

**OPTIMAL FORAGING IN BERNOULLI PATCHES  
WHICH ARE OF TWO TYPES**

by

Richard F. Green

Technical Report 88-4

Department of Mathematics and Statistics  
University of Minnesota  
Duluth, Minnesota 55812

December, 1988

## OPTIMAL FORAGING IN BERNOULLI PATCHES WHICH ARE OF TWO TYPES

Richard F. Green  
Department of Mathematics and Statistics  
University of Minnesota  
Duluth, Minnesota 55812

### INTRODUCTION

Animals that search for food in the natural world often find prey in patches. For example, a woodpecker searching for insects in and under the bark of trees searches for some time in one tree and then flies on to search another tree. A problem that such animals face is how to decide when to leave one patch and move on to another. Eric Charnov (1976) presented a solution to this problem in a case in which the rate of finding prey in each patch is a known, deterministic, continuous, negatively-accelerated function of the time spent in the patch. Charnov claimed that his result—the “marginal value theorem,” the best-known result of optimal foraging theory—also holds for stochastic models, but he did not make clear what he had in mind. Allan Oaten (1977) developed a stochastic model to expose Charnov’s claim.

In Oaten’s model, superficially similar patches are assumed to vary in the number of prey that they contain and foraging animals must use their knowledge of the environment and their experience in each patch to decide when to leave that patch. I have found the optimal patch-leaving rule for several examples of Oaten’s model (Green 1980, 1987a). In this paper I will consider another example, perhaps the simplest one of all.

In the example that I will consider, the environment consists of a large number of patches of two types. All patches will contain a binomially-distributed number of prey, with the number of prey in “bad” patches having distribution:  $bin(n, p_1)$ ; and the number of prey in “good” patches having distribution:  $bin(n, p_2)$ , where  $p_1 < p_2$ . Patches may be distinguished only by comparing the success achieved in finding prey *within the patches*.

The purpose of this paper is to describe the model, to find the optimal patch-leaving rule for this model, and to compare the performance of the optimal rule with that of some other rules which have been suggested for similar problems. The prey distribution assumed in this paper is probably the simplest of four simple Bayesian models which yield aggregated prey distributions. I have found the optimal rules for foragers faced

with the other distributions: (1) Beta mixture of binomial distributions. This model was treated twice (Green 1980, 1984). (2) Two-point mixture of Poisson distributions. This model, suggested by Joel Myerson for a Skinner-box experiment on pigeons which was never done, was described in an unpublished paper (Green 1981). (3) Gamma mixture of Poisson distributions (= negative binomial distribution). I treated this example, and some related examples, in an article in the Brown foraging conference volume, *Foraging Behavior* (Green 1987a), and in a series of technical reports (Green 1987b, c, d, 1988a, b). A version of the example described in this paper was treated earlier in an unpublished paper on central-place foraging (Green, Nuñez and Kacelnik 1983), but this earlier treatment added the rather ugly restriction that foragers were required to find at least one prey in each patch. The treatment in this paper will follow that given in Green (1980, 1984) for the very similar beta-binomial case.

## THE MODEL

The example that I treat in this paper is a particular, simple case of Oaten's (1977) general stochastic model of optimal foraging. I will describe my assumptions in a slightly different way than Oaten does his.

1. Foragers search for prey in patches which are superficially similar and are of the same size, each consisting of exactly  $n$  locations where prey may be found, and each location is searched in unit time. Patches are not revisited and the prey distribution is assumed not to change over time.
2. There are two types of patches, "good" and "bad." Within a patch of a given type the number of prey found in each location searched will be either zero or one, with probabilities  $1 - p$  and  $p$ , respectively. Within a patch the numbers of prey in different locations will be independent of each other. That is, within a patch the numbers of prey in different locations will be Bernoulli variables, with probabilities of finding prey in a given location being  $p_1$  for bad patches and  $p_2$  for good patches, with  $p_1 < p_2$ .
3. A patch may be left at any time after searching a location,  $t = 1, 2, \dots, n$ . Any patch searched completely will be left after searching the final bit, that is,  $t = n$ . The travel time to the next patch is  $\tau$ .
4. For a given patch-leaving rule the payoff for the forager is the long-term average rate of finding prey, given by

$$R = \frac{EG}{ET + \tau}, \quad (1)$$

where  $EG$  and  $ET$  are the expected number of prey found and the expected number of locations searched (the time spent searching) in each patch, respectively.

5. The optimal strategy is the one which maximizes  $R$ .

## FINDING THE OPTIMAL FORAGING STRATEGY

The optimal foraging strategy, that is, the patch-leaving rule which maximizes the long-term average rate of finding prey,  $R$ , is found by dynamic programming, using the same method used for a similar problem (Green 1980). There are several ideas on which the method is based. The main idea is that a forager should leave patches which seem bad and should remain in patches which seem good. The problem is how to decide whether patches are good or bad. We take advantage of the following facts:

1. Since it follows from my assumptions that prey are found at random within a patch, we have that if  $x$  prey have been found at time  $t$  in a patch (after  $t$  locations have been searched) then the number of prey found is sufficient for estimating patch quality. That is, the number of prey found by a particular time in a patch is all that matters; the times at which the prey were found are irrelevant. The problem in determining the optimal patch-leaving rule is to find points  $(t, x)$  such that the forager should leave a patch if exactly  $x$  prey have been found after the  $t$ th location has been searched. We might characterize the problem as one of partitioning all points of the form  $(t, x)$  into two sets: "continuation points," from which a forager should remain in the patch and search at least one more location, and "stopping points," from which the forager should leave a patch (or before which the forager should have left the patch) and go on to another patch.
2. Since there are a finite number of possible stopping points there are a finite number of possible non-randomized patch-leaving rules and one (possibly more) of these must achieve the highest possible rate  $R$ , which I call  $C^*$ .
3. At any point in a patch a forager can decide to leave and go on to the next patch. From the time at which a patch is left an optimal forager will find prey at a long-term rate of  $R = C^*$ . Therefore, the decision whether or not to remain in a patch at any point  $(t, x)$  will depend on whether or not the forager can, in a sense, find prey at a rate greater than  $C^*$  *from that point on, while remaining in the patch.*
4. The decision whether to remain in a patch at any time depends on the number of prey that have been found by that time. The decision may be interpreted in terms of the forager's opinion about the quality of the patch. For the model that I am considering, this opinion may be expressed in terms of the probability (or the odds) that a patch is "good," given that  $x$  prey have been found by time  $t$ . If we write this probability as  $p_2(t, x)$  then we have

