The Evolution of Information Gathering: Operational Constraints

Cynthia F. Kurtz

1991 Master's Thesis, SUNY Stony Brook, Ecology & Evolution

I present two new approaches to the study of information in foraging theory. First, rather than determine the cost a forager should pay to obtain information, I concentrate on the consequences of information use in an interacting population. I describe a density-dependent model which tracks genotypes with high and low information access through evolutionary time. Stable polymorphisms result. I suggest that the value of information is not monotonically increasing. Second, I present a scheme for partitioning the information used in the decision making process. Three types of information are recognized: internal information, or an individual's internal state; external information, or environmental factors; and relational information, or rules for predicting transformations of internal state. Interactions between the three types are examined in an extension of the basic model.

Introduction

Often the best answer to an old question is a new question. I examine two new questions about information in foraging theory.

According to Stephens and Krebs (1986), "Many authors view incomplete information as an unsolved problem in foraging theory," but "it is not unsolved because of a lack of effort." What makes it so difficult? In real situations, many types of information are woven into an intricate pattern. The brain's power to tease decisions out of pattern is not easy to represent with simple models. Consider an animal choosing between prey of two types. Type 1 is a model-mimic system with temporal variability in prey toxicity. Type 2 appears in ephemeral, shifting patches; thus it shows both temporal and spatial variability. Deciding whether to pursue an item of either of these prey types or to search for other prey involves a variety of information.

Also, the impact of information varies among situations, from a small energy loss to death from predation. It is difficult to generalize costs across currencies. These two problems, multiplicity of types and multiplicity of effects, have led to a "piecemeal case-by-case approach to information problems" (Stephens, 1989), and few clear generalizations.

Finally, information use by animals is not easy to observe. A few studies (Krebs et al. 1974, 1978, Orians 1981) have developed methods to observe information use, but the approach has not yet found enthusiastic experimenters.

I will address two common questions concerning foraging information. 1) What is the value of information? 2) How can problems of incomplete information be generalized across different situations?

What is the value of information?

The value of information was first defined in economic theory (Raiffa 1968, Gould 1974) and later applied to animal behavior (McNamara and Houston 1980, Stephens 1989). The argument is basically as follows. The environment can exist in one of many states. Each state is described by a reward function based on possible behaviors, and the function's maximum value defines the optimal behavior for that state. An optimally informed animal will choose the optimal behavior and receive the maximum reward. An animal with incomplete information will choose a behavior based on its inaccurate information. When its behavior misses the optimum for the environment's current state, it will receive a non-maximum reward. The value of information is the difference between rewards gained by the informed and uninformed animals.

Stephens (1989) suggested that the value of information in biological systems ought to be correlated with the variance in optimal behaviors. As viable options increase in number, information which helps to choose between them becomes more valuable. For example, consider a choice between three sites whose average energy outputs fluctuate around 1, 2, and 10 units. Gaining frequently updated information about site outputs will be of little value since the variance of the optimal behaviors is zero (the site with 10 units is always chosen). However, if the site energy outputs change to 1, 5, and 6 units, the same increase in information will be more valuable, because the variance in optimal behaviors is greater.

The above approaches calculate the cost a forager should pay to obtain information, and correlate this cost with information's value. This is a logical approach, since information cannot be free. But what would happen if one looked at information use from another perspective? I will consider not the cost, but the consequences of complete and incomplete information. In other words, if information were free, how much would one want? What is the value of free information? Are there factors involved in information use that might be obscured by concentrating on cost?

I present here a model which simulates the evolution of genes controlling cost-free information-gathering ability. Before describing the model, I will introduce a second question, which it also addresses. What is incomplete information?

Most studies of incomplete information concentrate on information about one of two types of uncertainty: temporal variability (e.g., fading of memory, McNamara & Houston 1987; sampling rules, Shettleworth et al. 1988; action under uncertain conditions, Mangel 1990); and spatial variability (e.g., sampling patches, Lima 1983; leaving patches, Green 1984; searching in patches, Stillman & Sutherland 1990). No matter where the uncertainty lies, foragers are assumed to lack needed information about some aspect of the environment: a prey item's type, a patch's food density, or a behavior's reward function. But is information about environmental parameters the only information needed to make decisions? What exactly is complete information? What does omniscience entail? What clear statements can be made about the state from which "incomplete information" departs?

