53

STATISTICAL PROPERTIES OF THE BUDDY SYSTEM

by

Paul W. Purdom Jr. and Stephen M. Stigler

Technical Report #60

April 1969

Computer Sciences Department
1210 West Dayton Street
Madison, Wisconsin 53706

STATISTICAL PROPERTIES OF THE BUDDY SYSTEM*

ABSTRACT

The utilization of space and the running speed of the buddy system are considered. Equations are derived that give various statistical properties of the buddy system. The bottom level with Poisson requests and exponential service times is investigated in great detail. Some equations are also given for investigating all levels at the same time and for non-Poisson requests. For the bottom level with Poisson requests and exponential service times the expected amount of space wasted by pairing full cells with empty cells and the amount of time spent by the bottom level requesting space from the next level is calculated for many rates of request. Asymptotic formulas are found which give a good fit to the calculated values. The results of a number of simulations of the buddy system are also given. These simulations indicate the behavior of many characteristics of the buddy system which can not be calculated in reasonable time with the formulas presented.

*This research was supported in part by The National Science Foundation, Grant number GP7069, The Wisconsin Alumni Research Foundation, and The Office of Naval Research under contract number Nonr 1202(17), Project number NR 042-222.

1. INTRODUCTION

The buddy system is an example of a dynamic storage allocation algorithm. It does the bookkeeping required for making available various size blocks of memory to other routines in a computer [1]. The buddy system provides only blocks of memory whose size is a power of two (times some basic size). It can, however, provide a block of storage very quickly. Since there are many algorithms which provide dynamic storage allocation, it is useful to analyze each algorithm to determine how quickly it can allocate memory and how much space it wastes. Knuth gives a description of several well-known methods of storage allocation and a brief comparison of the methods [2]. He also points out the need for additional analyses of the methods.

For most users of the buddy system the fact that memory is provided only in some multiple of the powers of two is a major cause of inefficiency. If the user's needs are not the same as the sizes provided, the system provides him with more space than is necessary, since the system must provide blocks of memory that are at least large enough to meet his needs. If the distribution of the size of future requests is known, the effect of this inefficiency is easy to calculate, and we will say nothing more about it.

In this paper we will concentrate on the inefficiency that remains even if the requests are always for blocks which are a power of two in size. First we consider the relation between the demand for blocks on

the bottom level (blocks of the smallest or basic size are said to be on the bottom level) and the number of cells that this demand makes unavailable on the next level (that is, all blocks of twice the basic size.) The results for the bottom level can be used on any level provided the demand for cells on the lower levels is so low that they do not have a significant effect. After the analysis of the bottom level, we show how to derive equations which describe the higher levels; however these equations take much longer to solve in the interesting cases. Once the various equations are derived we investigate the solutions of the equations, again concentrating on the bottom level. Finally the results of some simulations of the buddy system are given. These throw light on the behavior of the upper levels.

The three basic ideas of the buddy system are as follows. (1) A separate list of available blocks is kept for each size 2^k, for $0 \leq k \leq m$, where 2^m is the total amount of space. (2) When a block of size 2^k is requested, it is taken from the available space list for that size, or if no block of that size is available, the system requests a block of size 2^{k+1} which it splits into two equal parts. The resulting blocks are called buddies of each other. One half is used to fill the original request and the other is put on the available space list for size 2^k. If there is no space of size 2^{k+1} available then the request for size 2^{k+1} will of course result in additional system requests for larger sizes until either space is found or the method fails

because there is no block large enough for the original request. (3) When a block of size 2^k is returned it is combined with its buddy if its buddy is not in use. The resulting block of size 2^{k+1} is then combined with its buddy if its buddy is free. This is continued until a block is formed whose buddy is in use. The resulting block is added to the available space list for its size. The method is feasible because the buddy of any block can be found quickly: If the cells are numbered from zero to 2^n-1, a block of size 2^k will start with cell $i2^k$ for some integer i and its buddy will start at cell $(i+1)2^k$ if i is even and at cell $(i-1)2^k$ if i is odd. The block of size 2^n, of course, has no buddy, but in the algorithm we may think of it as having a buddy which is always in use. A detailed description of the algorithm for the buddy system is given by Knownton [1] and also by Knuth [2]. The algorithm is such that its running time depends only on the number of blocks that are requested by the user and on the number of additional requests that the system makes to itself because it does not have available a block of the required size.

2. ANALYSIS

2.1 Pairing on the bottom level

We will first consider a stochastic model where we keep track only of the pairing of blocks on the bottom level. In this model, which we shall call the restricted model, there are always $2n$ cells for

filling requests on the bottom level. Thus in this model we ignore the fact that, in the original system, orders on the upper levels can change the amount of space available for use by the bottom level. Orders are ignored when all of the $2n$ cells are in use. When the number of cells available is unlimited, we have what we call the unrestricted model. The two models behave in nearly the same way when $2n$ is sufficiently larger than the average number of cells needed to fill requests. We shall assume that the requests for blocks follow a Poisson process and that the lengths of time the blocks are used (service times) are given by independent random variables with an exponential distribution[†]. This assumption causes the future development of the system to depend only on the present state of the system and not on its previous history. An analysis of this model will permit one to determine when the system has enough cells to nearly always meet its requests, how much space is tied up in the available space list for the bottom level thus not being available for requests on the next level, and how often the bottom level requests space from the next level. The behavior of the model suggests how the system will behave when there are not enough cells to meet most requests. Also the model can be used for

---

[†] A stochastic process X_t is called a Poisson process with intensity λ if the probability $\text{Prob}\{X_{t+s} - X_s = K\} = e^{-\lambda t}(\lambda t)^K/K!$ for any $s,t \geq 0$, and $K=0,1,\cdots$ and also the random variables $X_{t_1} - X_{t_0}$, $X_{t_2} - X_{t_1}, \cdots, X_{t_n} - X_{t_{n-1}}$ are independent for any real numbers $0 \leqq t_0 \leqq t_1 \leqq \cdots \leqq t_n$. A random variable Y is said to have an exponential distribution with decay rate μ if $\text{Prob}\{Y > t\} = e^{-\mu t}$.

levels other than the bottom level so long as the rate of requests to levels below the one being investigated is so low that they have only a small effect on the level being studied.

The state of the model at time t is given by the bivariate stochastic process $Z_t = (E_t, F_t)$, where E_t is the number of pairs in which one block is in use and its buddy is available, and F_t is the number of pairs where both blocks (which are buddies of each other) are in use. We shall call E_t the number of empty pairs and F_t the number of full pairs. Let

$$P_{k\ell}(t) = \text{Prob}\{Z_t = (k,\ell)\} \quad \text{and}$$

$$Q^{ij}_{k\ell}(h) = \text{Prob}\{Z_{t+h} = (i,j) \mid Z_t = (k,\ell)\},$$

where $i,j,k,\ell,t,h \geq 0$. For the restricted model, Z_t is a Markov process with stationary transition probabilities satisfying the following conditions:

1) $Z_0 = (0,0)$

2) $$Q^{ij}_{k\ell}(h) = \begin{cases} \lambda h + o(h) & \text{if } i=1,\ k=0,\ j=\ell, \text{ and } j \leq n-1 \text{ or} \\ & \text{if } k=i+1,\ j=\ell+1, \text{ and } k+\ell \leq n; \\ k\mu h + o(h) & \text{if } k=i+1,\ j=\ell, \text{ and } k+\ell \leq n; \\ 2\ell\mu h + o(h) & \text{if } i=k+1,\ \ell=j+1, \text{ and } k+\ell \leq n; \\ 1-[\lambda+(k+2\ell)\mu]h + o(h) & \text{if } i=k,\ j=\ell,\ \ell \leq n-1, \text{ and} \\ & \quad k+\ell \leq n; \\ 1-2n\mu h + o(h) & \text{if } i=k=o \text{ and } j=\ell=n; \end{cases}$$

where $\lambda > 0$ is the rate of requests and $\mu > 0$ is the decay (or service) rate for a block that is in use. Then from the axioms of probability, the process obeys the following set of equations:

$$P_{k\ell}(t+h) = P_{k-1\ell}(t)\, Q^{k\ell}_{k-1\ \ell}(h) + P_{k+1\ \ell-1}(t)\, Q^{k\ \ell}_{k+1\ \ell-1}(h)$$
$$+ P_{k+1\ \ell}(t)\, Q^{k\ \ell}_{k+1\ \ell}(h) + P_{k-1\ \ell+1}(t)\, Q^{k\ \ell}_{k-1\ \ell+1}(h)$$
$$+ P_{k\ \ell}(t)\, Q^{k\ \ell}_{k\ \ell}(h) + o(h).$$

Taking the limit as h goes to zero and replacing the Q's by their values we get the following differential equations for the system:

$$P'_{k\ \ell}(t) = -(k+2\ell)\mu\, P_{k\ell}(t) - [\lambda P_{k\ell}(t)]_{\text{if } \ell \neq n}$$
$$+ (k+1)\mu P_{k+1\ \ell}(t) + [2(\ell+1)\mu P_{k-1\ \ell+1}(t)]_{\text{if } k \geq 1}$$
$$+ [\lambda P_{k+1\ \ell-1}(t)]_{\text{if } \ell \geq 1} + [\lambda P_{0\ell}(t)]_{\text{if } k=1}$$
$$\text{for } k, \ell \geq 0 \text{ and } k+\ell \leq n$$

subject to the conditions

$$P_{k\ \ell}(0) = \begin{cases} 1 & \text{if } k = \ell = 0 \text{ and} \\ 0 & \text{otherwise, and} \end{cases}$$

$$P_{k\ \ell}(t) = 0 \quad \text{for } k+\ell > n.$$

The terms in brackets are included only when their conditions are satisfied. The solutions to the equations also must satisfy the conditions

$$\sum_{k,\ell \geq 0} P_{k\ \ell}(t) = 1 \quad \text{for all } t \geq 0 \text{ and}$$

$$P_{k\ \ell}(t) \geq 0 \qquad \text{for all } k, \ell, t \geq 0.$$

We cannot say much about the distribution of Z_t for finite t. The quantity $E_t + 2F_t$ in the unrestricted case, however, is a simple birth and death process and has been studied in detail (see [3]).

It is well known that since Z_t is an irreducible continuous parameter Markov chain the limit $\lim_{t\to\infty} P_{k\ell}(t) = p_{k\ell}$ exists. These limits, called the stationary probabilities, can be obtained by setting the derivatives in the differential equations equal to zero to obtain the recurrance equations:

$$[\lambda p_{k\ell}]_{\text{if } \ell\neq n} + (k+2\ell)\mu p_{k\ell} = (k+1)\mu p_{k+1\,\ell} + [2(\ell+1)\mu p_{k-1\,\ell+1}]_{\text{if } k\geq 1} + [\lambda p_{k+1\,\ell-1}]_{\text{if } \ell\geq 1} + [\lambda p_{0\ell}]_{\text{if } k=1}$$

for $k, \ell \geq 0$ and $k+\ell \leq n$.

The side conditions $\sum_{k,\ell\geq 0} p_{k\ell} = 1$ and $p_{k\ell} = 0$ for $k + \ell > n$ are needed for the solution. Similar equations hold for the unrestricted model and can be found by setting $n = \infty$ with the side condition $p_{k\ell} \geq 0$. Numerical solutions to these equations are given later.