$$p_2(t, x) = \frac{(1 - \alpha)p_2^x q_2^{t-x}}{\alpha p_1^x q_1^{t-x} + (1 - \alpha)p_2^x q_2^{t-x}}, \quad (2)$$

where  $q_1 = 1 - p_1$  and  $q_2 = 1 - p_2$ . [The probability that the patch is bad, given by  $p_1(t, x)$ , is calculated analogously.] The probability  $p_2(t, x)$  is decreasing in  $t$  and increasing in  $x$ . As a consequence of this, patches should be left when  $t$  is large or  $x$

is small, and the set of stopping points will be a "corner set," having the property that if  $(t_0, x_0)$  is a stopping point then  $(t_0, x_1)$  will be a stopping point if  $x_1 < x_0$  and  $(t_1, x_0)$  will be a stopping point if  $t_1 > t_0$ .

The optimal patch-leaving rule is found by dynamic programming. First, a value  $C$  is guessed for the maximum long-term average rate of finding prey that a forager might achieve. Then a rule is found that "tries to achieve" rate  $C$ . Then the rate,  $R(C)$ , actually achieved by this rule is found. If  $R(C) = C$ , the best rule has been found. If  $R(C) \neq C$ , then another value of  $C$  is guessed [ $R(C)$  for the previous  $C$  is a good next guess for the value of  $C$ ] and the process is continued until  $R(C) = C$ . It should be noted that a rule found that "tries to achieve" rate  $C$  may not give the optimal strategy for the values of  $\alpha$ ,  $p_1$ ,  $p_2$  and  $\tau$  originally specified, but unless the value guessed for  $C$  is too large or too small, the rule found will be best for the original choice of  $\alpha$ ,  $p_1$ ,  $p_2$  and some other choice of  $\tau$ .

*Finding the rule that "tries to achieve" rate  $C$ .*

Once a value of  $C$  is guessed for the highest possible long-term average rate of finding prey, the rule that "tries to achieve" this rate is found. This rule consists of a series of points  $(t_k, k)$  for  $k = 0, 1, 2, \dots, m$ . A forager should leave a patch at time  $t_k$  if exactly  $k$  prey have been found by that time. If no such point is reached, then the forager should search the patch completely, leaving at time  $t = n$ , having found  $x > m$  prey. The stopping points are found by backward recurrence, starting at the end of a patch and working backward toward the beginning. I will describe how this is done in four steps:

- a) First, the value of  $m$  is found, where  $m$  is the largest number of prey that could be found in a patch and still have the forager leave the patch before searching it completely.
- b) Then we find  $t_m$ , the time (the number of locations searched) at which a forager should leave the patch if exactly  $m$  prey have been found.
- c) Then  $t_{m-1}$  is found.
- d) Then  $t_{m-2}, t_{m-3}, \dots, t_0$  are found. Steps b) - d) are treated separately because finding them involves considering different kinds of points at which a forager might stop if it does not stop at point  $(t, k)$ .

In each case I will decide whether candidate stopping points  $(t, x)$  are really continuation points by asking: If a forager, finding itself at a point  $(t, x)$ , decides to remain in the patch long enough to search one more location, to leave the patch if a prey is not found at that location and to remain in the patch until another stopping point is

reached if a prey is found in the next location, is the ratio

$$\frac{EG'(t, x, C)}{ET'(t, x, C)} > C, \quad (3)$$

where  $EG'(t, x, C)$  is the expected number of prey that would be found by a forager that decides to continue and  $ET'(t, x, C)$  is the expected number of locations the forager would search? If the  $EG'/ET'$  ratio at a point exceeds  $C$  then the forager should remain in the patch and search another location. The rule developed in this way by asking what to do at point  $(t, x)$  depends on what decisions will be made at subsequent points, but unless  $t = n - 1$  these decisions, which depend on the choice of  $C$ , will have already been specified. The values of  $EG'(t, x, C)$  and  $ET'(t, x, C)$  depend on the values of  $t, x$  and  $C$ , but to simplify the notation I will suppress this dependence and write  $EG'$  and  $ET'$ , when the simplification can be made without ambiguity.

Now I will look at steps a) - d) in detail.

a) The number  $m$  is found by assuming that a forager has searched all but one location in a patch, that is,  $t = n - 1$ , and by asking whether to search the last location. If the forager has found  $x$  prey by time  $t = n - 1$  then the probability that the patch is good ( $p = p_2$ ) will be given by  $p_2(n - 1, x)$ , while the probability that the patch is bad is given by  $p_1(n - 1, x)$ . If such a forager decides to search the final location, the probability of finding a prey there will be

$$r(n - 1, x) = p_1(n - 1, x)p_1 + p_2(n - 1, x)p_2. \quad (4)$$

Then the value of  $m$  will be the largest integer  $x$  such that

$$r(n - 1, x) \leq C. \quad (5)$$

b) Once  $m$  has been found we know that  $(n - 1, m)$  is a stopping point. The question is, what is  $t_m$ ? That is, what is the smallest integer  $t$  such that  $(t, m)$  is a stopping point? Starting with  $t = n - 2$ , and continuing by taking smaller values we ask what would happen if at time  $t$ , having found  $m$  prey, the forager decides to remain in the patch to search one more location, to leave the patch if no prey is found in the next location, and to remain until the end of the patch if a prey is found. If  $ET'$  and  $EG'$  are the expected time and the expected number of prey found in the remainder of a patch using this rule then

$$ET' = 1 + r(t, m)(n - t - 1), \quad (6)$$

and

$$EG' = r(t, m)[1 + p_1(t + 1, m + 1)(n - t - 1)p_1 + p_2(t + 1, m + 1)(n - t - 1)p_2]. \quad (7)$$

