrix for small n, (n = 2,3,4).

Table 1 at the end of this chapter (page 33) illustrates this dynamic program for the case n = 2 and table 2 (page 34) presents the case n = 2 as a tree diagram.

The dynamic program for $n = 3$ is presented as a table in Appendix 2. Since there are 122 information matrices for the case $n = 3$, no tree diagram is presented.

It is a very difficult task to write down explicitly an optimal policy for solving an urn model of order n. However, it is possible to give a general rule for the first $\binom{n-1}{2} + 1$ moves of an optimal strategy, for all n. If we use this strategy for the first $\binom{n-1}{2} + 1$ moves it is possible for small n, to give a complete description of an optimal policy.

The rule goes as follows:

First, draw one ball from each of n urns. Suppose this results in k white balls and $n - k$ black balls, then draw 1 more ball from each of $k - 1$ urns from which white balls have been seen and 1 more ball from each of $n-k-1$ urns from which a black ball has been seen. The urns from which 2 balls have been drawn have 3 possible compositions: WW, WB, or BB. Now from each of the urns of equally seen composition, break all ties by drawing one more ball from all but one of tied urns, and continue this process of breaking ties until the observed composition of all of the urns is distinct.

The reason that this process is the start of an optimal policy is two-fold: 1. no position is a terminal position if the observations from 2 urns are the same, and

2. no draws from a 3rd urn will ever distinguish between two tied urns. Since the tie will eventually have to be broken to result in a terminal position, and since no draw except from one of the tied urns will ever distinguish them, the rule of the preceeding paragraph is simply to break all ties as soon as possible.

This rule gives at least the first $\frac{n(n-1)}{2} + 1$ moves since, for every j, if we have drawn j balls from an urn there are at most $j + 1$ possible distinct patterns of black and white balls that can be observed. Drawing 1 ball from each of n urns takes n draws, and at most two compositions can result (black ball seen or white ball seen). Drawing one more ball from all but one of the urns from which a black ball has been seen, and one more from all but one of the urns from which a white ball has been seen requires at least $n - 2$ draws (it is possible that all of the balls seen are the same color). Now there are at most 3 groups of ties, and breaking them takes at least $n-2-3$ draws. Continuing, we see that the total number of draws until all ties are broken is at least

$$n + \sum_{j=1}^{n-1} (n-j-1) = n + n(n-1) - \frac{(n-1)n}{2} - (n - 1)$$

$$= \frac{n(n-1)}{2} + 1.$$

The rule given above cannot be a complete policy since

2. no draws from a 3rd urn will ever distinguish between two tied urns. Since the ties will eventually have to be broken to result in a terminal position, and since no draw except from one of the tied urns will ever break a tie, then the rule of the preceding paragraph is merely to break all ties as soon as possible.

This rule gives at least the $\binom{n}{2} + 1$ moves since, for every j, if we have drawn j balls from an urn there are at most $j + 1$ possible distinct patterns of black and white balls that can be observed. Drawing 1 ball from each of n urns takes n draws and at most two compositions can result (black ball seen or white ball seen). Drawing one more ball from all but one of the urns from which a black ball has been seen, and one more from all but one of the urns from which a white ball has been seen requires at least $n - 2$ draws (it is possible that all of the balls seen are the same color). Now there are at most 3 groups of ties, and breaking them takes at least $n-2-3$ draws. Continuing, we see that the total number of draws until all ties are broken is at least

$$n + \sum_{j=1}^{n-1} (n-j-1) = n + n(n-1) - \frac{n(n-1)}{2} - (n - 1)$$
$$= \frac{n(n-1)}{2} + 1.$$

The rule given above cannot be a complete policy since

$0_n \geq \frac{n(n+1)}{2}$ by theorem IV.1, and if $n \geq 2$

$$\frac{n(n+1)}{2} > \frac{n(n-1)}{2} + 1.$$

With this rule above one can easily verify that an optimal policy for $n = 2$ is to draw from urn 0 and urn 1. Then, draw from urn 0 and if the resulting M is not terminal, draw from 1 again. This strategy requires 3½ draws on the average and is the smallest possible.

For $n = 3$, this rule says to draw from urns 0, 1, and 2, then break all ties, reduce all matrices to the corresponding irreducible matrix and then reorder in accordance with convention IV.1. This will yield the following irreducible information matrices:

$$M_1 = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \quad M_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \end{bmatrix}$$

$$M_4 = \begin{bmatrix} 0 & 2 & 1 & 1 \\ 3 & 0 & 1 & 0 \end{bmatrix}, \quad M_5 = \begin{bmatrix} 3 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 \end{bmatrix}, \text{ and } M_6 = \begin{bmatrix} 3 & 2 & 1 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix}.$$

Referring to Appendix 2 the optimal policy for continuation from each of these can be found, and it will be seen that the minimum expected number of draws to determine the composition of all of the urns is $7 \frac{19}{36}$.

For $n = 4$, the computer printout of the dynamic program is available, on loan, from the Statistical Laboratory, Michigan State University, East Lansing, Michigan. There the minimum expected number of draws to determine the composition of all of the urns is found to be 13.136.