To study the moments of $p_{k\ell}$ it is useful to consider the exponential generating function

$$H(s_1, s_2) = \sum_{k,\ell\geq 0} p_{k\ell}\, e^{k s_1 + \ell s_2}.$$

With this generating function the coefficient of $s_1^i s_2^j$ gives the value of the moment $\sum_{k,\ell\geq 0} k^i \ell^j p_{k\ell}$. By multiplying the recurrance equations for the unrestricted case by $e^{k s_1 + \ell s_2}$ and summing, one gets the following partial differential equation:

$$\lambda[1-e^{s_2-s_1}]H(s_1,s_2) = [e^{-s_1}-1]\mu\frac{\partial}{\partial s_1}H(s_1,s_2)$$

$$+ 2[e^{(s_1-s_2)}-1]\mu\frac{\partial}{\partial s_2}H(s_1,s_2) + \lambda[e^{s_1}-e^{(s_2-s_1)}]H(-\infty,s_2),$$

where $H(-\infty,s_2) = \sum_{\ell} p_{0\ell} e^{\ell s_2}$. The presence of this last term makes it difficult to solve this equation. One can, however, obtain some useful relations by letting

$$H(s_1,s_2) = \sum_{k,\ell\geq 0} a_{k\ell} s_1^k s_2^\ell \quad \text{and} \quad H(-\infty,s_2) = \sum_{\ell\geq 0} b_\ell s_2^\ell .$$

Then we get relations among the moments $a_{k\ell}$, and the moments along $k=0$, b_ℓ, such as

$$a_{10} = \rho b_0$$

$$a_{11} = \frac{1}{2}\rho - \frac{1}{2}\rho b_0$$

$$a_{20} = \frac{2}{3}\rho b_1 - \frac{1}{3}\rho^2 b_0 + \frac{1}{3}\rho + \frac{2}{3}\rho b_0$$

$$a_{11} = \frac{1}{3}\rho b_1 + \frac{1}{3}\rho^2 b_0 - \frac{1}{3}\rho + \frac{1}{3}\rho b_0$$

$$a_{02} = \frac{1}{4}\rho^2 - \frac{1}{2}\rho b_1 - \frac{1}{4}\rho^2 b_0 + \frac{1}{2}\rho - \frac{1}{2}\rho b_0$$

where $\rho = \lambda/\mu$. Since $a_{10} + b_0 = \sum_{\ell\geq 0} p_{0\ell} + \sum_{k,\ell\geq 0} kp_{k\ell} \geq \sum_{k,\ell\geq 0} p_{k\ell} = 1$, the equation $a_{10} = \rho b_0$ leads to the limits $b_0 \geq 1/(1+\rho)$ and $a_{10} \geq \rho/(1+\rho)$ which are useful when ρ is small.

2.2 Requests of space generated by the bottom level

We will now consider how often the bottom level requests cells from the next level. Let the random variable $S_{k\ell}$ denote the total time elapsed from the time the system enters the state (k, ℓ) until it next requests a block from the next level (i.e. $E_t + F_t$ increases). Since Z_t is a strong Markov process, we can describe $S_{k\ell}$ as follows:

$$S_{k\ell} = \inf\{s: \lim_{h\to 0} [E_{t_0+s+h} + F_{t_0+s+h} - E_{t_0+s} - F_{t_0+s}] > 0\}$$

where $Z_{t_0} = (k, \ell)$.

Let $G_{k\ell}(t)$ be the Laplace transform of the density of $S_{k\ell}$ (i.e. $G_{k\ell}(t) = E\{e^{-t S_{k\ell}}\}$)† and let $T_{k\ell}$ be the time the system stays in state (k, ℓ). Then $S_{k\ell}$ equals $T_{k\ell}$ plus the time from the next state until the next order. Now, in the unrestricted system, a transition from state (k, ℓ) is to $(k+1, \ell-1)$ with probability $2\ell\mu/(\lambda+(k+2\ell)\mu$, to $(k-1, \ell)$ with probability $k\mu/(\lambda+(k+2\ell)\mu)$, if $k > 0$ to $(k-1, \ell+1)$ with probability $\lambda/(\lambda+(k+2\ell)\mu)$, and if $k=0$ to $(k+1, \ell)$ (causing a request on the next level) with probability $\lambda/(\lambda+2\ell\mu)$. Thus

$$G_{k\ell}(t) = E\{e^{-t T_{k\ell}}\}\left[\left(\frac{\lambda}{\lambda+(k+2\ell)\mu}\right) G_{k-1\ \ell+1}(t) + \left(\frac{k\mu}{\lambda+(k+2\ell)\mu}\right) G_{k-1\ \ell}(t) + \left(\frac{2\ell\mu}{\lambda+(k+2\ell)\mu}\right) G_{k+1\ \ell-1}(t)\right] \quad \text{for } k > 0 \text{ and}$$

---

† We use the notation $E(X)$ or $E\{X\}$ to denote the mathematical expectation (expected value) of the random variable X. We shall denote the variance of X by $\text{var}(X)$.

$$G_{0\ell}(t) = E\{e^{-t\,T_{0\ell}}\}\left[\left(\frac{\lambda}{\lambda+2\ell\mu}\right) + \left(\frac{2\ell\mu}{\lambda+2\ell\mu}\right) G_{1\ \ell-1}(t)\right].$$

Since Z_t is a Markov process, and thus has exponential waiting times,

$$E\{e^{-t\,T_{k\ell}}\} = \frac{\lambda+(k+2\ell)\mu}{\lambda+(k+2\ell)\mu+t}.$$

Thus we find the equations

$$(\lambda+(k+2\ell)\mu+t)\,G_{k\ell}(t) = \lambda G_{k-1\ \ell+1}(t) + k\mu G_{k-1\,\ell}(t) + 2\ell\mu G_{k+1\ \ell-1}(t)$$

for $k > 0$, and

$$(\lambda+2\ell\mu+t)\,G_{0\ell}(t) = \lambda + 2\ell\mu\, G_{1\ \ell-1}(t),$$

which can be solved recursively. By differentiating these equations and letting

$$a_{k\ell} = E(S_{k\ell}) = -\frac{d}{dt}G_{k\ell}(0) \quad \text{and}$$

$$b_{k\ell} = E(S_{k\ell}^2) = \frac{d^2}{dt^2}G_{k\ell}(0)$$

we find that

$$(\lambda+(k+2\ell)\mu)\,a_{k\ell} = [\lambda a_{k-1\ \ell+1} + k\mu a_{k-1\ \ell}]_{\text{if } k\geq 1} + 2\ell\mu a_{k+1\ \ell-1} + 1$$

and

$$(\lambda+(k+2\ell)\mu)b_{k\ell} - 2a_{0\ell} = [\lambda b_{k-1\ \ell+1} + k\mu b_{k-1\ \ell}]_{\text{if } k\geq 1} + 2\ell\mu b_{k+1\ \ell-1}.$$

These equations can be solved to find $E(S_{k\ell}) = a_{k\ell}$ and $\text{var}(S_{k\ell}) = b_{k\ell} - a_{k\ell}^2$.

To find the mean and variance of the time between orders to the next level once the process has become stationary let q_ℓ be the

stationary probability that we are in the state $(1,\ell)$ given that an order to the next level has just taken place. In other words

$$q_\ell = \lim_{t\to\infty} \text{Prob}\{Z_t = (1,\ell) | E_{t-} = 0,\ E_{t+} = 1\}.$$

Then q_ℓ is proportional to the probability that we arrive at $(1,\ell)$ from $(0,\ell)$ given that a transition takes place. This probability is equal to $p_{0\ell}\lambda/(\lambda+2\ell\mu)$. Thus

$$q_\ell = \frac{p_{0\ell}\lambda/(\lambda+2\ell\mu)}{\sum_{i\geq 0} p_{0i}\lambda/(\lambda+2i\mu)}.$$

The mean time between requests to the next level is

$$\sum_{\ell\geq 0} a_{1\ell}\, q_\ell$$

and the variance is

$$\sum_{\ell\geq 0} b_{1\ell}\, q_\ell - \left(\sum_{\ell\geq 0} a_{1\ell}\, q_\ell\right)^2.$$

Higher moments can be calculated in a similar manner. Numerical results from solving these equations are given later.

2.3 Other statistics for the requests

Thus far we have assumed that the requests (arrivals) follow a Poisson process; that is the interarrival times on the bottom level are independent and exponentially distributed. We now drop the assumption that they are exponentially distributed but will assume that they are independently and identically distributed according to some arbitrary distribution function $G(t)$, with $G(0) = 0$. We still assume that the service times are independent and exponentially distributed.

Now if $G(t)$ is not exponential, Z_t is no longer a Markov process. We do, however, have an embedded Markov chain. Let $t_0 = 0$ and t_n be the time of the n-th arrival on the bottom level. The difference $t_n - t_{n-1}$ has the distribution G. Let $Z_n = Z_{t_n^-} = (E_{t_n^-}, F_{t_n^-})$ define a discrete time stochastic process. In fact $\{Z_n\}$ is a Markov chain. To see this, let W_n be the number of pairs which are full at time t_n^+, both cells of which complete service before time t_{n+1}; let X_n be the number of pairs which are full at time t_n^+, one cell of which completes service before time t_{n+1}; and let Y_n be the number of pairs which are (half) empty at time t_n^+ and which complete service by time t_{n+1}. Since the service times are exponentially distributed, the conditional probability distribution of (W_n, X_n, Y_n) given Z_n is independent of $Z_1, Z_2, \cdots, Z_{n-1}$. Now

$$Z_{n+1} = \begin{cases} Z_n - (Y_n - X_n + 1, W_n + X_n - 1) & \text{if } E_{t_n^-} > 0 \\ Z_n - (Y_n - X_n - 1, W_n - X_n) & \text{if } E_{t_n^-} = 0 \end{cases}$$

so the conditional probability distribution of Z_{n+1} given Z_n is independent of $Z_1, \cdots, Z_{n-1}$, and the chain $\{Z_n\}$ is Markov.

Let $P_{k\ell}^{ij} = \text{Prob}\{Z_{n+1} = (i,j) \,|\, Z_n = (k,\ell)\}$, the one-step transition probabilities for $\{Z_n\}$. Then

$$P_{k\ell}^{ij} = \int_0^\infty P_{k\ell}^{ij}(t)\, d\, G(t), \quad \text{where}$$

$$P_{k\ell}^{ij}(t) = \text{Prob}\{Z_{n+1}(i,j) \,|\, Z_n = (k,\ell) \quad \text{and} \quad t_{n+1} - t_n = t\}.$$

Now for the unrestricted system

$$P_{k\ell}^{ij}(t) = \begin{cases} \sum_m \binom{\ell+1}{j} \binom{\ell+1-j}{m} \binom{k-1}{i-m} 2^m \left(1-e^{-\mu t}\right)^{2(\ell-j)+k-i+1} e^{-\mu(2j+i)t} \\ \qquad \text{if} \quad k>0, \ \ell+1 \geq j, \ \text{and} \ 2\ell+k+1 \geq 2j+1; \\ \sum_m \binom{\ell}{j} \binom{\ell-j}{m} \binom{1}{i-m} 2^m \left(1-e^{-\mu t}\right)^{2(\ell-j)+1-i} e^{-\mu(2j+i)t} \\ \qquad \text{if} \quad k=0, \ \ell \geq j \ \text{and} \ 2\ell+1 \geq 2j+1; \ \text{and} \\ 0 \quad \text{otherwise} \end{cases}$$

where the sums are all finite because the binomal coefficients vanish for extreme values of m. From these equations $P_{k\ell}^{ij}$ can be found. Letting $q_{ij} = \lim_{n\to\infty} \text{Prob}\{Z_n = (i,j)\}$, the stationary distribution for the chain $\{Z_n\}$ can be found by solving

$$q_{ij} = \sum_{k,\ell \geq 0} q_{k\ell} \, P_{k\ell}^{ij}$$

subject to

$$\sum_{i,j \geq 0} q_{ij} = 1.$$

A similar analysis can be done if the service times are general and the interarrival times are exponential. An embedded Markov chain is then obtained by looking at the process only at times when service is completed. Looking at the system only at these times, however, may not always give an accurate picture of the behavior of the system, since the distribution of times when service is completed will depend on the state of the system.