The value of  $t_m$  will be the smallest integer  $t$  such that

$$\frac{EG'}{ET'} \leq C. \quad (8)$$

c) Once  $t_m$  has been found the next step is to find  $t_{m-1}$ . Clearly,  $t_{m-1} < t_m - 1$  since if  $(t_m, m)$  is a stopping point then  $EG'(t_m, m, C)/ET'(t_m, m, C) \leq C \Rightarrow r(t_m, m) \leq C$ , and for the point  $(t_m - 1, m - 1)$  we have  $EG'(t_m - 1, m - 1, C)/ET'(t_m - 1, m - 1, C) = r(t_m - 1, m - 1) \leq C$ . Therefore, we consider candidate values,  $t$ , for  $t_{m-1}$ , beginning with  $t_m - 2$  and working backward. The required value for  $t_{m-1}$  is the smallest value of  $t$  such that

$$\frac{EG'(t, m - 1, C)}{ET'(t, m - 1, C)} \leq C \quad (9)$$

In order to determine whether a point  $(t, m - 1)$  is a stopping point, after having determined that  $(t + 1, m - 1)$  is a stopping point, we must calculate  $ET'$  and  $EG'$  by assuming that a forager at point  $(t, m - 1)$  will remain in the patch at least long enough to search one more location. If no prey is found at time  $t + 1$  the forager will leave the patch, having spent one unit more time in the patch and having found no prey. If a prey is found at time  $t + 1$  then the forager will have found  $m$  prey and should remain in the patch until time  $t_m$  if no further prey are found. If another prey is found between time  $t + 1$  and time  $t_m$  then the forager should remain in the patch until it has been searched completely. In order to calculate  $ET'$  and  $EG'$  we ask:

- (i) **Q:** What is the probability that a prey is found at time  $t + 1$ , given that  $m - 1$  prey have been found by time  $t$ ? **A:**  $r(t, m - 1)$ .
- (ii) **Q:** What is the probability that a patch is "good" ("bad") given that  $m$  prey have been found by time  $t + 1$ ? **A:**  $p_2(t + 1, m)$  [or,  $p_1(t + 1, m)$ ].
- (iii) **Q:** If a patch is "good" (or "bad") what is the probability that no additional prey will be found from time  $t + 1$  to time  $t_m$ ? **A:**  $q_2^{t_m - t - 1}$  [or  $q_1^{t_m - t - 1}$ ].
- (iv) **Q:** If a patch is "good" [or "bad"] and if  $m$  prey have been found by time  $t_m$ , what is the expected number of prey that remain in the patch (that will be missed by a forager that leaves the patch at time  $t_m$ )? **A:**  $(n - t_m)p_2$  [or  $(n - t_m)p_1$ ].

Using the ideas in (i) - (iv) we have

$$ET' = 1 + r(t, m - 1) \left\{ \sum_{i=1}^2 p_i(t + 1, m) [(n - t - 1) - q_i^{t_m - t - 1} (n - t_m)] \right\} \quad (10)$$

$$EG' = r(t, m-1) \left\{ 1 + \sum_{i=1}^2 p_i(t+1, m) [(n-t-1)p_i - q_i^{t_m-t-1} (n-t_m)p_i] \right\} \quad (11)$$

d) Once  $m, t_m$  and  $t_{m-1}$  have been found for a given value of  $C$ , we continue to find  $t_{m-2}, t_{m-3}, \dots, t_0$ . These cases are treated together because finding them all requires the same ideas, but they are treated separately from  $t_m$  and  $t_{m-1}$  because they require a new idea to overcome the difficulty that a forager, having found  $k+1$  prey at time  $t+1$ , where  $k+1 < m$ , may stop at time  $t_{k+1}$ , having found  $k+1$  prey, or at time  $t_{k+2}$ , having found  $k+2$  prey, and so on, up to time  $t_m$ , when a forager should leave the patch if exactly  $m$  prey have been found. The added difficulty is finding the probabilities of reaching the  $(t_j, j)$  points, for  $j = k+1, k+2, \dots, m$ , from the point  $(t+1, k+1)$ , given that a patch is "good" [or "bad"]. For  $j = k+1$ , formulas like those given in (iii) suffice, but for  $j > k+1$  we must consider the number of ways of reaching later stopping points which requires that no earlier stopping point was reached.

The number of paths from  $(t+1, k+1)$  to the stopping point  $(t_j, j)$ , avoiding intermediate stopping points, will be denoted by  $w(j, t+1, k+1)$ , which I will write as  $w(j)$  when the dependence on  $t$  and  $k$  can be suppressed without ambiguity. We have

$$w(k+1) = 1 \quad (12)$$

$$w(j) = \binom{t_j - t - 1}{j - k - 1} - \sum_{i=k+1}^{j-1} w(i) \binom{t_j - t_i}{j - i} \quad (13)$$

for  $j = k+2, k+3, \dots, m$ . The values of  $w(j)$  are used to find  $ET'$  and  $EG'$ . We have

$$ET' = 1 + r(t, k) \{ (n-t-1) \quad (14)$$

$$- p_1(t+1, k+1) \sum_{j=k+1}^m w(j) p_1^{j-k-1} q_1^{t_j-t-j+k} (n-t_j) \\ - p_2(t+1, k+1) \sum_{j=k+1}^m w(j) p_2^{j-k-1} q_2^{t_j-t-j+k} (n-t_j) \}$$

$$EG' = r(t, k) \{ 1 + p_1(t+1, k+1) [(n-t-1)p_1 \\ - \sum_{j=k+1}^m w(j) p_1^{j-k-1} q_1^{t_j-t-j+k} (n-t_j)p_1] \\ + p_2(t+1, k+1) [(n-t-1)p_2 \\ - \sum_{j=k+1}^m w(j) p_2^{j-k-1} q_2^{t_j-t-j+k} (n-t_j)p_2] \}. \quad (15)$$

The desired value  $t_k$  is the smallest integer  $t$  such that  $EG'/ET' \leq C$ .