Table 1

Info. Set	$P_0(B\|M)$	$P_1(B\|M)$	$P_2(B\|M)$	$Ef_N(D_0(M))$	$Ef_N(D_1(M))$	$Ef_N(D_2(M))$	$MinEf_N(D_j(M))$
$\begin{pmatrix}0&0&0\\0&0&0\end{pmatrix}$	1/2	1/2	1/2	3 1/2	3 1/2	3 1/2	3 1/2
$\begin{pmatrix}1&0&0\\0&0&0\end{pmatrix}$	2/3	1/3	1/3	2 1/2	2 1/2	2 1/2	2 1/2
$\begin{pmatrix}0&0&0\\1&0&0\end{pmatrix}$	1/3	2/3	2/3	2 1/2	2 1/2	2 1/2	2 1/2
$\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}$	3/4	1/4	1/2	1 3/4	1 3/4	2	1 3/4
$\begin{pmatrix}2&0&0\\0&0&0\end{pmatrix}$	0	1/4	1/4	2 3/4	1 3/4	1 3/4	1 3/4
$\begin{pmatrix}1&0&0\\1&0&0\end{pmatrix}$	0	1/2	1/2	2	1	1	1
$\begin{pmatrix}0&0&0\\2&0&0\end{pmatrix}$	0	3/4	3/4	2 3/4	1 3/4	1 3/4	1 3/4
$\begin{pmatrix}2&0&0\\0&1&0\end{pmatrix}$	0	1/3	1/3	2	1	1 2/3	1
$\begin{pmatrix}0&1&0\\2&0&0\end{pmatrix}$	0	2/3	2/3	2	1	1 2/3	1
$\begin{pmatrix}2&0&0\\0&1&1\end{pmatrix}$	0	1/2	1/2	2	1	1	1
$\begin{pmatrix}0&1&1\\2&0&0\end{pmatrix}$	0	1/2	1/2	2	1	1	1
$\begin{pmatrix}2&1&0\\0&1&2\end{pmatrix}$	0	0	0	0	0	0	0

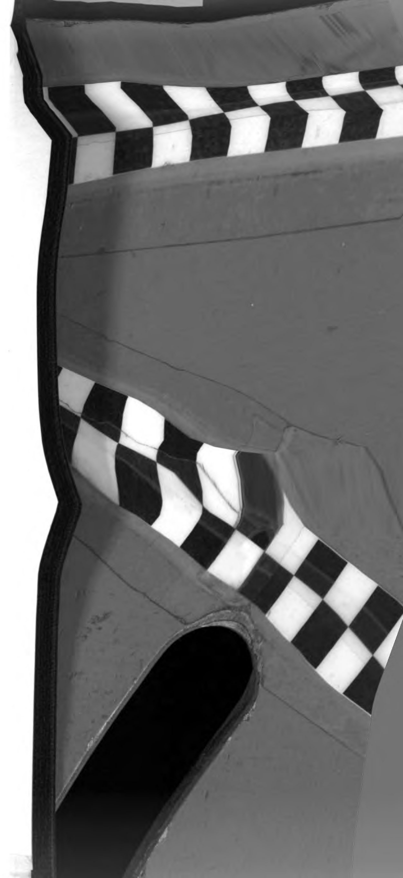
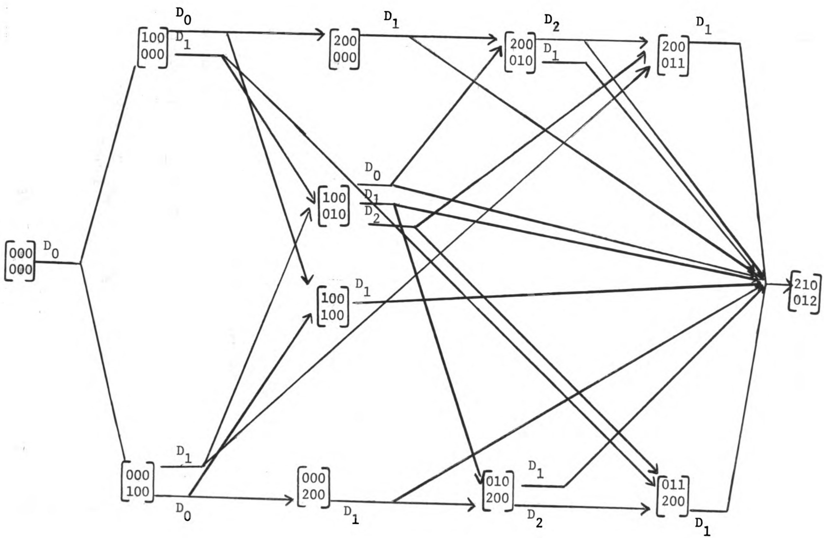


Table 2


000
000
D0
100
000
D0
D1
000
100
D1
D0
200
000
D1
100
010
D0
D1
D2
100
100
D1
000
200
D1
200
010
D2
D1
010
200
D1
D2
200
011
D1
011
200
D1
210
012

V. THE URN PROBLEM AS AN INFORMATION THEORY PROBLEM

A different approach to solving the problem in the first chapter is provided by treating it as a problem in information theory. The essence of this different approach was expressed by D.V. Lindley [4] as follows: "... although indisputably one purpose of experimentation is to reach decisions, another purpose is to gain knowledge about the state of nature (that is, about the parameter) without having specific actions in mind. This knowledge is measured by the amount of information ..."

The following decision procedure presents itself: choose at the first step to perform that experiment whose expected information gain is the greatest, and from the resulting state, perform next the experiment whose expected information gain is the greatest, continuing in this way until preassigned amount of information is achieved. This strategy will be called the "maximum information strategy".

As our measure of information we shall use the Shannon information.