2.4 All the levels

We will now consider briefly how to obtain equations when all levels are considered together. Such equations take much longer to solve than the corresponding equations for the bottom level. First let us consider the number of states the memory managed by the buddy system can be in. If the system has one cell, there are two states: the cell is either empty or full. We can number these states 0 and 1 respectively. Proceeding inductively, assume we have numbered the possible states for systems with 2^k cells. A system with 2^{k+1} cells can be viewed as a pair of subsystems, each with 2^k cells. Thus any state in the system corresponds to a pair of numbers (i_1,i_2), where i_1 is number of the state of the first subsystem, i_2 is the number of the second. Since the state (i_1,i_2) is equivalent to (i_2,i_1), we may assume that $i_1 \geq i_2$. We also have the additional state where all 2^{k+1} cells are reserved in a single block. The states may be numbered as $\binom{i_1}{2} + i_2$ and $\binom{n_k+1}{2} + 1$ where n_k is the total number of states for a system of 2^k cells. We can see that the number of states increases rapidly as the number of levels increases (See table 1). We note that this model ignores the arrangement of cells on the available space list. A model which represented this aspect of the system would have a much larger number of states.

The stationary probabilities for an n level buddy system obey the following equations:

$$\left\{\sum_i [\lambda_i]_{\text{if } k \in S_i} + \left(\sum_i N_{ki}\ \mu_i\right)\right\} p_k$$

$$= \sum_{\substack{k' \in U_{ki} \\ i}} M_{k'k}\ \mu_i\ p_{k'} + \sum_{\substack{k' \in L_{ki} \\ i}} K_{k'k}\ \lambda_i\ p_{k'}$$

where

1) the sums on i have $0 \leq i \leq n$,
2) λ_i is the rate of requests for blocks of size 2^i,
3) μ_i is the decay rate for blocks of size 2^i,
4) S_i is the set of states that have a free block of size 2^i or larger,
5) N_{ki} is number of blocks of size 2^i in use in state k,
6) U_{ki} is set of states that can decay to step k with a single block of size 2^i being made free,
7) $M_{k'k}$ is the number of ways state k' can go to state k with a single decay,
8) L_{ki} is the set of states that can go to state k by a single request for a block of size 2^i, and
9) $K_{k'k}$ is probability that state k' will go to state k when a request arrives.

A good approximation to $K_{k'k}$ can be obtained by assuming the blocks on each available space list are in random order. With this assumption

$$K_{k'k} = \begin{cases} \dfrac{B_{k'k}}{C_{k'i}} & \text{if } k \in A_{k'i} \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

where

1) 2^i is the size of the request that takes k' to k,

2) $A_{k'i}$ is the set of states that results when any free cell in k' is used to fill the request of size 2^i so long as the block is of the smallest available size (but no smaller than 2^i),

3) $C_{k'i}$ is the number of blocks of the smallest available size (but no smaller than 2^i) for state k, and

4) $B_{k'k}$ is the number of ways a block in state k' of smallest available size (but no smaller than 2^i) can be chosen for filling such that state k is produced.

For systems with a small number of levels (say 3) it is feasible to program a computer to produce and solve these equations, but more work is needed to find a practical way to handle systems with a large number of levels.

One can also find equations that keep track of the structure within blocks below some level but do not keep track of the higher levels (as we did for the bottom level). Keeping track of just two levels, however results in $\binom{n+9}{9}$ states where $4n$ is the number of

cells available. Later we give a few results for a small system where we keep track of 2 levels but only permit orders on the bottom level (here we have $\binom{n+5}{5}$ states). Techniques similar to the ones used here can also be applied to systems where those requests which arrive when no space is available are not turned away, as in our model, but are saved until space does become available.

3. NUMERICAL RESULTS

The equations for the stationary distribution of the bottom level of the restricted model with Poisson requests and exponential service times can be solved in a time proportional to the number of variables, $\binom{n+2}{2}$. It was therefore possible to investigate the model under many conditions.

Table 2 shows the results of varying the ratio of the request rate and the service time for systems which have enough cells to meet nearly all requests. For each value of $\rho = \lambda/\mu$, which we shall call the traffic intensity, we have checked to be sure that increasing the number of cells $(2n)$ by 20% does not change the tabulated numbers by more than a few parts in 10^8. For all values of ρ of 0.4 or above, we have found that reducing n by 20% will change the tabulated numbers by at least several parts in 10^8. Thus these results should reflect the behavior of the unrestricted model.

We let M_E denote the expected number of blocks in use paired with empty blocks, which measures how much unused space is tied up

in the available space list for the bottom level and thus not available for use on the next level. The values of $b_i = \sum_{\ell} \ell^i p_{0\ell}$ can be used in the formulas found from the exponential generating function to compute the various moments of the number of empty pairs and the number of full pairs. The mean waiting time for orders to the next level, W, indicates how fast the buddy system runs. The variance, V, of this time may be useful in calculating the effect of the bottom level on the next level.

For $\rho = 10$ and $\rho = 100$ the probabilities $p_{k\ell}$ of various values of k and ℓ are given in figures 1, 2, 3, and 4.

One is usually interested in various characteristics of the buddy system for large values of ρ. Table 3 shows the results of fitting the original eight-significant-figure data that was used to prepare table 2 with polynomials in $\rho^{1/2}$. The leading term was selected by noting which half-integer power of ρ gave the best fit. Then fits were made with 1, 2, 3, 4, and 5 terms of a decreasing power series in $\rho^{1/2}$ to the last 1, 2, 3, 4, and 5 values in table 2 for each item in table 3, the number of values always being the same as the number of terms. Fits were also made to the next to last 1, 2, 3, 4, and 5 values. Table 3 shows the results of the 4 parameter fit. The number of figures reported has been chosen so that the coefficients do not differ by more than 3 in the last figure in the three following situations: 1) 3 parameters are used in the fit, 2) 5 parameters are used in the fit, and

3) 4 parameters are used, but the data starting with the next to last entry ($\rho = 200$) is used. We therefore feel safe in using these series with values of ρ other than those used in the fit. Table 4 gives various moments of the numbers of full and empty pairs which can be calculated with the fits to b_0 and b_1. It should be noted that the process cannot be asymptotically normal because the mean and variance of the number of empty cells are of the same order for large ρ, while the probability that the number of empty pairs is negative is zero.

If one wishes formulas for small values of ρ, he should find algebraic solutions to the equations for a system with a small number of cells. Table 2 shows that for $\rho \leq .2$, four pairs of cells is enough to obtain very accurate solutions for any system with four or more pairs of cells.

In table 5 the results for a complete buddy system with 8 cells are given for the case that all orders come in on the bottom level and $\rho = 1$.

4. SIMULATIONS

A number of simulations were run to investigate features of the buddy system that were inconvenient or impossible to calculate directly. For the simulations, a random number generator of the form $x_{i+1} \leftarrow (7577 \times x_i + c) \bmod 2^{25}$ was used. Various odd numbers were used for c. In all of the simulations, the arrivals followed a Poisson process and the service times of blocks were exponentially distributed.

Since the simulation gives information about the system as a function of time, it was necessary to investigate how rapidly the system approaches stationarity so that the results of this section can be compared with those of the previous section. Figures 5 and 6 show for $\rho = 10$ (all orders coming in on the bottom level), E_i, the mean number of blocks with 2^i cells (where some of those cells are in use) that have a buddy which is empty. These results are shown as a function of the total number of orders received. The average is over the entire length of the simulation up to that point. The error bars show the standard deviation to be expected from one run. Under each error bar the number of runs made is indicated. Figures 7 and 8 show similar information for $\rho = 100$. These figures show that we reach a stationary distribution much faster for $\rho = 10$ than we do for $\rho = 100$. By 10000 orders for $\rho = 10$ and by 50000 orders for $\rho = 100$ we are so close to a stationary distribution that we can neglect any error caused by the fact that we have not yet achieved stationarity; The variation from one run to the next is much more significant than the error due to a lack of stationarity.

Table 6 gives various statistics for $\rho = 1$, 10, and 100 where all of the orders were coming in on the bottom level. Figures 9 and 10 and tables 7, 8, and 9 give statistics for $\lambda_0 + \lambda_1 = 10$, $\mu_0 = 1$, and $\mu_1 = 1$ where λ_0 is the arrival rate on the bottom level, λ_1 is the

arrival rate on the next level, μ_0 is the decay rate for the bottom level, and μ_1 is the decay rate for the next level (for blocks that were ordered from outside, thus μ_1 does not count the decay of blocks that were split up to meet orders on the bottom level). Most of those results of the simulation which can be compared with the calculations of the previous section are in good agreement. The results for W_1, the mean time between orders from the bottom level to the next level, and for V_1, the variance of that time, however, while close to the calculated values, are different by several standard deviations. We suspect the difference is caused by a slight non-random behavior of the random number generator.

There are three sets of results from the simulations that we would like to call attention to in particular. First consider setting

$$\rho_{eff} = \frac{\lambda_1}{\mu_1} + \left(\frac{E_0+F_0}{2}\right)\frac{W_1}{\sqrt{V_1}}$$

where λ_1 is the rate of requests on the next to bottom level, μ_1 is the decay rate on the next to bottom level, $(E_0+F_0)/2$ is the mean number of pairs of cells in use on the bottom level, W_1 is the mean time between orders from the bottom level to the next level, and V_1 is the variance of this time. If the resulting value of ρ_{eff} is somewhat bigger than one, then the mean number of blocks paired with empties on the next to bottom level can be estimated using ρ_{eff} in place of ρ in the equation for the

bottom level, and gives nearly the same answer for E_1 as the simulations. It would be interesting to know whether this formula would be useful at values of λ_0, λ_1, μ_0, and μ_1 other than those we have tested. Even from our data, it is clear that the method does not work exactly. Second, while it is not clear what the relation is between ρ on the bottom level and the rate at which the bottom level generates requests which go above the next level, it is evident that the fraction of requests that goes above the next level decreases as ρ increases. Thus for large values of ρ we can get a good idea of how much time the system will spend breaking up blocks by looking only at how much time is spent breaking up blocks from just one level up. Third, the mean time between orders from one level to the one above it depends almost entirely on the rate at which that level receives orders, unless the levels below it receive orders at a much higher rate. From just our one set of runs one can not tell in general when the lower levels will have a significant effect on this quantity.

5. CONCLUSIONS

We have given formulas which permit one to calculate the effect of the bottom level of the buddy system on the next level. These show that for Poisson arrivals and exponential service times the mean number of blocks tied up on the available space list for the bottom level is proportional to the square root of the mean number of blocks it is using

to fill requests. The rate at which it asks for blocks from the next level is proportional to the square-root of the traffic intensity. The simulations indicate that when the average rate of blocks being requested on one level is not small with respect to the rate of blocks being requested from lower levels, the rate at which that level orders blocks from the next higher level is not greatly effected by the rate of orders on the lower levels. Thus it appears a good indication of the running time can be computed by considering each pair of levels independently. There are also indications that the amount of space on the available space lists can be calculated from the results on pairs of levels.

6. ACKNOWLEDGEMENTS

We wish to thank Mr. Cheam for preparing some of the programs we used. We wish to thank Professor Cryer who gave us much advice on the numerical aspects of solving our equations.

TABLE 1

k	n_k
0	2
1	4
2	11
3	67
4	2279
5	2598061

The number of states, n_k, in the memory managed by a buddy system as a function of the number of levels in the buddy system, k.