*Finding the rate  $R$ , given the strategy.*

Once the stopping times  $t_0, t_1, \dots, t_m$  are found for a guess  $C$  for the maximum rate, we find the rate  $R(C)$  that this strategy actually achieves. We begin by finding the number of ways,  $w(j, 0, 0)$ , of reaching the stopping points  $(t_j, j)$  starting from the origin. We have

$$w(0) = 1, \quad (16)$$

and, for  $j = 1, 2, \dots, m$

$$w(j) = \binom{t_j}{j} - \sum_{i=0}^{j-1} w(i) \binom{t_j - t_i}{j - i} \quad (17)$$

Then the  $w(j)$  values are used to find  $ET$  and  $EG$ ,

$$ET = \alpha \left[ n - \sum_{j=0}^m w(j) p_1^j q_1^{t_j - j} (n - t_j) \right] \quad (18)$$

$$+ (1 - \alpha) \left[ n - \sum_{j=0}^m w(j) p_2^j q_2^{t_j - j} (n - t_j) \right]$$

$$EG = \alpha \left[ np_1 - \sum_{j=0}^m w(j) p_1^j q_1^{t_j - j} (n - t_j) p_1 \right] \quad (19)$$

$$+ (1 - \alpha) \left[ np_2 - \sum_{j=0}^m w(j) p_2^j q_2^{t_j - j} (n - t_j) p_2 \right].$$

and

$$R(C) = \frac{EG}{ET + \tau}. \quad (1)$$

An annotated BASIC program that performs these calculations for the case  $n = 20$ ,  $\alpha = 0.5$ ,  $p_1 = 0.1$  and  $p_2 = 0.4$  is given in Appendix 1.

## OTHER STOPPING RULES

A number of other stopping rules have been suggested for problems like the one that I describe in this paper. [See Krebs, Ryan and Charnov (1974).] I will discuss three suboptimal rules which I have treated for a similar model (Green 1980, 1984). These rules are: (i) the "naive" rule [sometimes called the "fixed-time" rule (Breck 1978), or the "time expectation" rule (Krebs *et al.* 1974)]—ignore experience in a patch and search every patch completely; (ii) the "odds" rule [a version of what I have called the "instantaneous rate" rule (Green 1980), examples of which have mistakenly been treated as optimal (Pyke 1978; Iwasa, Higashi and Yamamura 1981)]—leave a patch when the odds

that the patch is good fall to some threshold, and (iii) the “giving-up-time” rule [the “GUT” rule, which also has been mistakenly treated as optimal (Krebs *et al.* 1974)]—leave a patch whenever some critical time  $T$  [the “giving-up-time”] has passed in the patch without finding a prey.

(i) *The “naive” rule.*

The long-term average rate of finding prey is easy to find for this case since each patch is searched completely, and all the prey are found in each patch searched. We have

$$ET = n \quad (20)$$

$$EG = n[\alpha p_1 + (1 - \alpha)p_2] \quad (21)$$

and

$$R = \frac{n[\alpha p_1 + (1 - \alpha)p_2]}{n + \tau} \quad (22)$$

(ii) *The “odds” rule.*

If  $x$  prey have been found by time  $t$  then the odds that the patch is good are given by

$$\begin{aligned} Odds &= \frac{p_2(t, x)}{p_1(t, x)} = \frac{\alpha p_1^x q_1^{t-x}}{(1 - \alpha) p_2^x q_2^{t-x}} \\ &= \frac{\alpha}{1 - \alpha} \left( \frac{q_1}{q_2} \right)^t \left( \frac{p_1 q_2}{p_2 q_1} \right)^x \end{aligned} \quad (23)$$

If we set the odds equal to a constant and take logs we have

$$\begin{aligned} \log(Odds) &= \log(\alpha) - \log(1 - \alpha) \\ &\quad + t[\log q_1 - \log q_2] + x[\log(p_1 q_2) - \log(p_2 q_1)] \\ &= Const. \end{aligned} \quad (24)$$

which is equivalent to saying that  $t$  is a linear function of  $x$ . The odds rule says that for any number  $x$  of prey found, leave the patch at the smallest time  $t$  lying on or below the line given in (24). For any choice of the odds we can use (24) to find points  $t_0, t_1, \dots, t_m$ , and we can calculate  $ET$ ,  $EG$  and  $R$  using the same technique used above to find these values after a candidate optimal rule has been found that “tries to achieve” some guessed rate  $C$ .

(iii) *The giving-up-time rule.*

The main difficulty in determining the performance of the giving-up-time rule for a particular choice of the giving-up-time  $T$  is to find the number of ways that a forager

could get to a point  $(t, x)$ , having found no prey in the last  $T$  locations, having found a prey in the last location searched before these  $T$  failures (that is, at time  $t - T$ ), and not having searched  $T$  consecutive locations unsuccessfully before finding the  $x$ th prey at time  $t - T$ . The required numbers of ways are found by iteration, beginning with

$$w(T, 0) = 1, \quad (25)$$

and

$$\begin{aligned} w(t, 0) &= 0 && \text{for } t \neq T, \\ w(T + i, 1) &= 1 && \text{for } i = 1, 2, \dots, T, \\ &= 0 && \text{otherwise.} \end{aligned} \quad (26)$$

In order to find the values for  $w(t, x)$  for  $x \geq 2$  and  $T + x \leq t \leq n - 1$  we iterate for  $x = 2, 3, \dots$

$$w(t, x) = \sum_{i=1}^T w(t - i, x - 1). \quad (27)$$

The expression on the left of (27) refers to cases in which one prey is found at time  $t - T$ , none are found after that time through time  $t$ , and  $x - 1$  prey are found before time  $t - T$ , with no time interval of length  $T$  yielding no prey. We might ask when the last of these  $x - 1$  prey is found. The number of ways this prey can occur at time  $t - T - 1$  is the same as the number of ways of stopping  $T$  time units later [that is, at time  $t - 1$ ], having found  $x - 1$  prey. This number is  $w(t - 1, x - 1)$ . The number of ways of having the  $x - 1$ st prey at location  $t - T - i$  is  $w(t - i, x - 1)$ . Thus we have the right-hand side of (27).

The case in which  $t = n$  is different from the others because at this time a patch has been searched completely and the forager should leave no matter how long it has been since the last prey was found. The values for  $w(n, x)$  are found by extending the calculations given in (27) for  $t = n, n + 1, \dots, n + T$ . Call the new numbers of ways,  $v(t, x)$ . Then

$$w(n, x) = \sum_{t=n}^{n+T} v(t, x), \quad \text{for } x = 0, 1, 2, \dots, n. \quad (28)$$

Once we find the numbers of ways of reaching the points  $(t, x)$  in such a way that the patch must be left we can write the probabilities of reaching these points

$$p(t, x) = w(t, x)[\alpha p_1^x q_1^{t-x} + (1 - \alpha) p_2^x q_2^{t-x}] \quad (29)$$

and we can use these probabilities to calculate  $ET$  and  $EG$  :

$$ET = \sum_t \sum_x t p(t, x), \quad (30)$$

$$EG = \sum_t \sum_x x p(t, x). \quad (31)$$

An annotated BASIC program that performs these calculations is given in Appendix 2.