<u>Definition V.1</u> For $M \in \mathcal{M}^*$, let

$$I(M) = \sum_{\pi \in H(M)} p_M(\pi) \log p_M(\pi)$$

V. THE URN PROBLEM [illegible]

A different app[illegible]

first chapter is [illegible]

information theo[illegible]

was expressed by [illegible]

indisputably one [illegible]

decisions, another [illegible]

state of nature (that is [illegible] about

having specific actions [illegible] This [illegible] measured

by the amount of informa[illegible]

The following [illegible]

choose at the first step [illegible]

expected information gain is the greatest [illegible]

resulting state, perform next the [illegible]

information gain is the greatest, continuing in this way

until preassigned amount of information is achieved. This

strategy will be called the "maximum information strategy".

As our measure of information we shall use the

Shannon information.

Definition V.1 For $M \in \mathfrak{M}^*$, let

$$I(M) = -\sum_{\pi \in \Pi(M)} p_M(\pi) \log p_M(\pi)$$

The reasons for using Shannon information as a measure of information are well known; see [3], [4].

The following definition and the theorem are due to Lindley [4] and are stated here in the notation we have developed for the urn problem of order n.

<u>Definition V.2</u> (Lindley [4], Defn. 1) The expected information provided by an experiment $D(M,j)$ with prior knowledge M is

$$I(D(M,j),M) = I(M_j^+)P_j(B|M) + I(M_j^-)(1-P_j(B|M)) - I(M)$$

<u>Theorem V.1</u> (Lindley [4], Thm. 1) $I(D(M,j),M) \geq 0$ for all $j \in I_0^n$ and $M \in \mathcal{M}^*$.

The following theorem is a characterization of a terminal set in terms of information.

<u>Theorem V.2</u> M is a terminal set if and only if $I(M) = 0$.

<u>Proof</u>: If $I(M) = 0$, then

$$\sum_{\pi \in S_{n+1}} P_M(\pi) \log P_M(\pi) = 0$$

which implies $P_M(\pi) = 0$ or 1 for all $\pi \in S_{n+1}$ or that there exists $\pi_0 \in S_{n+1}$ such that

$$P_M(\pi_0) = 1 \text{ and } P_M(\pi) = 0 \text{ if } \pi \neq \pi_0$$

since

$$\sum_{\pi \in S_{n+1}} P_M(\pi) = 1.$$

$\therefore$ H(M) is a singleton.

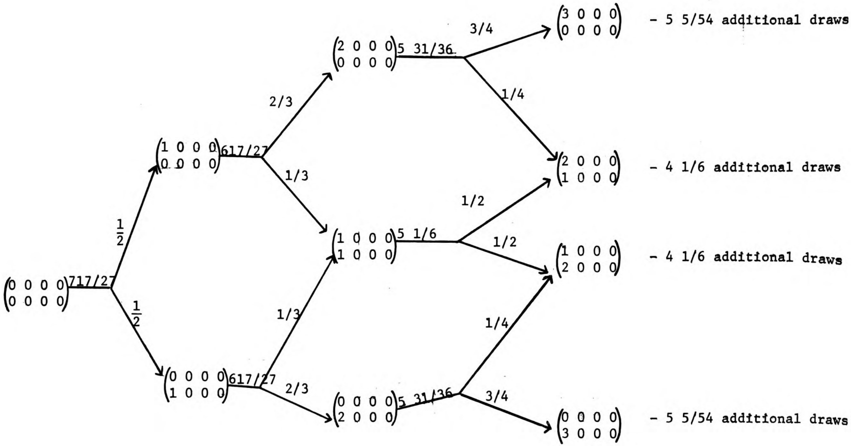
If M is terminal then there exists π_0 such that $H(M) = \{\pi_0\}$ and $P_M(\pi_0) = 1$ and $P_M(\pi) = 0$, for $\pi \neq \pi_0$ therefore

$$\sum_{\pi \in S_{n+1}} P_M(\pi) \log P_M(\pi) = P_M(\pi_0) \log P_M(\pi_0) = 0$$

Heuristically, information provides a criterion function which at the very least goes in the correct direction, in that, greater information is "closer" to a terminal position and further sampling will lead on the average to an increase in information. Thus, the procedure which at each stage chooses that experiment which has greatest expected information gain is not an obviously bad strategy. For n = 2, it is one of the optimal strategies.

An example from the urn model of size 3 will show that the maximum information strategy is not a good strategy. Its expected number of draws until a terminal position is strictly greater than an optimal procedure. The tree diagram for the first 3 draws of the maximum information strategy is shown in figure 3. The succeeding draws for the maximum information strategy coincide with an optimal strategy.

Table 3


717/27
1/2
1/2
617/27
617/27
2/3
1/3
1/3
2/3
5 31/36
5 1/6
5 31/36
3/4
1/4
1/2
1/2
1/4
3/4
- 5 5/54 additional draws
- 4 1/6 additional draws
- 4 1/6 additional draws
- 5 5/54 additional draws

The first difference between the maximum information strategy and an optimal strategy occurs at $M = \binom{2\ 0\ 0\ 0}{0\ 0\ 0\ 0}$. The maximum information strategy is to draw from the 0^{th} urn, while the optimal strategy is to draw from any other urn. The maximum information strategy has expected number of draws until the composition of all of the urns is determined equal to $7\frac{17}{27}$, whereas the smallest expected number of draws until the composition of all of the urns is determined as $7\frac{19}{36}$.

The question of the optimality of the maximum information strategy for an urn problem of order n for $n > 3$ is unanswered. Also unanswered are questions of optimality of the maximum information strategy for measures of information other than the Shannon information and for criterion functions other than the expected number of draws to determine the composition of all of the urns.

The first difference between [illegible] information strategy and an optimal [illegible] The maximum information [illegible] from the urn, while the [illegible] other urn. The maximum [illegible] expected number of draws until [illegible] termined equal to [illegible] expected number of draws until the [illegible] determined as $7 \frac{19}{36}$.