Table 2 shows the results of calculations of the stationary behavior of the restricted model for the bottom level of the buddy system with Poisson requests at rate λ and exponential service times with decay rate μ. The number of pairs of cells in the system is n. The expected number of full blocks paired with empty blocks is M_E. The value of $\sum_{\ell} \ell^i p_{0\ell}$ is given by b_i. The mean time between requests to the next level is W and V is the variance of this time. In most cases only 4 significant figures are given.

TABLE 2

$\rho=\lambda/\mu$	n	M_E	b_0	b_1	b_2	b_3	$W\mu$	$V\mu^2$
0.001	4	0.000999	0.9990	0.00000050	0.00000050	0.00000050	1001.	1000001.
0.002	4	0.001996	0.9980	0.00000199	0.00000199	0.00000199	501.0	250001.
0.004	4	0.003984	0.9960	0.00000796	0.00000796	0.00000796	251.0	62501.
0.01	4	0.009901	0.9901	0.00004934	0.00004934	0.00004934	101.0	10001.
0.02	4	0.01961	0.9804	0.0001948	0.0001948	0.0001948	51.00	2501.
0.04	4	0.03846	0.9615	0.0007588	0.0007590	0.0007594	26.00	626.0
0.1	4	0.09092	0.9092	0.004388	0.004395	0.004410	11.00	101.1
0.2	4	0.1668	0.8341	0.01549	0.01559	0.01580	5.983	26.13
0.4	5	0.2875	0.7186	0.04898	0.05033	0.05303	3.444	7.386
1.	8	0.5235	0.5235	0.1691	0.1971	0.2558	1.813	1.922
2.	10	0.7690	0.3845	0.3312	0.5176	0.9623	1.193	0.9182
4.	16	1.099	0.2748	0.5437	1.397	4.165	0.8434	0.5459
10.	20	1.735	0.1735	0.9133	5.273	32.84	0.5476	0.3172
20.	32	2.441	0.1220	1.295	14.35	165.6	0.3945	0.2210
40.	64	3.433	0.08583	1.811	39.05	859.4	0.2833	0.1569
100.	100	5.394	0.05394	2.810	147.6	7821.	0.1820	0.1007
200.	150	7.600	0.03800	3.922	406.5	42315.	0.1298	0.0721
400.	300	10.72	0.02680	5.488	1126.	231627.	0.0924	0.05154

TABLE 3

Quantity	Fit for large $\rho = \lambda/\mu$
b_0	$0.53188\ \rho^{-1/2} + 0.087\ \rho^{-1} - 0.122\ \rho^{-3/2}$
b_1	$0.26594\ \rho^{1/2} + 0.187 - 0.37\ \rho^{-1/2}$
b_2	$0.13296\ \rho^{3/2} + 0.166\ \rho - 0.199\ \rho^{1/2}$
b_3	$0.06647\ \rho^{5/2} + 0.119\ \rho^{2} - 0.03\ \rho^{3/2}$
$W\mu$	$1.8800\ \rho^{-1/2} - 0.68\ \rho^{-1} + 0.8\ \rho^{-3/2}$
$V\mu^2$	$1.065\ \rho^{-1/2} - 0.80\ \rho^{-1} + 2.4\ \rho^{-3/2}$

Table 3 gives the formulas which give the best fit to the calculations of b_i, W, and V for large values of ρ.

TABLE 4

Quantity	Fit from b_0 and b_1
M_E	$0.53188\ \rho^{1/2} + 0.087 - 0.122\ \rho^{-1/2}$
M_F	$\frac{1}{2}\rho - 0.26594\ \rho^{1/2} - 0.043 + 0.061\ \rho^{-1/2}$
M_E2	$0.429\ \rho + 0.15\ \rho^{1/2}$
M_{EF}	$0.26596\ \rho^{3/2} - 0.242\ \rho + 0.01\ \rho^{1/2}$
M_F2	$\frac{1}{4}\rho^2 - .26594\ \rho^{3/2} + 1.539\ \rho + 1.92\ \rho^{1/2}$
σ_E2	$0.157\rho + 0.07\ \rho^{1/2}$
σ_F2	$1.511\ \rho + 1.87\ \rho^{1/2}$
$\rho_{EF}\sigma_E\sigma_F$	$0.154\ \rho + 0.103\ \rho^{1/2}$

Table 4 shows the mean of E, the number of pairs on the bottom level where one block is in use and its buddy is not, the mean of F, the number of pairs on the bottom level where both blocks of the buddy pair are in use, the mean of E^2, the mean of EF, the mean of F^2, the variance of E, the variance of F, and the product of the standard deviations and the correlation of E and F. These were calculated from the fits to b_0 and b_1.

Table 5 gives the probabilities of various contigurations in a 3 level buddy system when all orders are to the bottom level and $\rho = 1$. The states are numbered in the way indicated in section 2.4 on analysis. There would be 67 states if orders were permitted on all levels. The example column shows a memory configuration in the equivalence class of the state. A cell in use is indicated by a 1. A free cell is indicated with a 0. For states 4, 11, and 12 the buddy system permits one of several nonequivalent free cells to be filled, depending on the order of cells on the available space list. In the "random" column the results of the model in the text where the available space list is always in random order is given. In the "selective" column the results are given for a model where, when there are several blocks that the buddy system permits to be filled, the block that results in the highest numbered state is chosen. In the example column we have underlined the cell that will be chosen in this model. An algorithm for this model would pack the space more efficiently than the original buddy system, but it would run much more slowly.

TABLE 5

State	Example	Blocks in Use	Probability for $\rho = 1$	
			Random	Selective
0	00000000	0	0.36788	0.36788
1	10000000	1	0.36788	0.36788
2	10001000	2	0.00211	0.00201
3	10100000	2	0.03923	0.03933
4	10101000	3	0.00138	0.00138
5	10101010	4	0.00002	0.00002
6	11000000	2	0.14260	0.14260
7	11001000	3	0.00178	0.00163
8	11001010	4	0.00036	0.00017
9	11001100	4	0.00049	0.00041
10	11100000	3	0.05815	0.05830
11	11101000	4	0.00236	0.00255
12	11101010	5	0.00005	0.00005
13	11101100	5	0.00065	0.00040
14	11101110	6	0.00005	0.00004
15	11110000	4	0.01210	0.01218
16	11111000	5	0.00237	0.00261
17	11111010	6	0.00002	0.00003
18	11111100	6	0.00044	0.00044
19	11111110	7	0.00007	0.00007
20	11111111	8	0.000009	0.000009

Table 6 shows for orders coming in only on the bottom level and with $\rho = 1, 10,$ and 100 (1) E_i, mean number of blocks of size 2^i which are paired with empty blocks, (2) W_i, the mean time that passes between each order to level i from level $i-1$, and (3) V_i, the variance of this time. All runs were sufficiently long to achieve stationarity within a very close approximation.

TABLE 6

	$\rho = 1$	$\rho = 10$	$\rho = 100$
	10000 orders	10000 orders	50000 orders
E_0	$0.524 \pm 0.004(0.014)$	$1.724 \pm 0.015(0.037)$	$5.400 \pm 0.032(0.087)$
E_1	$0.519 \pm 0.004(0.014)$	$1.221 \pm 0.018(0.044)$	$4.930 \pm 0.051(0.134)$
E_2	$0.616 \pm 0.005(0.017)$	$0.806 \pm 0.018(0.045)$	$2.804 \pm 0.092(0.243)$
E_3	$0.626 \pm 0.005(0.015)$	$0.553 \pm 0.066(0.162)$	$1.47 \pm 0.15(0.39)$
E_4	$0.626 \pm 0.005(0.015)$	$0.640 \pm 0.042(0.102)$	$1.032 \pm 0.064(0.170)$
W_1	$1.921 \pm 0.020(0.063)$	$0.576 \pm 0.005(0.012)$	$0.1827 \pm 0.0011(0.0030)$
W_2	$2.623 \pm 0.023(0.072)$	$2.276 \pm 0.05(0.124)$	$1.633 \pm 0.014(0.038)$
W_3	$2.693 \pm 0.020(0.063)$	$8.15 \pm 0.37(0.90)$	$11.28 \pm 0.55(1.45)$
W_4	$2.693 \pm 0.020(0.063)$	$18.3 \pm 1.1(2.6)$	$23.8 \pm 5.1(13.4)$
V_1	$2.048 \pm 0.050(0.161)$	$0.343 \pm 0.005(0.012)$	$0.1044 \pm 0.0019(0.0051)$
V_2	$4.42 \pm 0.20(0.63)$	$4.32 \pm 0.21(0.51)$	$6.72 \pm 0.12(0.32)$
V_3	$5.01 \pm 0.21(0.65)$	$56.1 \pm 6.0(14.6)$	$287 \pm 40(107)$
V_4	$5.01 \pm 0.21(0.65)$	$325 \pm 44(109)$	$3132 \pm 899(2379)$

Table 7 shows E_i, the mean number of blocks of size 2^i paired with empty blocks as a function of λ_0 and λ_1, the arrival rates on the bottom and next levels. In all cases the decay rates were equal to one and each run consisted of 10000 orders. Each entry gives the expected value plus or minus the standard error with the standard deviation expected from one run in parentheses.

TABLE 7

λ_0	λ_1	E_0	E_1	E_2	E_3	E_4
10	0	1.724±0.015(0.037)	1.221±0.018(0.044)	0.806±0.018(0.045)	0.553±0.066(0.162)	0.640±0.042(0.102)
9	1	1.656±0.003(0.008)	1.334±0.012(0.026)	0.959±0.022(0.055)	0.762±0.033(0.081)	0.421±0.018(0.044)
8	2	1.552±0.014(0.034)	1.396±0.018(0.043)	0.994±0.025(0.060)	0.928±0.029(0.072)	0.267±0.022(0.053)
7	3	1.459±0.013(0.033)	1.456±0.009(0.021)	1.005±0.009(0.023)	0.923±0.021(0.053)	0.211±0.016(0.039)
6	4	1.356±0.013(0.030)	1.4857+0.011(0.025)	1.078±0.021(0.047)	0.892±0.015(0.033)	0.178+0.010(0.023)
5	5	1.224±0.013(0.028)	1.558±0.008(0.018)	1.076±0.016(0.036)	0.884±0.010(0.021)	0.217±0.038(0.084)
4	6	1.092±0.013(0.029)	1.591±0.011(0.025)	1.120±0.020(0.044)	0.837±0.013(0.029)	0.201±0.019(0.043)
3	7	0.958±0.014(0.030)	1.627±0.007(0.015)	1.154±0.022(0.048)	0.802±0.008(0.017)	0.332±0.061(0.137)
2	8	0.776±0.005(0.013)	1.678±0.015(0.037)	1.182±0.009(0.021)	0.758±0.022(0.054)	0.423±0.052(0.127)
1	9	0.518±0.004(0.011)	1.713±0.007(0.017)	1.211±0.021(0.052)	0.818±0.020(0.050)	0.548±0.035(0.086)
0	10	0	1.724±0.015(0.037)	1.221±0.018(0.044)	0.806±0.018(0.045)	0.553±0.066(0.162)

Table 8 shows W_i, the mean time between orders to level i from lower levels. The values of the parameters and the form of the table are the same as table 7.