## RESULTS

The best rule for the case  $\alpha = 0.5$ ,  $p_1 = 0.1$ ,  $p_2 = 0.4$ , and  $\tau = 1$  is  $t_0 = 2$ ,  $t_1 = 6$ ,  $t_2 = 10$ ,  $t_3 = 14$ ,  $t_4 = 18$ , with  $m = 4$ . For this rule,  $EG = 2.627514$ ,  $ET = 7.97344$ , and  $R = 0.2988039$ . The best rule is very similar to the best odds rule, which has stopping times  $t_0 = 2$ ,  $t_1 = 6$ ,  $t_2 = 10$ ,  $t_3 = 15$ ,  $t_4 = 19$ . The odds leading to this rule are 13:20, but these odds are not unique. [There is an interval of values of the odds (including 13:20) that lead to the same rule.] For the best odds rule,  $EG = 2.638142$ ,  $ET = 7.831994$ , and  $R = 0.2987029$ . The rate for this rule differs from that achieved by the best rule by less than 0.01%.

The best giving-up-time rule, for which the giving-up time is 5, yields values  $EG = 3.503759$ ,  $ET = 11.35003$ , and  $R = 0.2837045$ . This rate is only about 5% less than the rate for the best rule. The rates for all the rules may be compared with the rate achieved by the "naive" rule, which ignores experience in patches. This rate is  $R = 0.238095$ , which is about 83% of the rate for the best rule. There seems to be some advantage in using information obtained while foraging to decide when to leave a patch. The rates achieved by the four types of rules are given in Table 1 for various values of travel time,  $\tau$ . The same prey distribution is used in each case:  $\alpha = 0.5$ ,  $p_1 = 0.1$  and  $p_2 = 0.4$ . Values of  $ET$ ,  $EG$  and  $R$  are given in Table 2 for the giving-up-time rule for all possible values of the giving-up time for the same prey distribution.

In Figure 1 the values of  $EG$  are plotted against  $ET$  for a number of candidate optimal patch-leaving rules (most of which are best for *some* travel time), and for giving-up-time rules for all possible giving-up times.

## LITERATURE CITED

- Breck, J. E. 1978. Suboptimal foraging strategies for a patchy environment. Ph. D. dissertation, Michigan State University.
- Charnov, E. L. 1976. Optimal foraging: the marginal value theorem. *Theoretical Population Biology*, 9:129-136.
- Green, R. F. 1980. Bayesian birds: a simple example of Oaten's stochastic model of optimal foraging. *Theoretical Population Biology*, 18:244-256.
- Green, R. F. 1981. Optimal foraging and functional response. (Unpublished manuscript)
- Green, R. F. 1984. Stopping rules for optimal foragers. *American Naturalist*, 123:30-43.

- Green, R. F. 1987a. Stochastic models of optimal foraging. In: *Foraging Behavior*, (ed. by A. C. Kamil, J. R. Krebs, and H. R. Pulliam), pp. 273-302. Plenum Press, New York.
- Green, R. F. 1987b. A stochastic model of optimal foraging: systematic search for negative-binomially distributed prey. Computer Science, Mathematics and Statistics, University of Minnesota, Duluth, Technical Report No. 87-2.
- Green, R. F. 1987c. The giving-up-time rule as a strategy for animals foraging systematically in a patchy environment. Computer Science, Mathematics and Statistics, University of Minnesota, Duluth, Technical Report No. 87-3.
- Green, R. F. 1987d. Optimal foraging in patches, each of which contains the same number of prey. Computer Science, Mathematics and Statistics, University of Minnesota, Duluth, Technical Report No. 87-4.
- Green, R. F. 1988a. Optimal foraging and pessimal foraging, or the optimal avoidance of predators. Computer Science, Mathematics and Statistics, University of Minnesota, Duluth, Technical Report No. 88-1.
- Green, R. F. 1988b. Optimal foraging for patchily distributed prey: random search. Computer Science, Mathematics and Statistics, Technical Report No. 88-2.
- Green, R. F., Nuñez, A. T., and Kacelnik, A. 1983. Central-place foraging in a stochastic environment: a multiple-prey loader. (Unpublished manuscript)
- Iwasa, Y., Higashi, M., and Yamamura, N. 1981. Prey distribution as a factor determining the choice of optimal foraging strategy. *American Naturalist*, **117**:710-723.
- Krebs, J. R., Ryan, J. C., and Charnov, E. L. 1974. Hunting by expectation or optimal foraging? A study of patch use by chickadees. *Animal Behaviour*, **22**:953-964.
- Oaten, A. 1977. Optimal foraging in patches: a case for stochasticity. *Theoretical Population Biology*, **12**:263-285.