The question of the optimality of the maximum information strategy for [illegible] $n > 3$ is unanswered. Also unanswered are questions of optimality of the maximal information strategy for measures of information other than the Shannon information and for criterion functions other than the expected number of draws to determine the composition of all of the urns.

BIBLIOGRAPHY

[1] Bellman, R. *Dynamic Programming*, Princeton University Press, Princeton, New Jersey.

[2] Blackwell, D. and M.A. Girshick *Theory of Games and Statistical Decisions*, Wiley, New York.

[3] Khinchin, A.I. *Mathematical Foundations of Information Theory*, Dover, New York.

[4] Lindley, D.V. "On a Measure of the Information Provided by an Experiment", *Annals of Math. Statist.*, 27: 986-1005.

[5] Shen, Mak-Kong "Generation of Permutations in Lexicographical Order", *Comm. ACM* (1963), p. 517.

APPENDIX 1

FORTRAN IV program for the dynamic program to solve the urn problem of order n.

DICTIONARY

N	=	the number of urns in an urn problem of order n. $N = n + 1$
NFACT	=	$(n+1)!$
Z(I)	=	the information matrix M_I in packed form
P(I,J)	=	the J^{th} coordinate of the I^{th} permutation on I_0^n in lexicographical order
PROB(I,J)	=	$P_j(B\|M_I)$, the conditional probability of drawing a black ball from the j^{th} urn given the past observations M_I
PP(I)	=	the posterior probability of the I^{th} permutation of I_0^n
G(I)	=	the smallest expected number of draws to termination from information matrix M_I


```
      PROGRAM URNS
      TYPE INTEGER P, S, W,Z, U
      DIMENSION P(120,5),PP(120),PROB(3000,5),Z(3000),S(10),G(3000)
     1,H(5),M(5,5),ITOT(5)
      N=5
      NFACT=120
      DO 1 L=1,N
      P(1,L)=L-1
 1    S(L)=L
      JJ=2
 2    W=N
      IF(S(W).GE.S(W-1)) GO TO 4
      IF(2.EQ.W) GO TO 25
      W=W-1
      GO TO 3
 4    U=S(W-1)
      INDEX1 =N-W+1
      DO 5 II=1,INDEX1
      L =N+1-II
      IF (S(L).LE.U) GO TO 5
      S(W-1)=S(L)
      S(L)=U
      GO TO 6
 5    CONTINUE
 6    INDEX2 = (N-W+1)/2+0.1
      DO 7 II=1,INDEX2
      L=II-1
      U=S(N-L)
      S(N-L) = S(W+L)
      S(W+L) = U
```


```
7       CONTINUE
        DO 8 K=1,N
8       P(JJ,K)= S(K) -1
        JJ=JJ+1
        GO TO 2
25      CONTINUE
        IF(JJ.NE.(NFACT +1)) GO TO 2200
        KK=1
        Z(KK) = 0
        DO 99 I=1,N
99      PROB(KK,I) =.5
        GO TO 123
123     INDEX3 = 3
        KK=KK+1
        DO 124 I=1,10
        I10=10**(I+1)
124     S(I)=(Z(KK-1)-(Z(KK-1)/I10)*I10)/(I10/10)
1241    KKK=0
        DO 1242 I=1,N
        ITOT (I) = S(I) + S(I+N)
        IF (ITOT(I) .GT. 0) KKK=KKK+1
1242    CONTINUE
        IF(KKK.GE.1) GO TO 128
        S(1) = 1
        ITOT(1) = 1
        GO TO 250
128     DO 150 II= 1,KKK
        I = KKK-II+1
        IF(S(I) .EQ. 0) GO TO 140
        S(I) = S(I) -1
```

```
      S(I+N) = ITOT(I) - S(I)
       IF(I.EQ. KKK) GO TO 2100
       DO 131 JJ= I,KKK
       J= JJ+1
       IF(ITOT(J) .EQ. ITOT(J-1)) GO TO 129
       S(J) = ITOT(J)
       S(J+N) = ITOT(J) - S(J)
       GO TO 130
  129  S(J) = S(J-1)
       S(J+N) = S(J+N-1)
  130  IF(J.EQ.KKK) GO TO 2100
  131  CONTINUE
  149  IF(I.EQ.1) GO TO 151
  150  CONTINUE
  151  NNN= N-1
       DO 152 I=1,NNN
       IF( ITOT(N+1-I) .GE. ITOT(N- I)) GO TO 152
       ITOT(N+1-I) = ITOT(N+1 -I) +1
       LLL=N+2-I
       IF(LLL.GT.N) GO TO 155
       DO 1511 J=LLL,N
 1511    ITOT(J) =0
      GO TO 155
  152 CONTINUE
      IF( ITOT(1) .GE. (N-1)) GO TO 280
      ITOT(1)= ITOT(1) + 1
      DO 154 J=2,N
  154 ITOT(J) =0
  155 DO 156 I=1,N
       S(I) = ITOT(I)
```


```
156     S(I+N) =0
        IF(ITOT(N-1) .EQ. 0  GO TO 2100
        S(N-1) = 0
        S(N) =0
        S(2*N-1) = ITOT(N-1)
        S(2*N) = ITOT(N)
        GO TO 2100
200     IF (KKKK.EQ. 0 ) GO TO 1241
        DO 201 J= 1,N
        DO 201 I= 1,N
        IF (M(I,J).NE. KKKK) GO TO 201
        IF((S(I).NE.(J-1)).OR. (S(I+N).NE.(N-J))) GO TO 1241