TABLE 8

λ_0	λ_1	W_1	W_2	W_3	W_4
10	0	0.576±0.005(0.012)	2.276±0.051(0.124)	8.15±0.37(0.90)	18.3±1.1(2.6)
9	1	0.604±0.003(0.008)	1.615±0.023(0.056)	5.57±0.23(0.56)	10.30±0.67(1.64)
8	2	0.641±0.004(0.009)	1.307±0.024(0.059)	4.67±0.20(0.50)	12.3±1.1(2.8)
7	3	0.687±0.004(0.009)	1.068±0.009(0.023)	3.846±0.074(0.182)	10.93±0.51(1.26)
6	4	0.748±0.005(0.010)	0.932±0.009(0.020)	3.326+0.070(0.157)	10.68±0.41(0.92)
5	5	0.800±0.007(0.019)	0.834±0.010(0.021)	3.212±0.059(0.133)	11.72±0.59(1.33)
4	6	0.917±0.008(0.018)	0.756±0.003(0.008)	2.842±0.062(0.140)	9.74±0.32(0.71)
3	7	1.059±0.006(0.012)	0.688±0.004(0.009)	2.696±0.039(0.088)	9.56±0.50(1.11)
2	8	1.301±0.007(0.017)	0.638±0.006(0.015)	2.493±0.036(0.088)	9.45±0.42(1.02)
1	9	1.925±0.020(0.049)	0.597±0.003(0.007)	2.411±0.040(0.097)	9.16±0.41(0.99)
0	10	∞	0.576±0.005(0.012)	2.276±0.051(0.124)	8.15±0.37(0.90)

Table 9 shows V_i, the variance of the time between orders to level i from lower levels. The values of the parameters and the form of the table are the same as table 7.

TABLE 9

λ_0	λ_1	V_1	V_2	V_3	V_4
10	0	0.343±0.005(0.012)	4.32±0.21(0.51)	56.1±6.0(14.6)	325±44(109)
9	1	0.364±0.011(0.027)	2.481±0.074(0.181)	27.0±2.4(5.8)	157±42(104)
8	2	0.390±0.011(0.027)	1.573±0.044(0.108)	20.9±2.6(6.3)	284±70(173)
7	3	0.428±0.007(0.018)	1.095±0.008(0.019)	13.52±0.63(1.56)	212±22(54)
6	4	0.475±0.006(0.014)	0.819±0.029(0.065)	10.20±0.46(1.02)	179±14(31)
5	5	0.520±0.009(0.019)	0.673±0.025(0.055)	8.73±0.45(1.00)	216±28(63)
4	6	0.625±0.010(0.023)	0.555±0.018(0.040)	7.83±0.23(0.52)	131±12(28)
3	7	0.753±0.024(0.054)	0.483±0.007(0.017)	6.41±0.20(0.45)	110±10(23)
2	8	1.112±0.031(0.075)	0.422±0.006(0.015)	5.61±0.25(0.61)	126±17(41)
1	9	1.998±0.055(0.137)	0.364±0.004(0.011)	5.55±0.34(0.83)	80.0±9.5(23.3)
0	10	∞	0.343±0.005(0.012)	4.32±0.21(0.51)	56.1±6.0(14.6)

Figure 1 shows the probabilities of various pairings of cells on the bottom level of the buddy system for $\rho = 10$. Each curve connects together the points where the total number of pairs of cells being used is constant. The probability of each pairing on the bottom level is shown as a function of the number of cells paired with empty cells.

FIGURE 1

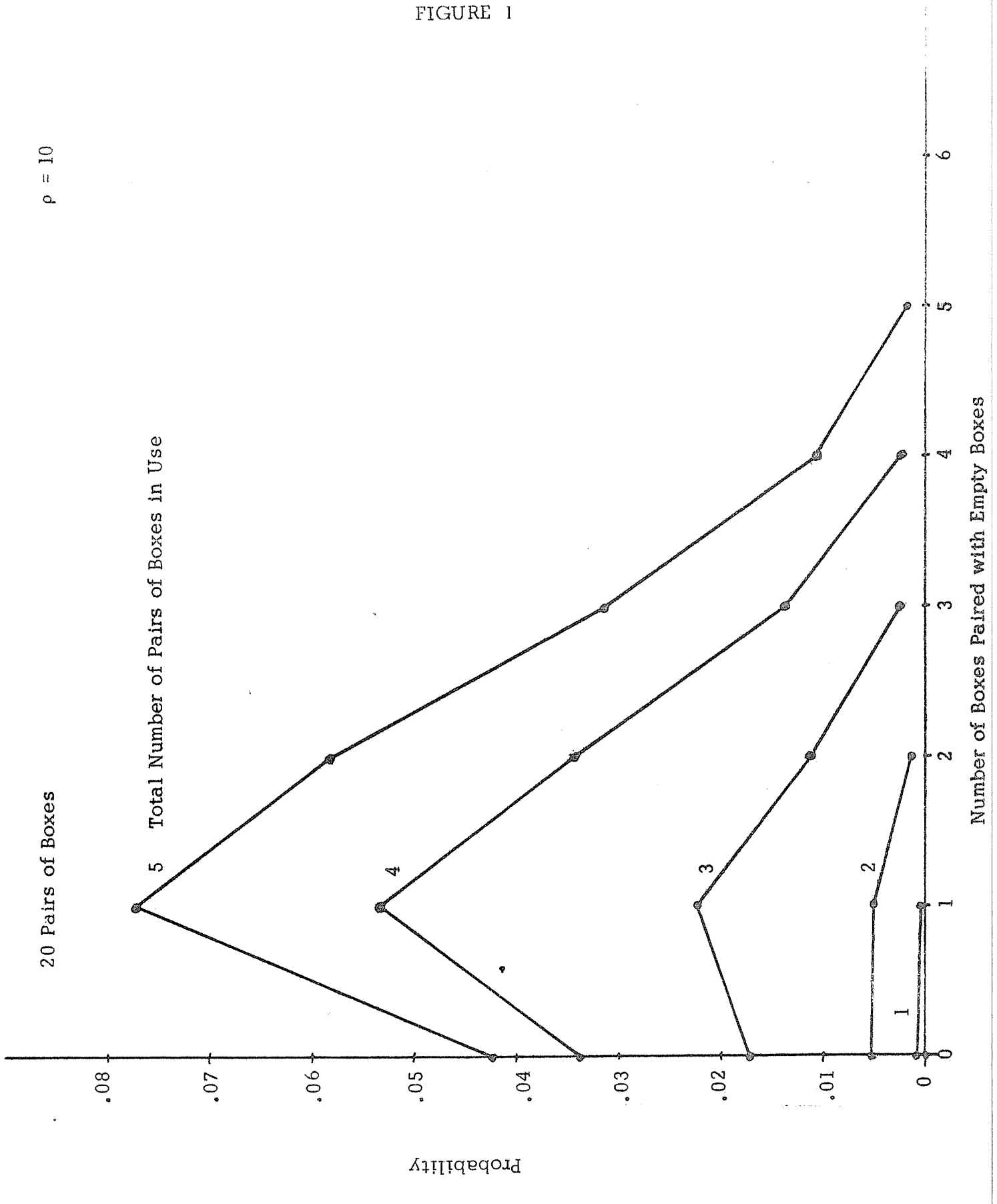

ρ = 10
20 Pairs of Boxes
5 Total Number of Pairs of Boxes in Use
4
3
2
1
Probability
.08
.07
.06
.05
.04
.03
.02
.01
0
0 1 2 3 4 5 6
Number of Boxes Paired with Empty Boxes

Figure 2 shows the probabilities of various pairings of cells on the bottom level of the buddy system for $\rho = 10$. Each curve connects together the points where the total number of pairs of cells being used is constant. The probability of each pairing on the bottom level is shown as a function of the number of cells paired with empty cells.

FIGURE 2

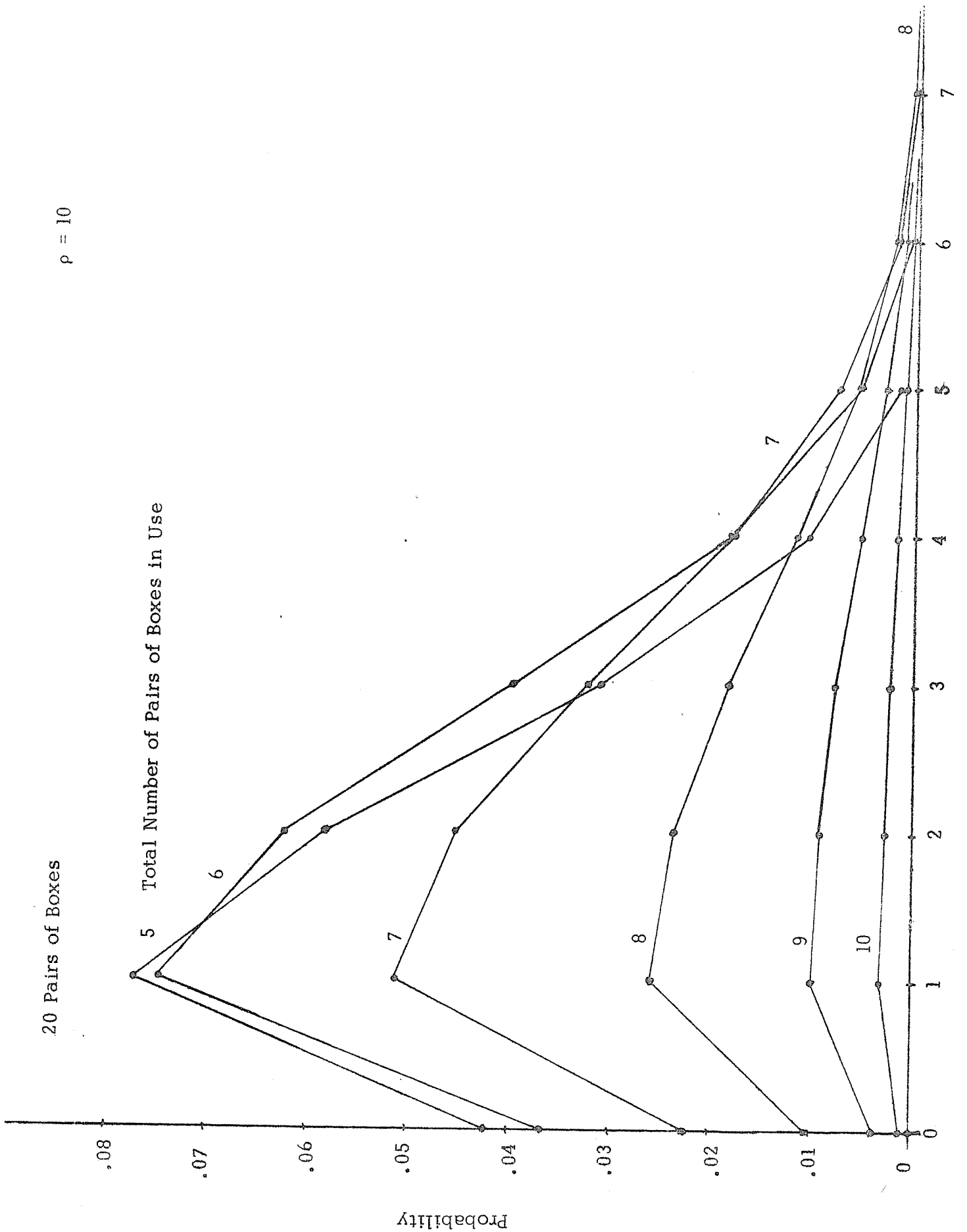

20 Pairs of Boxes
ρ = 10
Total Number of Pairs of Boxes in Use
5
6
7
8
9
10
7
Probability
.08
.07
.06
.05
.04
.03
.02
.01
0
0
1
2
3
4
5
6
7
8
Number of Boxes Paired with Empty Boxes

Figure 3 shows the probabilities of various pairings of cells on the bottom level of the buddy system for $\rho = 100$. Each curve connects together the points where the total number of pairs of cells being used is constant. The probability of each pairing on the bottom level is shown as a function of the number of cells paired with empty cells.