To begin, let us examine the act of making a decision: selecting one behavior from the repertoire available. We assume that the omniscient animal chooses an action by predicting the effects of each possible behavior on its future success (ultimately fitness, but practically maximization of some quantity). Prediction requires that the animal create some representation of the real processes that determine the consequences of its behavior. The success of the animal's choice relies on how well its prediction represents reality.

What essential features does the representation have? Let us take a simplified look at the process of survival and reproduction. Each individual in a population at a given time has an internal state which largely determines its current reproductive potential. (See Mangel & Clark (1986) for a review of the state variable approach in foraging theory.) That state involves many factors, including stored energy, health, and fertility. The lifetime of any individual can be represented as a time series of internal state transformations. It is these transformations that the individual attempts to influence by its choice of behavior. However, other factors also influence the individual's future state, including biotic and abiotic environmental conditions and the actions of conspecifics. The interaction of these forces to produce changes in internal state is governed by a set of rules I will collectively call a transforming function. The transforming function reflects physical laws and does not change over ecological time.

To represent the transformation process, an omniscient individual needs three types of information. First, it knows its internal state: call this internal information. Second, it knows the pertinent environmental conditions: external information. Third, it knows the transforming function itself: relational information. All three information types join in the prediction, decision, and resulting behavior; and the consequences of that behavior become information used in the next prediction (Figure 1).
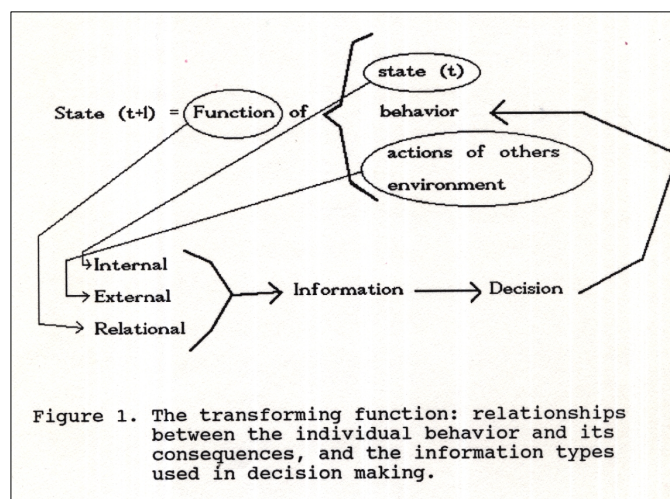
To clarify this scheme, picture a small bird foraging in winter. Suppose it must choose between two feeding sites. One contains energy-rich food but is frequently visited by a predator, and the second is of low energy content but is predator-free. What information does the bird use to choose a site? First, it senses that it is hungry. In fact, this bird has been operating on a low energy budget for some time, and it may starve tonight if it cannot gather enough energy for the night's heat production during the day. Also, the bird is aware of the resource quality and predation rate at each of the two sites, and that today the temperature is especially low. The amount of food a small bird must eat to survive the night is a specific function of ambient temperature — this determines how much is "enough." If the bird has incomplete knowledge of that function and underestimates its peril, it may decide to avoid predation by feeding at the safe site and starve as a result. Relational information is the framework into which the other information types fit to make the prediction accurate.

Figure 1. The transforming function: relationships between the individual behavior and its consequences, and the information types used in decision making.

Now consider foraging models with these three information types in mind. The classical models (e.g., Schoener 1971, Charnov 1976b) assume complete external and relational information. Since they are static models, they ignore internal state. Explicit incomplete information studies have dealt with deficiencies in external information only (as cited above). The dynamic optimization approach considers state-dependent decision making (Mangel & Clark 1986, 1988), but assumes complete information on internal state. Studies of rules of thumb (Janetos & Cole 1981, Green 1984) implicitly consider reductions in relational information, but assume complete internal and external information.