201     CONTINUE
250     INDEX1 = NFACT/N *S(1) +1
        INDEX2 = NFACT -(N-1)*S(N+1)
251     DO 252 L = INDEX1, INDEX2
252     PP(L) =1.
        DEN =0.
        DO 254 L = INDEX1, INDEX2
        DO 253 I=1,N
        II= I+N
253     PP(L) = PP(L)*BIC(P(L,I),S(I))* BIC(N-1-P(L,I), S(II))
254     DEN = DEN + PP(L)
        IF (DEN) 1241, 1241, 255
255     DO 256 L = INDEX1, INDEX2
256     PP(L) = PP(L)/ DEN
        DO 260 I=1,N
260     PROP(KK,I)=0.
        DO 262 I=1,N
        IF( I.EQ. 1) GO TO 261
```

```
      IF((S(I).NE. S(I-1)).OR. (S(I+N) .NE. S(I+N-1))) GO TO 261
      PROB(KK,I) = PROB(KK,I-1)
261   IF((S(I) + S(I+N)).EQ. (N-1)) GO TO 262
      DO 262 L= INDEX1, INDEX2
      GO TO 262
      PROB(KK,I) = PROB(KK,I) + PP(L)*(P(L,I)-S(I))/(N-1-S(I)-S(I+N))
262   CONTINUE
      Z(KK)=0
      DO 263 I=1,10
263   Z(KK)=S(I)*10**I +Z(KK)
      DO 264 I=1,10
      I10=10**(I+1)
264   S(I)=(Z(KK)-(Z(KK)/I10)*I10)/(I10/10)
      WRITE(61,2011) KK,Z(KK),(S(I),I=1,10),(PROB(KK,I),I=1,5)
2011  FORMAT(I5,I15,10I5,5F10.3)
      GO TO 123
30    Z(KK)=0
      DO 281 I=1,N
      Z(KK)=Z(KK) +(N-I)*10**I+(I-1)*10**(N+I)
      S(I)=N-I
      S(I+N)= I-1
281   PROB (KK,I) = 0.
      GO TO 2000
C     DYNAMIC PROGRAM
2000  I TOTAL= KK
      G(KK) = 0.0
      WRITE(61,3100) KK,(S(I),I=1,5),G(KK),(S(I),I=6,10)
1000  K= KK-1
      DO 1008 I=1,5
1008  H(I) = 100.
```


```
      DO 2009 J=1,N
      DO 1997 I=1,10
      I10=10**(I+1)
 1997 S(I)=(Z(KK)-(Z(KK)/I10)*I10)/(I10/10)
      IF (S(J) +S(J+N)-N+1) 2002,2009,2200
 2002 S(J) =S(J) + 1
      INDEX3=1
      GO TO 2100
 2004 DO2005 I=1,10
      I10=10**(I+1)
 2005 S(I)=(Z(KK)-(Z(KK)/I10)*I10)/(I10/10)
      S(J+N) = S(J+N) +1
      INDEX3 =2
 2100 INDEX1 = NFACT/N *S(J) +1
      KKKK= 0
      INDEX2 = NFACT -(N-1)*S(N+1)
      DO 2101 I=1,N
      DO 2101 JJ =1,N
 2101 M(I,JJ) =0
      DO 2104 I = INDEX1, INDEX2
      DO 2102 II =1,N
      IF((S(II).GT. P(I,II)).OR. (S(II+N) .GT.(N-1-P(I,II)))) GO TO
     2104
 2102 CONTINUE
      KKKK= KKKK+1
      DO 2103 II=1,N
      LL = P(I,II)+1
 2103 M(II,LL) = M(II,LL) +1
 2104 CONTINUE
```

```
      DO 2105 II=1, N
      DO 2105 JJ=1, N
      IF(M(II,JJ) .NE. KKKK) GO TO 2105
      S(II) =JJ-1
      S(II+N) = N-1-S(II)
 2105 CONTINUE
C
C     REORDER THE INFORMATION MATRIX
      DO 2106 JJ =1, N
      DO 2106 I=JJ,N
      IF(((S(JJ)+S(JJ+N)).GT. (S(I) +S(I+N))).OR.(((S(JJ)+S(JJ+N)).EQ
     1.(S(I)+S(I+N)).AND.(S(JJ).GE.S(I)))) GO TO 2106
      I1 = S(JJ)
      I2 = S(JJ+N)
      S(JJ)= S(I)
      S(JJ+N) =S(I+N)
      S(I) =I1
      S(I+N) =I2
 2106 CONTINUE
      IZ=0
      DO 2107 I=1,10
 2107 IZ=IZ+S(I)*10**I
      DO 2108 I = KK, ITOTAL
      IF(IZ.NE.Z(I )) GO TO 2108
      PP(INDEX3) = G(I) +1
      IF(INDEX3.EQ.1 ) GO TO 2004
      GO TO 2008
 2108 CONTINUE
 2008 P(J)= PROB(KK,J) *PP(1) +(1-PROB(KK,J))*PP(2)
 2009 CONTINUE
```

```
      G(KK) = AMIN1(H(1), H(2), H(3), H(4), H(5))
      IF(G(KK).GE.100.0)  G(KK)=0.0
      DO 2500 I=1,10
      I10=10**(I+1)
 2500 S(I)=(Z(KK)-(Z(KK)/I10)*I10)/(I10/10)
      WRITE(61,3003) KK,(S(I),I=1,5),G(KK), H(1),H(2),H(3),
     1 H(4),H(5),(S(I),I=6,10)
 3100 FORMAT(*1START OF DYNAMIC PROGRAM*/*0INFORMATION MATRIX*,6I4,
     1F10.3/ 23X ,5I4)
 3003 FORMAT(*0 INFORMATION MATRIX*,6I4,6F10.3/24X,5I4)
      IF (KK.GT.1)  GO TO 1999
      GO TO 3006
 2200 WRITE(61,3004 )
 3004 FORMAT(*DEN LESS OR EQUAT ZERO*)
 3006 CONTINUE
      END
      FUNCTION BIC(M,N)
      IF( M.GE. N) GO TO 70
      BIC =  0.0
      RETURN
 70   GO TO (71,71,73,74,75,76) M+1
 73   GO TO (71, 172, 71) N+1
 74   GO TO (71, 173, 173, 71) N+1
 75   GO TO (71,174, 176, 174, 71) N+1
 76   GO TO (71, 175, 180, 180, 175, 71) N+1
 71   BIC =  1.
      RETURN
 172  BIC =  2.
      RETURN
 173  BIC =  3.
```

```
      RETURN
174   BIC = 4.
      RETURN
175   BIC = 5.0
      RETURN
176   BIC = 6.0
      RETURN
180   BIC = 10.
      RETURN
      END
'RUN,30,,15000
```