FIGURE 3

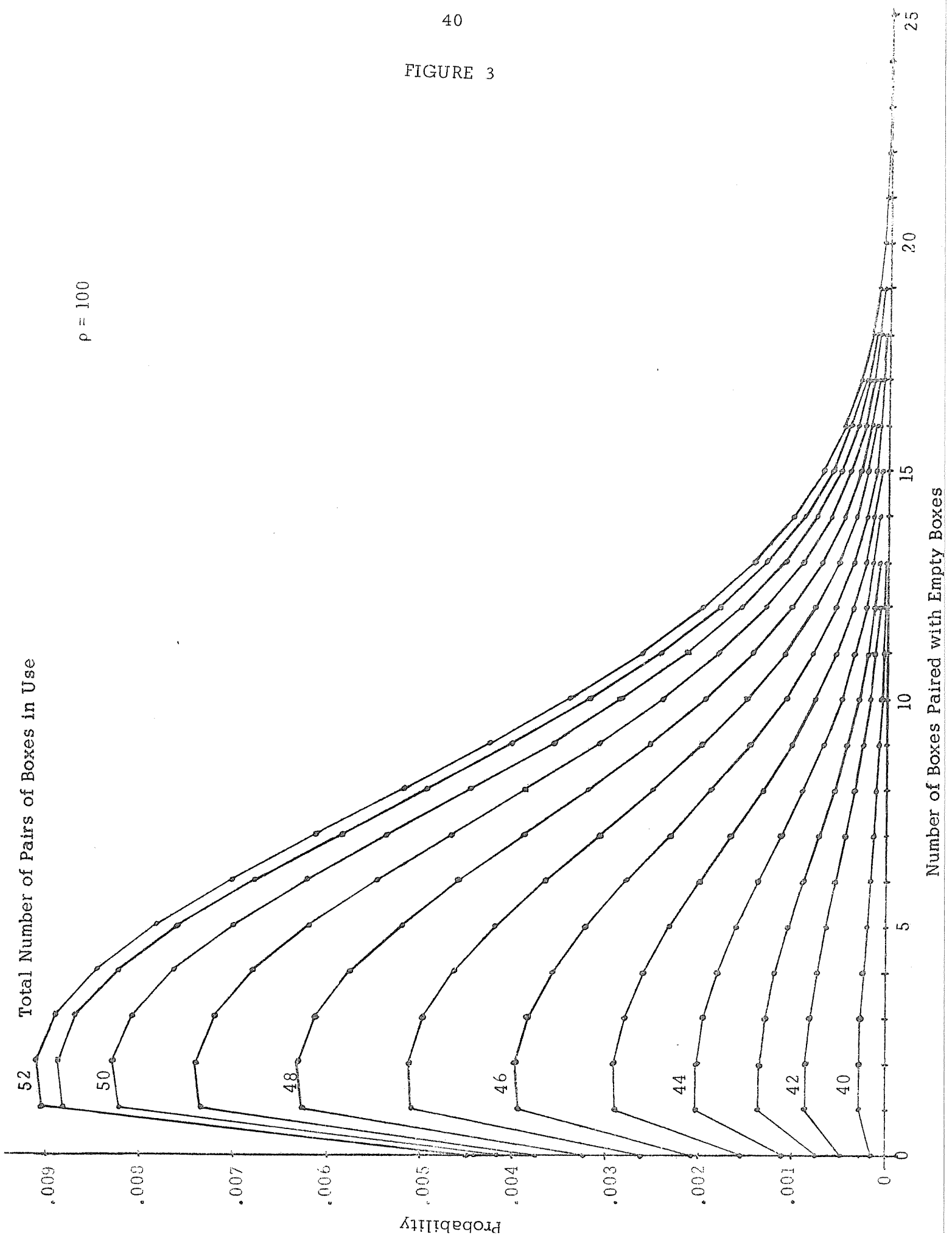

ρ = 100
Total Number of Pairs of Boxes in Use
52
50
48
46
44
42
40
Probability
.009
.008
.007
.006
.005
.004
.003
.002
.001
0
0
5
10
15
20
25
Number of Boxes Paired with Empty Boxes

Figure 4 shows the probabilities of various pairings of cells on the bottom level of the buddy system for $\rho = 100$. Each curve connects together the points where the total number of pairs of cells being used is constant. The probability of each pairing on the bottom level is shown as a function of the number of cells paired with empty cells.

FIGURE 4

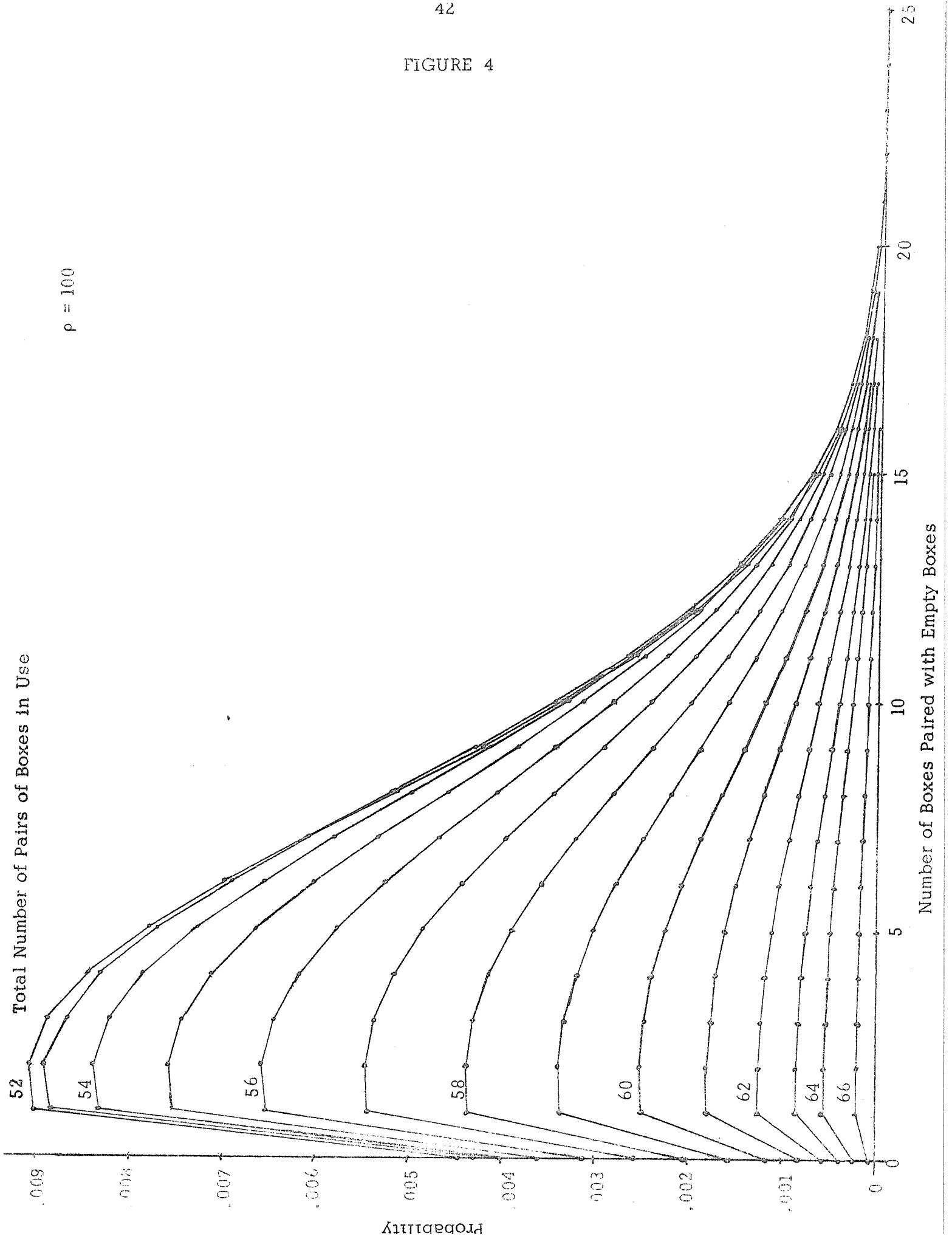

ρ = 100
Total Number of Pairs of Boxes in Use
52
54
56
58
60
62
64
66
Probability
.009
.008
.007
.006
.005
.004
.003
.002
.001
0
0
5
10
15
20
25
Number of Boxes Paired with Empty Boxes

Figure 5 shows for $\rho = 10$ (all orders coming in on the bottom level), E_i, the mean number of blocks with 2^i cells (where some of those cells are in use) that have a buddy which is empty. These results are shown as a function of the total number of orders received. The average is taken over the entire length of the simulation up to that point. The error bars show the standard deviation to be expected from one run. Under each error bar the number of runs made is indicated. To get the expected standard error in E_i at that point, the standard deviation expected from one run should be divided by the square-root of the number of runs. Since the same set of runs was used for many points, the error on adjacent points may be highly correlated. There were 3 runs of 900 orders, 3 runs of 1000 orders, and 1 run of 9000 orders which printed the results every 100 orders. There were 6 runs of 10000 orders and 1 run of 3000 orders which printed the results every 1000 orders.

FIGURE 5

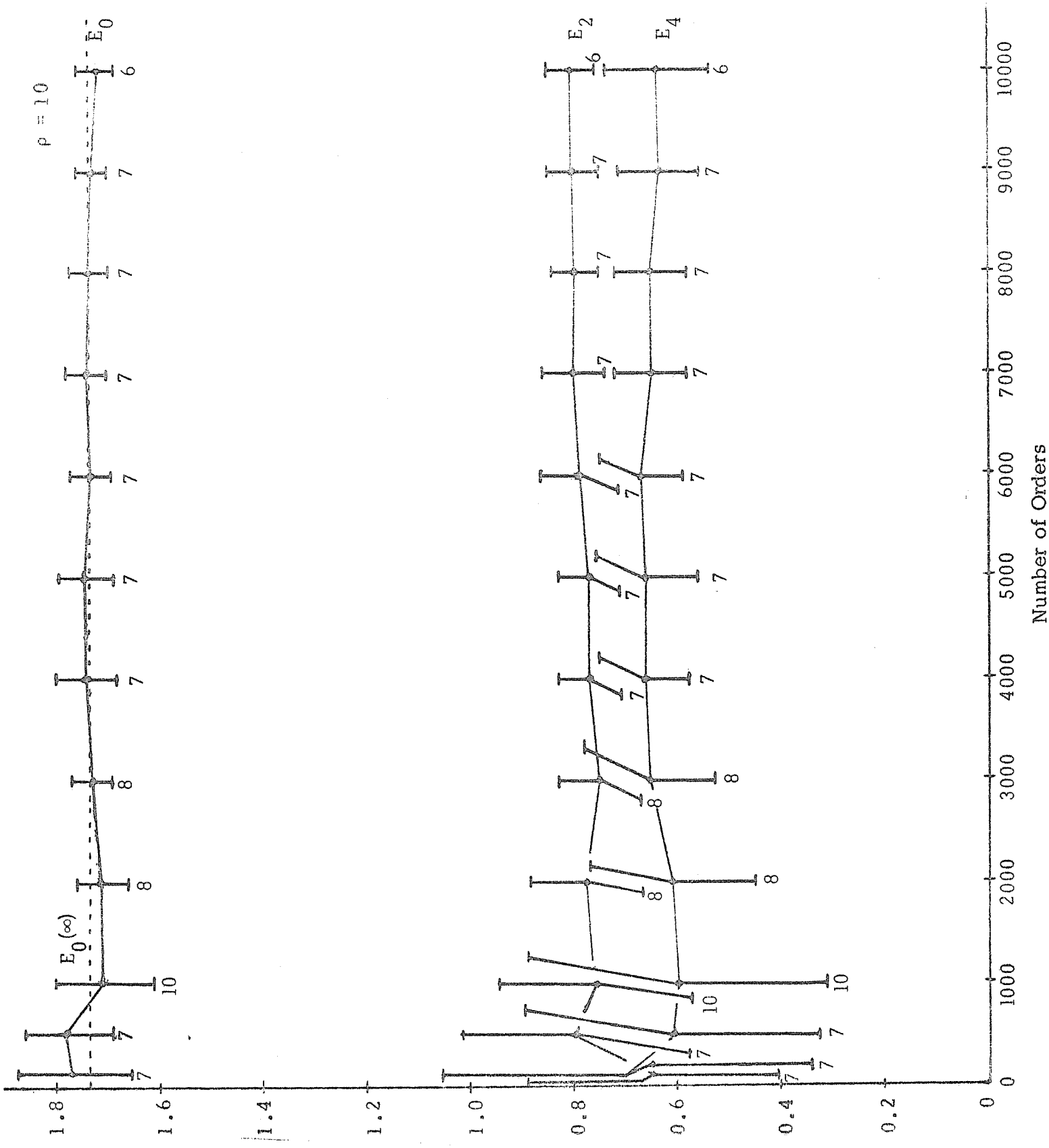

ρ = 10
E_0
$E_0(\infty)$
E_2
E_4
Mean Number of Boxes Paired with Empty Boxes
Number of Orders
0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000