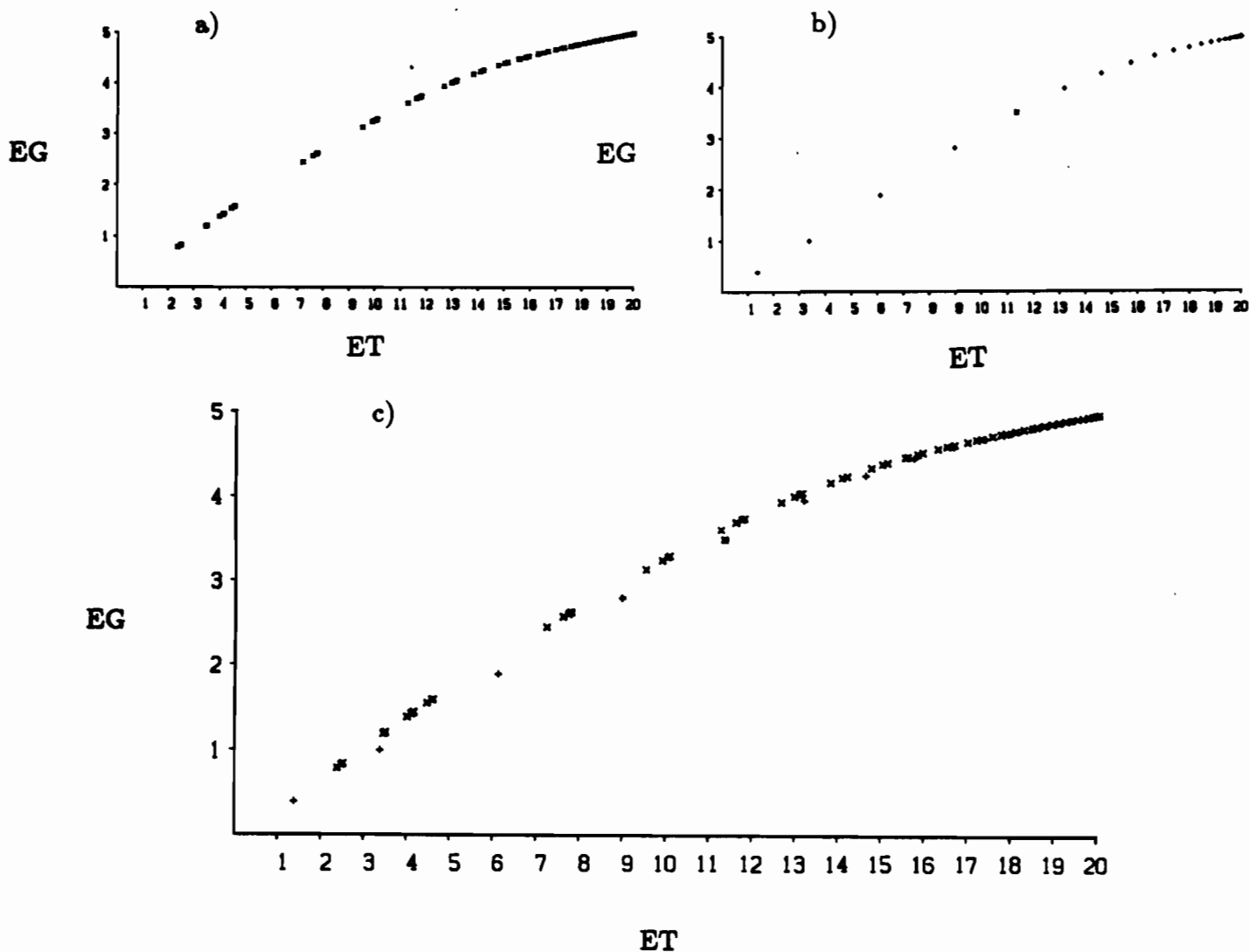
**TABLE 1.** Long-term average rates of finding prey achieved by the best rules of each of four types, for the same prey distribution:  $\alpha = 0.5$ ,  $p_1 = 0.1$ ,  $p_2 = 0.4$  and  $n = 20$ , and several values of  $\tau$ , the travel time between patches. The “Best” Rule is the rule that achieves the highest possible rate of finding prey among all rules that rely on information gained while searching patches. For the Odds Rule and the GUT Rule values of the odds and the GUT are given that yield the rates given in the table. In the “Naive” Rule experience in patches is ignored and each patch is searched completely.

Travel time, $\tau$	Rates achieved by the following rules:			
	Best	Odds	GUT	Naive
0	.34656	.34652 (1.7)	.31243 (4)	.25000
1	.29880	.29870 (0.65)	.28370 (5)	.23810
2	.27344	.27328 (0.42)	.26245 (5)	.22727
3	.25373	.25354 (0.28)	.24546 (6)	.21739
4	.23764	.23754 (0.28)	.23117 (6)	.20833
5	.22390	.22372 (0.19)	.21846 (6)	.20000
10	.17638	.17634 (0.13)	.17396 (8)	.16667
20	.12653	.12650 (0.035)	.12607 (10)	.12500

**TABLE 2.** Performance of the giving-up-time rule for various values of the giving-up time (GUT). The prey distribution is a two-point mixture of binomials, with  $\alpha = 0.5$ ,  $p_1 = 0.1$ ,  $p_2 = 0.4$ ,  $n = 20$ , and  $\tau = 1$ .

GUT	EG	ET	R
1	0.38889	1.38889	0.16279
2	1.00450	3.39088	0.22877
3	1.89129	6.12224	0.26555
4	2.80418	8.97529	0.28111
5	3.50376	11.35003	0.28370
6	3.97212	13.18255	0.28007
7	4.27600	14.60511	0.27401
8	4.47677	15.73389	0.26753
9	4.61462	16.64502	0.26153
10	4.71335	17.38743	0.25634
11	4.78532	17.98054	0.25212
12	4.83913	18.45398	0.24875
13	4.88032	18.83339	0.24607
14	4.91233	19.13741	0.24394
15	4.93739	19.38021	0.24226
16	4.95702	19.57290	0.24095
17	4.97234	19.72447	0.23993
18	4.98419	19.84227	0.23914
19	4.99323	19.93242	0.23854
20	5.00000	20.00000	0.23810

**FIGURE 1.** Expected number of prey found,  $EG$ , plotted against expected time spent in a patch,  $ET$ , a) for various candidate Best Rules, b) for all possible Giving-up-time Rules, and c) for both types of rules plotted together. Points for candidate Best Rules are marked with x's, while points for Giving-up-time Rules are marked with plus signs. The points marked with asterisks are for the best rules of the two types when travel time between patches is  $\tau = 1$ . These points satisfy a kind of "marginal value theorem" which could be illustrated by extending the  $ET$ -axis to the left from the origin and drawing an upper tangent line from the point  $(-\tau, 0)$  to the envelope generated by the  $(ET, EG)$  points for rules of a particular type. All calculations are done for the case:  $\alpha = 0.5, p_1 = 0.1, p_2 = 0.4$  and  $n = 20$ . Notice that for some candidate Best Rules (for  $C \geq 0.347$ ) there is no travel time such that these rules are optimal. That is, a tangent line to the envelope determined by the set of  $(ET, EG)$  points for candidate Best Rules which passes through a point corresponding to a rule which tries to achieve a rate greater than or equal to 0.347 will hit the  $ET$ -axis to the right of the origin, suggesting a negative travel time between patches.



## APPENDIX 1

This BASIC program finds a candidate best foraging strategy and calculates the long-term average rate that it achieves for a model in which the prey distribution is a mixture of two binomials. The proportion of "bad" patches,  $\alpha$ , is given as  $A$  in line 10. The success rate in "bad" patches,  $p_1$ , is given in line 20, while the success rate in "good" patches,  $p_2$ , is given in line 30. The number of locations per patch,  $n$ , is given in line 40, and the travel time between patches,  $\tau$ , is given as  $T1$  in line 50. The value  $C$ , guessed for the highest rate that can be achieved, is given in line 60.