APPENDIX 2

The dynamic program for the urn problem of order 3.

Information Matrix	$P_0(B\ M)$	$P_1(B\ M)$	$P_2(B\ M)$	$P_3(B\ M)$	$E(f_N(D_0(M)))$	$E(f_N(D_1(M)))$	$E(f_N(D_2(M)))$	$E(f_N(D_3(M)))$
0 0 0 0 0 0 0 0	1/2	1/2	1/2	1/2	7 19/36	7 19/36	7 19/36	7 19/36
1 0 0 0 0 0 0 0	2/3	11/27	11/27	11/27	6 19/36	6 19/36	6 19/36	6 19/36
0 0 0 0 1 0 0 0	1/3	16/27	16/27	16/27	6 19/36	6 19/36	6 19/36	6 19/36
1 1 0 0 0 0 0 0	13/22	13/22	3/11	3/11	5 15/44	5 15/44	5 15/44	5 15/44
1 0 0 0 0 1 0 0	23/32	9/32	1/2	1/2	5 21/32	5 21/32	5 21/32	5 21/32
0 0 0 0 1 1 0 0	9/22	9/22	8/11	8/11	5 15/44	5 15/44	5 15/44	5 15/44
1 1 0 0 0 0 1 0	5/8	5/8	3/16	5/16	4 21/32	4 21/32	4 21/32	4 303/320
1 0 0 0 0 1 1 0	13/16	3/8	3/8	11/16	4 21/32	4 21/32	4 21/32	4 303/320
1 1 0 0 0 0 1 1	7/10	7/10	3/10	3/10	4 3/20	4 3/20	4 3/20	4 3/20
2 0 0 0 0 0 0 0	3/4	13/36	13/36	13/36	5 31/36	5 17/24	5 17/24	5 17/24
1 0 0 0 1 0 0 0	1/2	1/2	1/2	1/2	5 1/6	5 1/6	5 1/6	5 1/6
0 0 0 0 2 0 0 0	1/4	23/36	23/36	23/36	5 31/36	5 17/24	5 17/24	5 17/24

2 1 0 0 0 0 0 0	9/13	6/13	3/13	3/13	4 15/26	4 1/2	4 6/13	4 6/13
2 0 0 0 0 1 0 0	18/23	7/23	10/23	10/23	5 1/46	4 39/46	4 39/46	4 39/46
1 1 0 0 1 0 0 0	4/9	7/9	1/3	1/3	4 1/6	4 1/6	4 1/6	4 1/6
1 0 0 0 1 1 0 0	5/9	2/9	2/3	2/3	4 1/6	4 1/6	4 1/6	4 1/6
0 1 0 0 2 0 0 0	5/23	16/23	13/23	13/23	5 1/46	4 39/46	4 39/46	4 39/46
0 0 0 0 2 1 0 0	4/13	7/13	10/13	10/13	4 15/26	4 1/2	4 6/13	4 6/13
2 1 0 0 0 0 1 0	7/10	1/2	2/10	3/10	3 9/10	3 4/5	3 3/4	4 3/50
2 0 0 0 0 1 1 0	11/13	5/13	5/13	7/13	4 3/26	3 12/13	3 12/13	4 6/13
1 1 0 0 1 0 1 0	1/2	5/6	1/6	1/2	3 1/2	3 1/2	3 1/2	3 1/2
0 1 1 0 2 0 0 0	3/13	8/13	8/13	6/13	4 3/26	3 12/13	3 12/13	4 6/13
0 1 0 0 2 0 1 0	3/10	4/5	1/2	7/10	3 17/20	3 3/4	3 4/5	4 3/50
2 1 0 0 0 0 1 1	5/7	4/7	2/7	2/7	3 3/7	3 17/28	3 3/7	3 3/7
0 1 1 0 2 0 0 1	2/7	5/7	5/7	4/7	3 3/7	3 3/7	3 3/7	3 17/28