Figure 6 shows for $\rho = 10$ (all orders coming in on the bottom level), E_i, the mean number of blocks with 2^i cells (where some of those cells are in use) that have a buddy which is empty. These results are shown as a function of the total number of orders received. The average is taken over the entire length of the simulation up to that point. The error bars show the standard deviation to be expected from one run. Under each error bar the number of runs made is indicated. To get the expected standard error in E_i at that point, the standard deviation expected from one run should be divided by the square-root of the number of runs. Since the same set of runs was used for many points, the error on adjacent points may be highly correlated. There were 3 runs of 900 orders, 3 runs of 1000 orders, and 1 run of 9000 orders which printed the results every 100 orders. There were 6 runs of 10000 orders and 1 run of 3000 orders which printed the results every 1000 orders.

FIGURE 6

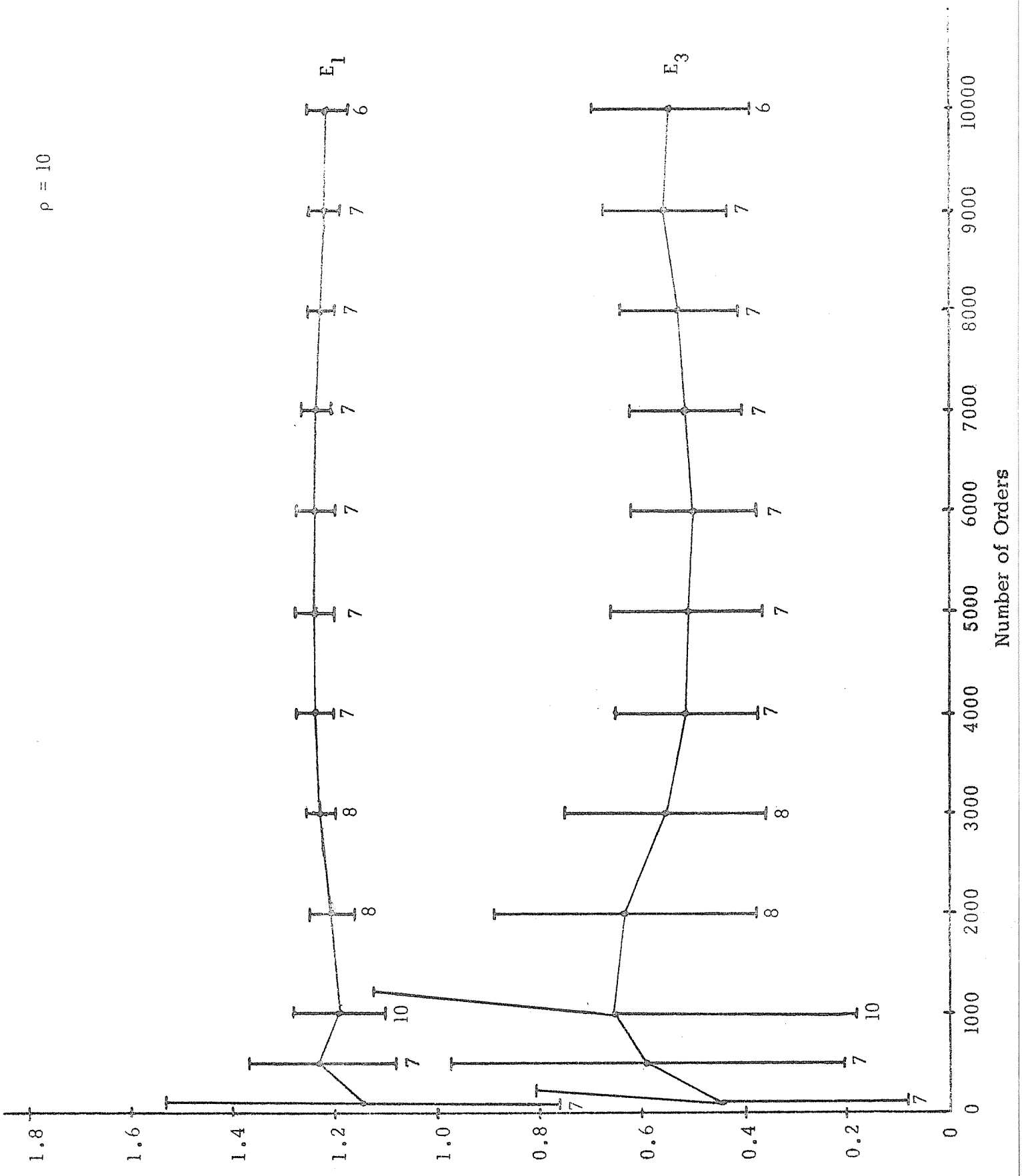

ρ = 10
E_1
E_3
Mean Number of Boxes Paired with Empty Boxes
Number of Orders
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8

Figure 7 shows the same information as figures 5 and 6 for ρ equal to 100. There were 6 runs of 9000 orders which printed results every 1000 orders and 7 runs of 50000 orders which printed results every 10000 orders.

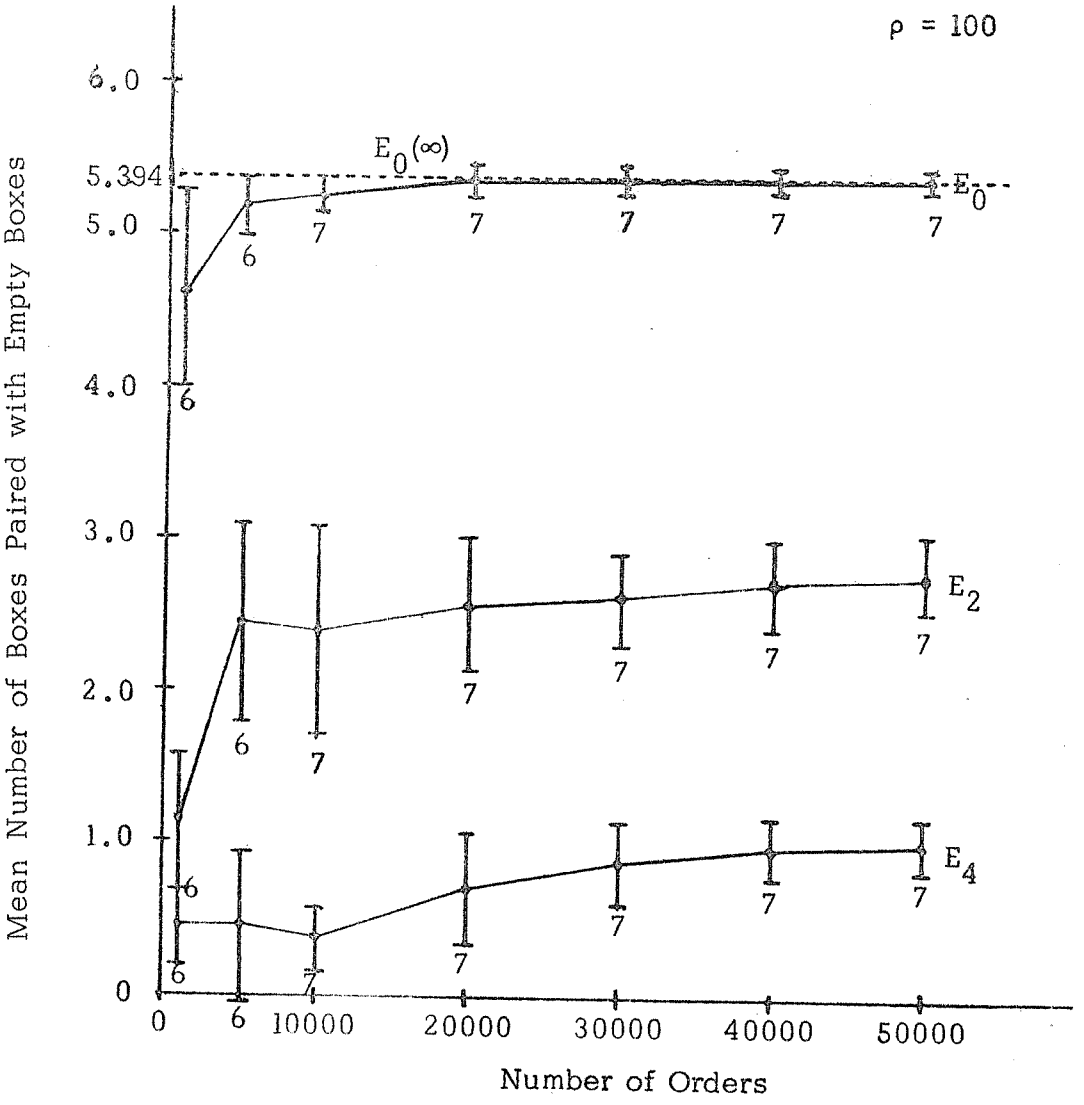


FIGURE 7

Figure 8 shows the same information as figures 5 and 6 for ρ equal to 100. There were 6 runs of 9000 orders which printed results every 1000 orders and 7 runs of 50000 orders which printed results every 10000 orders.