Binomial coefficients,  $C(I, J)$ , are calculated in lines 150-240. The probability that a patch is bad, given that  $J$  prey have been found by time  $I$ , denoted by  $p_1(i, j)$  in the text, is calculated in lines 300-390, and is given as  $P5(I, J)$  in line 370. The formula for  $p_2(i, j)$  is given in expression (2) in the text.

The value of  $m$  is found in lines 400-450. The stopping time  $t_m$  is found in lines 500-580, and the result is printed in line 590.  $ET'$  and  $EG'$  are found in lines 540 and 550, respectively. Lines 520 and 530 give  $r(t, m)$  and  $r(t + 1, m + 1)$ , denoted as  $R1$  and  $R2$ , respectively. Stopping time  $t_{m-1}$  is found in lines 600 to 700, and the result is printed in line 710.  $ET'$  and  $EG'$ , denoted by  $S1$  and  $G1$ , respectively, are calculated in lines 660 and 670, while values for  $r(t, m - 1)$ ,  $r(t + 1, m)$  and  $r(t_m, m)$  are denoted by  $R1$ ,  $R2$ , and  $R3$ , respectively, and are calculated in lines 620, 630 and 640.

Stopping times  $t_{m-2}, t_{m-3}, \dots, t_0$  are found in lines 800 to 1020, with the results printed in line 1010. The number of ways of getting from the point  $(t + 1, k + 1)$  to the point  $(t_{k+1+i}, i)$  without hitting intervening stopping points, denoted as  $W0(I)$ , is calculated in lines 890 to 940. Formula (13) in the text is calculated in line 930.  $ET'$  and  $EG'$  are calculated, beginning in lines 850 and 860, respectively, continuing in lines 870 and 880, respectively, and finishing in lines 950 and 960, respectively. Lines 884-885, 944-945 and 974-975 are now-avoided checks of the calculations.

Once the value  $m$  and the times  $t_0, t_1, \dots, t_m$  have been found for a given choice of  $C$ , the rate,  $R(C)$ , achieved by the rule specified by these values, is calculated in lines 1100 to 1200, and the results are printed in line 1300. The number of ways of reaching stopping point  $(t_k, k)$ , and avoiding intervening stopping points, is calculated in lines 1120 to 1170, with formula (17) in the text calculated in line 1160.  $EG$  and  $ET$ , de-

noted by  $E1$  and  $E2$ , respectively, are calculated beginning in lines 1100 and 1110, respectively, and finishing in lines 1180 and 1190, respectively.

```

5 PRINT "twopt"
10 A = .5
20 P1 = .1
30 P2 = .4
40 N = 20
50 T1 = 1
55 LPRINT "a =" ; A ; "p1 =" ; P1 ; "p2 =" ; P2 ; "tau =" ; T1
60 C = .35
65 LPRINT "C =" ; C
70 DIM C(N,N)
80 DIM P5(N,N)
90 DIM T(N)
100 DIM W(N)
110 DIM W0(N)
120 Q1 = 1 - P1
130 Q2 = 1 - P2
150 C(0,0) = 1
160 FOR I = 1 TO N
170 C(I,0) = 1
180 C(I,I) = 1
190 NEXT I
200 FOR I = 2 TO N
210 FOR J = 1 TO I-1
220 C(I,J) = C(I-1,J-1) + C(I-1,J)
230 NEXT J
240 NEXT I
300 FOR I = 1 TO N-1
310 P3 = A*(Q1^I)
320 P4 = (1 - A)*(Q2^I)
330 P5(I,0) = P3/(P3 + P4)
340 FOR J = 1 TO I
350 P3 = A*(P1^J)*(Q1^(I-J))
360 P4 = (1 - A)*(P2^J)*(Q2^(I-J))
370 P5(I,J) = P3/(P3 + P4)
380 NEXT J
390 NEXT I
400 FOR X = N-1 TO 0 STEP -1
410 M = X
420 R1 = P5(N-1,X)*P1 + (1 - P5(N-1,X))*P2
430 IF R1 <= C THEN GOTO 450
440 NEXT X
450 LPRINT "M =" ; M
500 T(M) = N-1
510 FOR T = N-2 TO M STEP -1
520 R1 = P5(T,M)*P1 + (1 - P5(T,M))*P2
530 R2 = P5(T+1,M+1)*P1 + (1 - P5(T+1,M+1))*P2
540 S1 = 1 + R1*(N - T - 1)
550 G1 = R1*(1 + R2*(N - T - 1))
560 IF G1/S1 > C THEN GOTO 590
570 T(M) = T
580 NEXT T
590 LPRINT "t (m) =" ; T(M)
600 T(M-1) = T(M) - 1

```

```

610 FOR T = T(M) - 2 TO M-1 STEP -1
620 R1 = P5(T,M-1)*P1 + (1 - P5(T,M-1))*P2
630 R2 = P5(T+1,M)*P1 + (1 - P5(T+1,M))*P2
640 R3 = P5(T(M),M)*P1 + (1 - P5(T(M),M))*P2
650 P0 = P5(T+1,M)*Q1^(T(M) - T - 1) + (1-P5(T+1,M))*Q2^(T(M) - T - 1)
660 S1 = 1 + R1*((N - T - 1) - P0*(N - T(M)))
670 G1 = R1*(1 + R2*(N - T - 1) - P0*R3*(N - T(M)))
680 IF G1/S1 > C THEN GOTO 710
690 T(M-1) = T
700 NEXT T
710 LPRINT "t(m-1) =";T(M-1)

800 FOR K = M-2 TO 0 STEP -1
810 T(K) = T(K+1) - 1
820 FOR T = T(K+1) - 2 TO K STEP -1
830 R1 = P5(T,K)*P1 + (1 - P5(T,K))*P2
840 R2 = P5(T+1,K+1)*P1 + (1 - P5(T+1,K+1))*P2
850 G1 = R1*(1 + R2*(N - T - 1))
860 S1 = 1 + R1*(N - T - 1)

870 G1 = G1 - R1*(P5(T+1,K+1)*(Q1^(T(K+1) - T - 1))*P1*(N - T(K+1))
+ (1 - P5(T+1,K+1))*(Q2^(T(K+1) - T - 1))*P2*(N - T(K+1)))

880 S1 = S1 - R1*(P5(T+1,K+1)*(Q1^(T(K+1) - T - 1))*(N - T(K+1))
+ (1 - P5(T+1,K+1))*(Q2^(T(K+1) - T - 1))*(N - T(K+1)))

884 GOTO 890
885 LPRINT "first";G1;S1
890 W0(0) = 1
900 FOR I = 1 TO M-K-1
910 W0(I) = C(T(K+1+I) - T - 1, I)
920 FOR J = 0 TO I - 1
930 W0(I) = W0(I) - W0(J)*C(T(K+1+I) - T(K+1+J), I-J)
940 NEXT J
944 GOTO 950
945 LPRINT I;W0(I)

950 S1 = S1 - R1*(P5(T+1,K+1)*W0(I)*(P1^I)*(Q1^(T(K+1+I) - I - T - 1))
*(N - T(K+1+I)) + (1 - P5(T+1,K+1))*W0(I)*(P2^I)*(Q2^(T(K+1+I) - I - T - 1))
*(N - T(K+1+I)))

960 G1 = G1 - R1*(P5(T+1,K+1)*W0(I)*(P1^I)*(Q1^(T(K+1+I) - I - T - 1))
*P1*(N - T(K+1+I))
+ (1 - P5(T+1,K+1))*W0(I)*(P2^I)*(Q2^(T(K+1+I) - I - T - 1))
*P2*(N - T(K+1+I)))

970 NEXT I
974 GOTO 980
975 LPRINT K;G1;S1
980 IF G1/S1 > C THEN GOTO 1010
990 T(K) = T
1000 NEXT T
1010 LPRINT K;"t(k) =";T(K)
1020 NEXT K