2 2 0 0 0 0 0 0	1/2	1/2	1/6	1/6	3 1/2	3 1/2	3 2/3	3 2/3
2 1 0 0 0 1 0 0	6/7	3/7	2/7	2/7	3 11/14	3 1/2	3 1/2	3 1/2
2 0 0 0 0 2 0 0	3/4	1/4	1/2	1/2	4 3/16	4 3/16	4	4
1 1 0 0 1 1 0 0	1/2	1/2	1/2	1/2	2	2	2	2
1 0 0 0 1 2 0 0	4/7	1/7	5/7	5/7	3 1/2	3 11/14	3 1/2	3 1/2
0 0 0 0 2 2 0 0	1/2	1/2	5/6	5/6	3 1/2	3 1/2	3 1/2	3 1/2
2 2 0 0 0 0 1 0	1/2	1/2	1/5	1/5	2 4/5	2 4/5	2 4/5	3 2/5
2 1 0 0 0 1 1 0	9/10	1/2	1/5	2/5	3 1/5	2 4/5	2 4/5	3 1/5
2 0 1 0 0 2 0 0	11/16	3/16	1/2	3/8	3 3/16	3 3/16	3	3 5/8
2 0 0 0 0 2 1 0	13/16	5/16	1/2	5/8	3 3/16	3 3/16	3	3 5/8
1 0 1 0 1 2 0 0	1/2	1/10	4/5	4/5	2 4/5	3 1/5	2 4/5	3 1/5
0 0 1 0 2 2 0 0	1/2	1/2	4/5	4/5	2 4/5	2 4/5	3 3/5	3 2/5
2 2 0 0 0 0 1 1	1/2	1/2	1/4	1/4	2 3/4	2 3/4	2 3/4	2 3/4

2 0 1 0 0 2 0 1	7/10	3/10	3/5	2/5	2 7/10	2 7/10	2 9/10	2 9/10
0 0 1 1 2 2 0 0	1/2	1/2	3/4	3/4	2 3/4	2 3/4	2 3/4	2 3/4
2 2 0 0 0 0 2 0	1/2	1/2	1/4	1/4	2	2	2	3
2 1 0 0 0 1 2 0	7/8	1/2	1/8	1/2	2 1/2	2	2 1/2	2 3/4
2 0 0 0 0 2 2 0	3/4	1/2	1/2	3/4	2	2	2	3
2 2 0 0 0 0 2 1	1/2	1/2	1/3	1/3	2	2	2	2 2/3
2 0 0 1 0 2 2 0	2/3	1/2	1/2	2/3	2	2	2	2 2/3
2 2 0 0 0 0 2 2	1/2	1/2	1/2	1/2	2	2	2	2
3 0 0 0 0 0 0 0	0	1/3	1/3	1/3	∞	5 5/54	5 5/54	5 5/54
2 0 0 0 1 0 0 0	0	4/9	4/9	4/9	∞	4 1/6	4 1/6	4 1/6
1 0 0 0 2 0 0 0	0	5/9	5/9	5/9	∞	4 1/6	4 1/6	4 1/6
0 0 0 0 3 0 0 0	0	2/3	2/3	2/3	∞	5 5/54	5 5/54	5 5/54

3 1 0 0 0 0 0 0	0	1/3	2/9	2/9	∞	3 8/9	3 7/9	3 7/9
3 0 0 0 0 1 0 0	0	1/3	7/18	7/18	∞	4 1/4	4 1/4	4 1/4
2 1 0 0 1 0 0 0	0	3/4	1/4	1/4	∞	3 1/8	3 1/8	3 1/8
2 0 0 0 1 1 0 0	0	1/5	3/5	3/5	∞	3 1/5	3 1/5	3 1/5
1 1 0 0 2 0 0 0	0	4/5	2/5	2/5	∞	3 1/5	3 1/5	3 1/5
1 0 0 0 2 1 0 0	0	1/4	3/4	3/4	∞	3 1/8	3 3/8	3 3/8
0 1 0 0 3 0 0 0	0	2/3	11/18	11/18	∞	4 1/4	4 1/4	4 1/4
0 0 0 0 3 1 0 0	0	2/3	7/9	7/9	∞	3 8/9	3 7/9	3 7/9
3 1 0 0 0 0 1 0	0	5/14	3/14	2/7	∞	3 13/28	3 1/4	3 3/7
3 0 0 0 0 1 1 0	0	9/22	9/22	5/11	∞	3 4/11	3 4/11	4 3/22
2 1 0 0 1 0 1 0	0	5/6	1/6	1/3	∞	2 1/2	2 1/2	2 1/2
1 1 0 0 2 0 1 0	0	5/6	1/6	2/3	∞	2 1/2	2 1/2	2 1/2

0 1 1 0 3 0 0 0	0	13/22	13/22	6/11	∞	3 4/11	3 4/11	4 3/22
0 1 0 0 3 0 1 0	0	11/14	9/14	5/7	∞	3 1/14	3 3/14	3 3/7
3 1 0 0 0 0 1 1	0	2/5	3/10	3/10	∞	3 1/5	2 7/10	2 7/10
3 0 0 0 0 1 1 1	0	1/2	1/2	1/2	∞	3 1/2	3 1/2	3 1/2
0 1 1 1 3 0 0 0	0	1/2	1/2	1/2	∞	3 1/2	3 1/2	3 1/2
0 1 1 0 3 0 0 1	0	7/10	7/10	6/10	∞	2 7/10	2 7/10	3 1/5
3 1 0 0 0 1 0 0	0	1/2	1/4	1/4	∞	3 1/12	3 1/12	3 1/12
3 0 0 0 0 2 0 0	0	1/4	11/24	11/24	∞	3 5/12	3 1/3	3 1/3
2 0 0 0 1 2 0 0	0	1/4	5/8	5/8	∞	2 1/2	2 1/2	2 1/2
1 2 0 0 2 0 0 0	0	3/4	3/8	3/8	∞	2 1/2	2 1/2	2 1/2
0 2 0 0 3 0 0 0	0	3/4	13/24	13/24	∞	3 5/12	3 1/3	3 1/3
0 1 0 0 3 1 0 0	0	1/2	3/4	3/4	∞	3 1/12	3 1/12	3 1/12