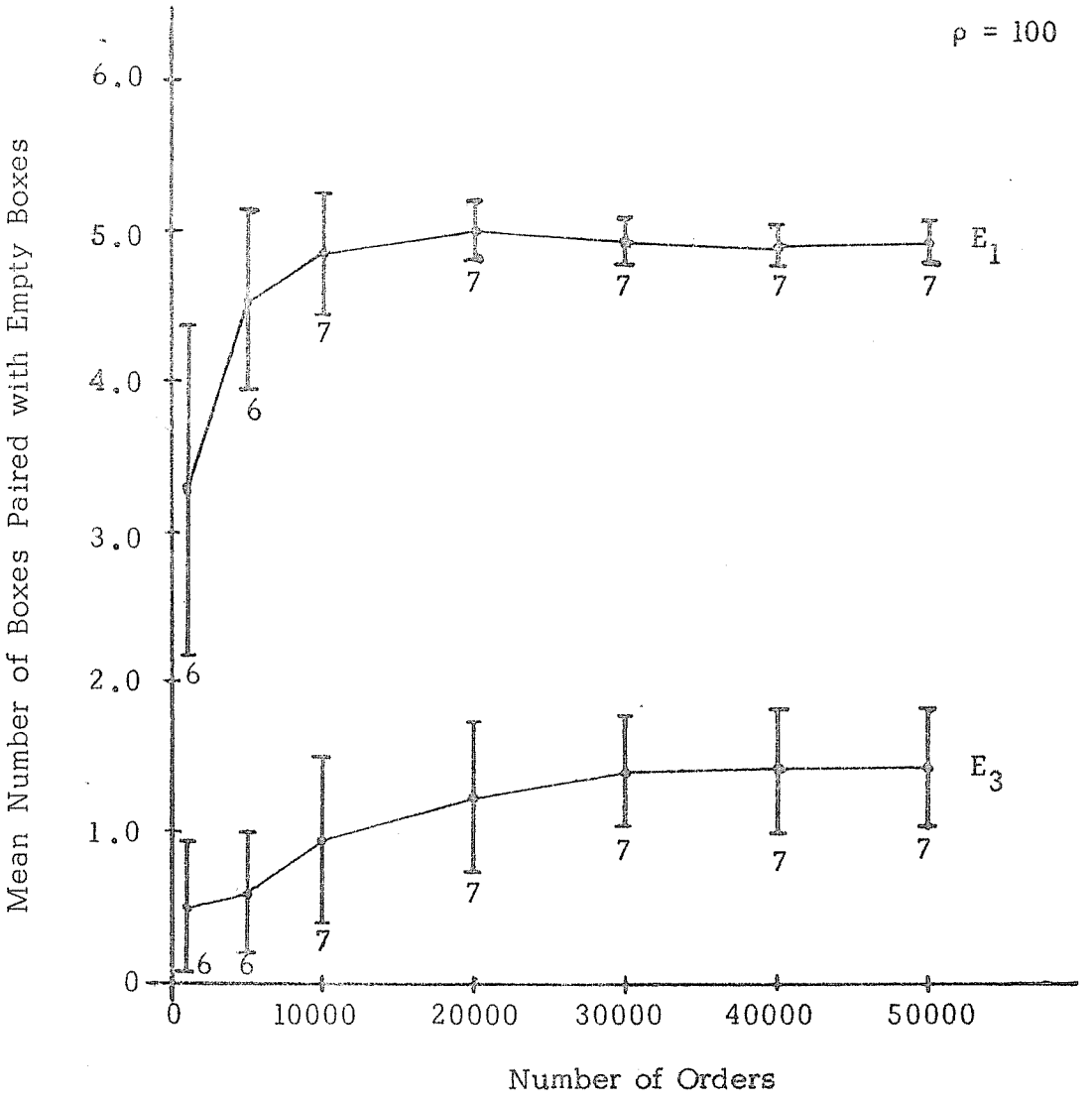


FIGURE 8

Figure 9 shows E_i, the mean number of blocks with 2^i cells which have an empty buddy. Orders from the bottom level were at rate λ_0 and from the next level at rate λ_1. In both cases the decay rate was one. Each run was for a total of 10000 orders. The number of runs for each point is shown below it. The error bars indicate the expected standard deviation from one run. To get the standard deviation for the mean, divide by the square root of the number of runs.

FIGURE 9

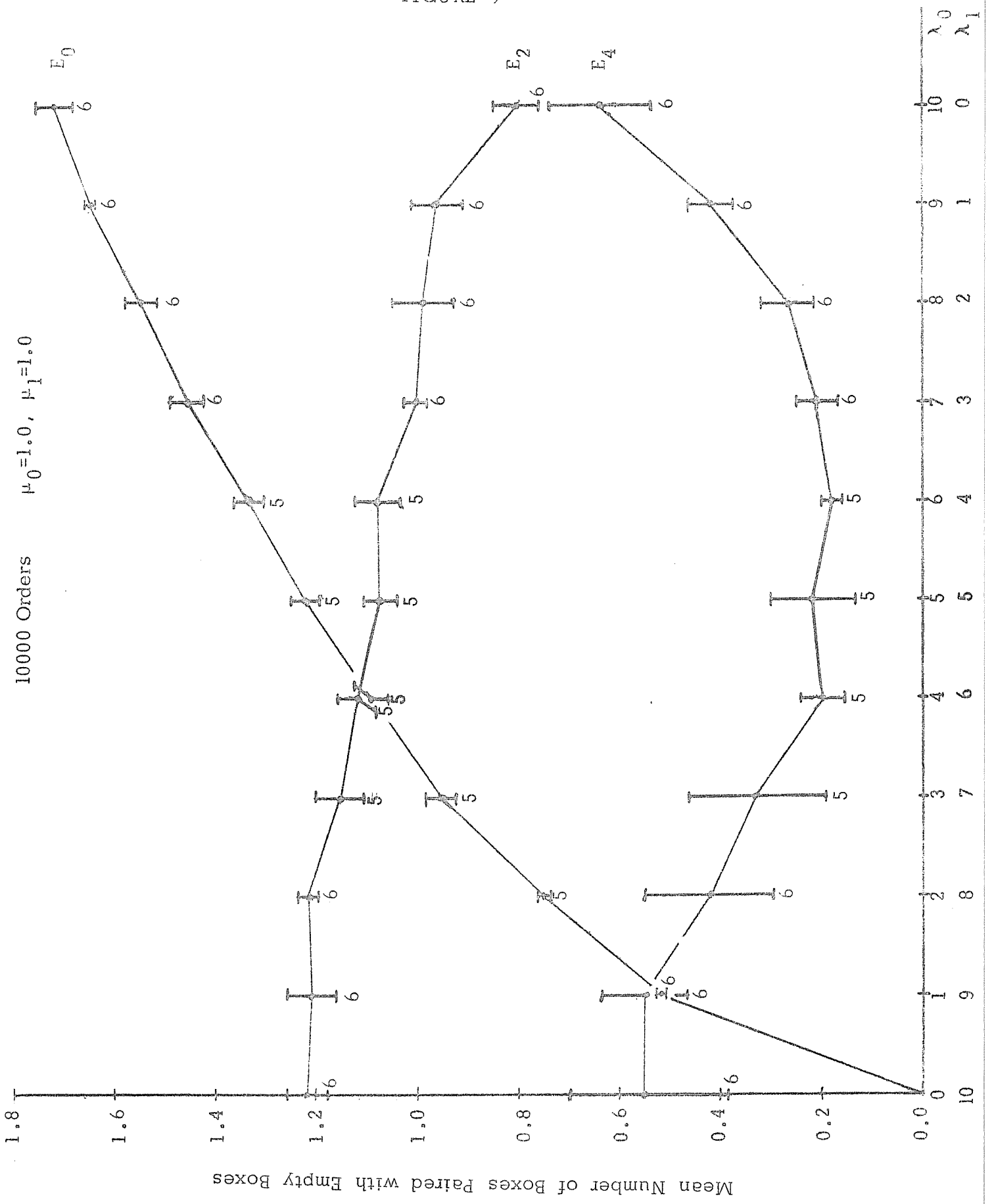
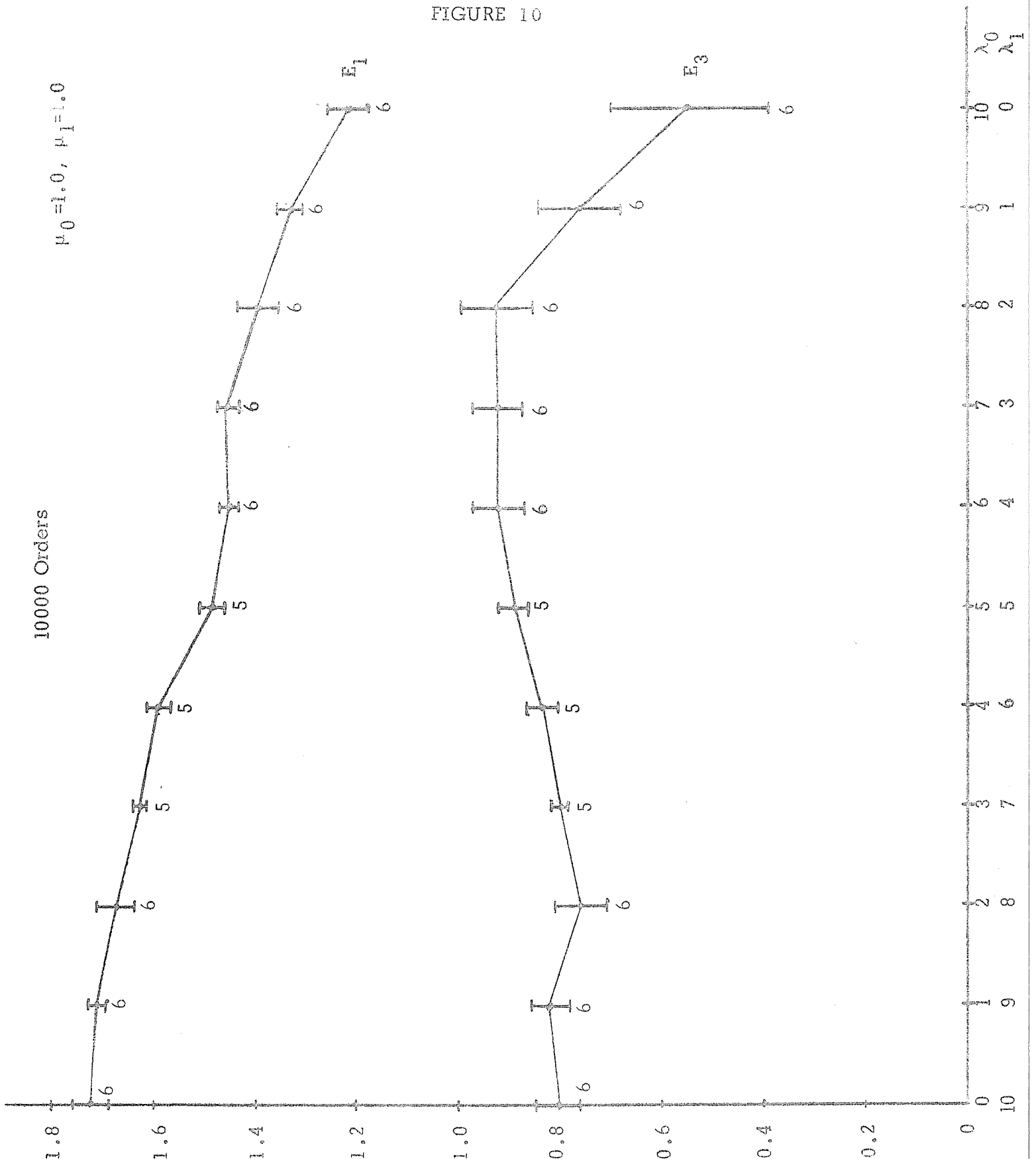

Mean Number of Boxes Paired with Empty Boxes
10000 Orders
μ_0=1.0, μ_1=1.0
E_0
E_2
E_4
λ_0
λ_1
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8

Figure 10 shows E_i, the mean number of blocks with 2^i cells which have an empty buddy. Orders from the bottom level were at rate λ_0 and from the next level at rate λ_1. In both cases the decay rate was one. Each run was for a total of 10000 orders. The number of runs for each point is shown below it. The error bars indicate the expected standard deviation from one run. To get the standard deviation for the mean, divide by the square root of the number of runs.

FIGURE 10


μ_0=1.0, μ_1=1.0
10000 Orders
E_1
E_3
λ_0
λ_1
Mean Number of Boxes Filled with Empty Boxes

References

1. Knowlton, Kenneth C., "A Fast Storage Allocator" Communications of The ACM 8 (1965) pp. 623-625.

2. Knuth, Donald E., The Art of Computer Programming, Vol. 1, Addison-Wesley, Reading, Mass. (1968), pp. 435-455.

   There is a slight error in the algorithms for the buddy system in the first printing of this volume. (The algorithms in the first printing do not keep track of the size of free blocks.)

3. Karlin, Samuel, A First Course in Stochastic Processes, Academic Press, New York (1966), pp. 189-194.