```

```

1100 E1 = A *(N*P1 - (Q1^T(0))*(N - T(0))*P1) + (1 - A)
      *(N*P2 - (Q2^T(0))*(N - T(0))*P2)

1110 E2 = A*(N - (Q1^T(0))*(N - T(0))) + (1 - A)*(N - (Q2^T(0))*(N - T(0)))

1120 W(0) = 1
1130 FOR I = 1 TO M
1140 W(I) = C(T(I), I)
1150 FOR J = 0 TO I-1
1160 W(I) = W(I) - W(J)*C(T(I) - T(J), I - J)
1170 NEXT J

1180 E1 = E1 - A*W(I)*(P1^I)*(Q1^(T(I) - I))*P1*(N - T(I))
      - (1 - A)*W(I)*(P2^I)*(Q2^(T(I) - I))*P2*(N - T(I))

1190 E2 = E2 - A*W(I)*(P1^I)*(Q1^(T(I) - I))*(N - T(I))
      - (1 - A)*W(I)*(P2^I)*(Q2^(T(I) - I))*(N - T(I))

1200 NEXT I
1300 LPRINT "EG =";E1;"ET =";E2;"Rate =";E1/(E2 + T1)

```

## APPENDIX 2

This BASIC program calculates the long-term average rate,  $R$ , achieved by a forager using the giving-up-time rule, with the value of the giving-up-time,  $T$ , denoted by  $G$  in line 100. The parameters of the model,  $\alpha$ ,  $p_1$ ,  $p_2$ ,  $\tau$  and  $n$ , denoted by  $A$ ,  $P1$ ,  $P2$ ,  $T1$  and  $N$ , respectively, are given in lines 20, 30, 40, 70 and 80, respectively. The number of ways,  $w(t, x)$ , of finding  $x$  prey by time  $t$  and then leaving a patch, having failed to find prey in the previous  $G$  locations searched, is calculated in lines 140 to 180 for  $G = 1$ , and in lines 200 to 440 for  $G > 1$ .  $W(G, 0) = 1$  [expression (25)] is given in line 200,  $W(G + I, 1) = 1$ , for  $I = 1, 2, \dots, G$  [expression (26)], is given in lines 210-230, while the other values for  $w(t, x)$ , and for  $v(t, x)$ , are calculated in lines 240 to 310, with (27) given in line 280. The numbers of ways,  $w(n, x)$ , are calculated in lines 400 to 440, with expression (28) given in line 420. Once the values of  $w(t, x)$  have been found, the values of  $EG$  and  $ET$ , given as  $E1$  and  $E2$  in lines 570 and 580, respectively, are calculated in lines 500 to 600, using the probabilities,  $p(t, x)$ , [expression (29)] of reaching the various possible patch-leaving points, which are found in line 560 of the program. The value chosen for the giving-up-time, and the resulting values of  $EG$ ,  $ET$  and  $R$  are printed in line 700.

```

10 PRINT "GUT rule for 2-pt. mixture of binomials"
20 A = .5
30 P1 = .1
40 P2 = .4
50 Q1 = 1 - P1
60 Q2 = 1 - P2
70 T1 = 1
80 N = 20
85 GOTO 100
90 LPRINT "p1 =";P1;"p2 =";P2;"Alpha =";A;"n =";N;"Tau =";T1
100 G = 11
110 DIM W(N+G,N+G)
120 DIM P(N,N)
130 IF G > 1 THEN GOTO 200
140 W(N,N) = 1
150 W(N,N-1) = 1
160 FOR I = 1 TO N-1
170 W(I,I-1) = 1
180 NEXT I
190 GOTO 500
200 W(G,0) = 1
210 FOR I = 1 TO G
220 W(G+I,1) = 1
230 NEXT I
240 FOR T = G+2 TO G+N
250 FOR X = 2 TO N
260 W(T,S) = 0
270 FOR I = 1 TO G
280 W(T,X) = W(T,X) + W(T-I,X-1)
290 NEXT I
300 NEXT X
310 NEXT T
400 FOR X = 0 TO N
410 FOR I = 1 TO G
420 W(N,X) = W(N,X) + W(N+I,X)
430 NEXT I
440 NEXT X
500 E1 = 0
510 E2 = 0
520 FOR T = G TO N
530 FOR X = 0 TO T
540 F1 = A*(P1^X)*(Q1^(T-X))
550 F2 = (1-A)*(P2^X)*(Q2^(T-X))
560 P(T,X) = W(T,X)*(F1 + F2)
570 E1 = E1 + X*P(T,X)
580 E2 = E2 + T*P(T,X)
590 NEXT X
600 NEXT T
700 LPRINT "GUT =";G;"EG =";E1;"ET =";E2;"R =";E1/(E2 + T1)

```