3 1 0 0 0 1 1 0	0	5/9	2/9	1/3	∞	2 4/9	2 4/9	3
3 0 1 0 0 2 0 0	0	2/11	4/11	4/11	∞	2 4/11	2 4/11	3 6/11
3 0 0 0 0 2 1 0	0	4/13	7/13	7/13	∞	2 9/13	2 4/13	3 3/13
2 0 1 0 1 2 0 0	0	1/5	4/5	2/5	∞	1 4/5	1 4/5	2
1 2 0 0 2 0 1 0	0	4/5	1/5	3/5	∞	1 4/5	1 4/5	2 3/10
0 2 1 0 2 0 1 0	0	9/13	6/13	6/13	∞	2 6/13	2 4/13	3 3/13
0 2 0 0 3 0 1 0	0	9/11	7/11	7/11	∞	2 4/11	2 4/11	2 10/11
0 1 1 0 3 1 0 0	0	4/9	7/9	2/3	∞	2 4/9	2 4/9	3
3 1 0 0 0 1 1 1	0	2/3	1/3	1/3	∞	2 1/2	2 1/2	2 1/2
3 0 1 0 0 2 0 1	0	2/7	3/7	3/7	∞	2 1/7	2 3/7	2
3 0 0 0 0 2 1 1	0	1/3	2/3	2/3	∞	2 1/2	2 1/2	2 1/2
0 2 1 1 3 0 0 0	0	2/3	1/3	1/3	∞	2 1/2	2 1/2	2 1/2

0 2 1 0 3 0 0 1	0	5/7	4/7	4/7	∞	2 1/7	2	2 3/7
0 1 1 1 3 1 0 0	0	1/3	2/3	2/3	∞	2 1/2	2 1/2	2 1/2
3 1 0 0 0 1 2 0	0	4/7	1/7	3/7	∞	1 4/7	1 6/7	2 3/7
0 2 1 0 3 0 1 0	0	6/7	3/7	4/7	∞	1 6/7	1 4/7	2 3/7
3 1 0 0 0 1 2 1	0	3/4	1/4	1/2	∞	1 3/4	1 3/4	2
0 2 1 1 3 0 1 0	0	3/4	1/4	1/2	∞	1 3/4	1 3/4	2
3 2 0 0 0 1 0 0	0	0	1/6	1/6	∞	∞	2 1/2	2 1/2
3 1 0 0 0 2 0 0	0	0	1/3	1/3	∞	∞	1 2/3	1 2/3
3 0 0 0 0 2 0 0	0	0	1/2	1/2	∞	∞	2 2/3	2 2/3
2 1 0 0 1 2 0 0	0	0	1/2	1/2	∞	∞	1	1
2 0 0 0 1 3 0 0	0	0	2/3	2/3	∞	∞	1 2/3	1 2/3
1 0 0 0 2 3 0 0	0	0	5/6	5/6	∞	∞	2 1/2	2 1/2

3 2 0 0 0 1 1 0	0	0	1/5	1/5	∞	∞	1 4/5	2 2/5
3 1 0 0 0 2 1 0	0	0	1/4	1/2	∞	∞	1	1 3/4
3 0 1 0 0 3 0 0	0	0	1/3	4/9	∞	∞	1 2/3	2 2/3
3 0 0 0 0 3 1 0	0	0	2/3	5/9	∞	∞	1 2/3	2 2/3
2 0 1 0 1 3 0 0	0	0	3/4	1/4	∞	∞	1	1 3/4
1 0 1 0 2 3 0 0	0	0	4/5	4/5	∞	∞	1 4/5	2 2/5
3 2 0 0 0 1 1 1	0	0	1/4	1/4	∞	∞	1 3/4	1 3/4
3 1 0 0 0 2 1 1	0	0	1/2	1/2	∞	∞	1	1
3 0 1 1 0 3 0 0	0	0	1/4	1/4	∞	∞	1 3/4	1 3/4
3 0 1 0 0 3 0 1	0	0	2/5	3/5	∞	∞	1 3/5	1 3/5
3 0 0 0 0 3 1 1	0	0	3/4	3/4	∞	∞	1 3/4	1 3/4
2 0 1 1 1 3 0 0	0	0	1/2	1/2	∞	∞	1	1

1 0 1 1 2 3 0 0	0	0	3/4	3/4	∞	∞	1 3/4	1 3/4
3 2 0 0 0 1 2 0	0	0	1/4	1/4	∞	∞	1	1 3/4
3 0 1 0 0 3 1 0	0	0	1/2	1/2	∞	∞	1	1 1/2
1 0 2 0 2 3 0 0	0	0	3/4	3/4	∞	∞	1	1 3/4
3 2 0 0 0 1 2 1	0	0	1/3	1/3	∞	∞	1	1 2/3
3 0 1 1 0 3 1 0	0	0	1/3	1/3	∞	∞	1	1 2/3
3 0 1 0 0 3 1 1	0	0	2/3	2/3	∞	∞	1	1 2/3
1 0 2 1 2 3 0 0	0	0	2/3	2/3	∞	∞	1	1 2/3
3 2 0 0 0 1 2 2	0	0	1/2	1/2	∞	∞	1	1
3 0 1 1 0 3 1 1	0	0	1/2	1/2	∞	∞	1	1
1 0 2 2 2 3 0 0	0	0	1/2	1/2	∞	∞	1	1
3 2 1 0 0 1 2 3	0	0	0	0	0	0	0	0