This approach combines internal, external and relational information in one model. It asks: how do the three types differ in their impact on decision making? What sorts of interactions occur?
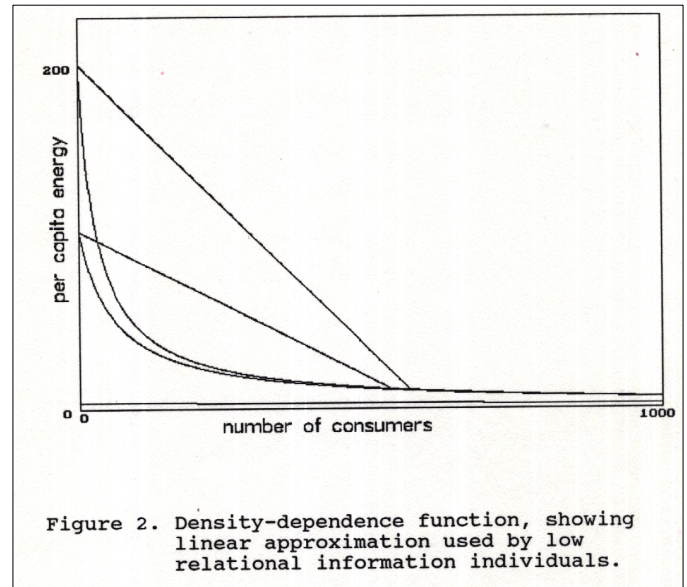
The cost-free information model mentioned above also studied the dynamics of internal, external and relational information. Three genes were monitored in the population, one for each type of information.

The Model

The model simulates a three-locus, haploid genetic system with one gene for each type of information: internal, external and relational. Each locus has one high- and one low-information allele. (These will be abbreviated in the remainder of the paper as + and -). Since reproduction is asexual, there is no recombination between genotypes, and the population consists of eight strains (+++, ++-, +-+, -++, +--, -+-, --+, and ---). The frequency of high-information alleles is calculated from genotype counts made throughout the simulation.

Each individual has an energy supply which is increased by consumption and decreased by maintenance. Reproduction and death occur probabilistically, with probabilities dependent on energy states. Mutation does not occur.

The model environment is divided into three sites within which resources are evenly distributed. Each site is replenished in every time frame with a random amount of energy drawn from a normal distribution about that site's mean. Also, patch mean energies are randomly reassigned in every 25th time frame, so that for example the good patch may become the bad, the bad the medium, and so on. These two processes represent fluctuations in site quality that might be associated with daily and seasonal changes, respectively. A good site gives an individual feeding alone an average of 200 units of energy, a medium site 100 units, and a bad site 5 units (Figure 2).

Figure 2. Density-dependence function, showing linear approximation used by low relational information individuals.

To begin the simulation, three hundred individuals are placed randomly in the patches, and they begin to forage. Each individual's energy state at time t, $X(i,t)$, changes according to the following function:

$$X(i,t) = X(i,t-1) + \wedge X(j,t) - C(m) - C(r) - C(v)$$

where $\wedge X(j,t)$ is the amount of energy any individual consumes given its choice of feeding site j; $C(m)$ is a unit maintenance cost; $C(r)$ is a conditional reproduction cost; and $C(v)$ is a movement cost subtracted when an individual switches sites. The density-dependence function that determines $\wedge X(j,t)$ is

$$T \wedge X(j,t) = N(j,t) + K/Q(j,t)$$

where $N(j,t)$ is the number of individuals in patch j at time t, $Q(j,t)$ is the quality of patch j at time t, T is a constant regulating the maximum energy available to an individual foraging alone, and K is a constant determining the steepness of the curve (see Figure 2). This equation is the model's representation of what I have above called the transforming function.

Each individual may reproduce only in every tenth time frame, as determined by its age. (The age, prior residence time and time since last reproduction of each individual are randomly assigned at the start. Thus the simulation begins as though it were already in motion.) The probability of reproduction is calculated by the function

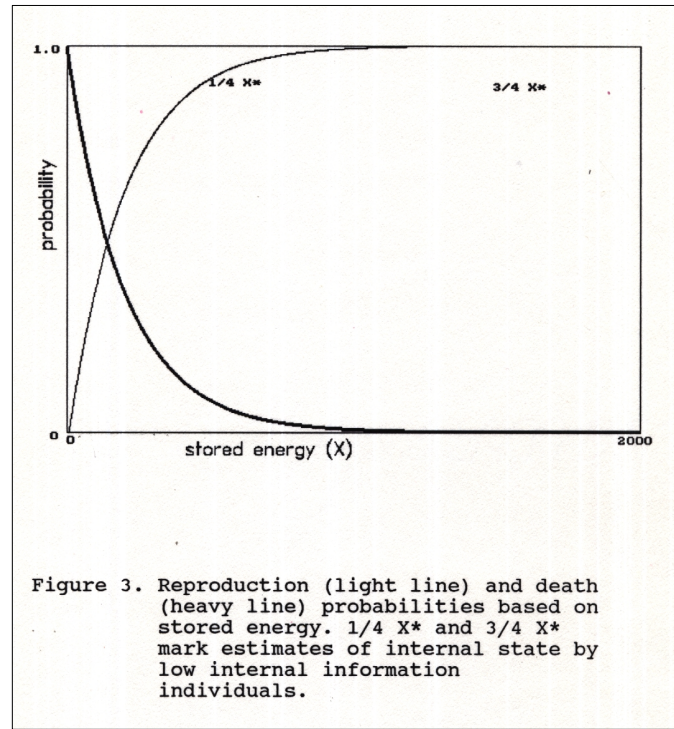$$R[X(i,t)] = 1 - \exp[a(X(i,t) - b)]$$

4

where a determines the steepness of the function and b is an energy threshold below which individuals do not reproduce (Figure 3). Reproduction occurs when a random number selected from a uniform distribution falls below R[X(i,t)]. Death is also a probabilistic function of energy:

$$D[X(i,t)] = -\exp[a(X(i,t))].$$

(To generate sufficient genetic flux for evolution to occur in a reasonable time, the coefficients used in both the reproduction and death equations were high. In effect, evolution was speeded up. However, since the model was designed to represent interactions in a population rather than exact parameters, this does not affect the general results.)

Individuals select a site by calculating how much energy they can expect to gain from feeding at each site, translating these gains into effects on reproductive probability, and choosing the site whose effect is greatest. Travel is instantaneous. After switching sites, an individual must stay in the new site for three time frames. Thus only 33% of the population may move in any one time step.

Figure 3. Reproduction (light line) and death (heavy line) probabilities based on stored energy. 1/4 X* and 3/4 X* mark estimates of internal state by low internal information individuals.

Information-Gathering Genes

High- and low-information alleles for the internal gene work as follows. Each individual keeps an estimate of its internal energy supply. Individuals with the internal high-information allele (I+) have an accurate estimate:

$$E[X(i,t)] = X(i,t).$$

Individuals with low internal information (I-) estimate their energy level as one of two values:

$$E[X(i,t)] = 1/4\ X^* \text{ if } X(i,t) < 1/2\ X^* \text{ or } 3/4\ X^* \text{ if } X(i,t) > 1/2\ X^*$$

where X* is [X(R=0.99) - X(R=0.0)]; so 1/2 X* is the energy level halfway between maximum and minimum reproductive probability. Thus the animal's estimate of its energy supply is only accurate to one half of the range of energy values (see Figure 3). The step function represents a crude perception of hunger versus satiation.

The external information gene affects how individuals update their running estimates of site qualities (Q(j,t)) and densities (N(j,t)). Individuals with the high-information external allele (E+) receive accurate information on all sites every fifth time frame. Low-information individuals (E-) update their information every thirtieth time frame. Thus the E+ individual tracks changes in the environment with six times more precision than the E- type. These updates are provided "from above," without cost and without having to visit sites. Estimates are updated according to a one-step linear operator equation:

$$E[Q(j,t)] = mQ(j,t) + (1-m)E[Q(j,t-1)],$$

where m is a memory coefficient which defines the weighting of past and present estimates (larger m implies faster memory fade).

Differences in relational information affect perception of the density-dependence function. Individuals with the high-information relational allele (R+) use the function itself to predict energy gain at each site. Individuals with the low-information allele (R-) use a linear approximation to the curve (see Figure 2),

$$E[\wedge X(j)] = T - \wedge X(j,1) - C(m) * E[N(j)]. \ 1 - N(j, \wedge Xi = C(m))$$

All individuals use the true reproduction function to predict the effects of possible energy gains on reproductive probability. Thus they maximize actual reproductive probability and not some proximal quantity. I do not suggest that animals make decisions on the basis of reproductive probability, but use the approach to model departures from omniscience.

Results

In preliminary runs, allele frequencies reached stable equilibria within 5000 time frames. To test whether the system would be attracted to these equilibria from any starting conditions, the following analysis was designed. In six 5000-time-frame runs, high-information allele frequencies were begun at 0.25 for each gene. In a second six runs, frequencies were started at 0.50, and in a third six runs at 0.75. Each run began with a different seed value for the random number generator, to simulate different starting conditions. An analysis of variance showed that seed values had no effect on equilibria. Mean high-information allele frequencies were calculated for each run from 190 sample periods in the last 4500 time frames of simulation. A grand mean was then calculated from mean frequencies for the 18 runs.

Table 1 shows the means and standard deviations of allele frequencies for the primary results, and Figures 4, 5, and 6 show sample runs with starting frequencies of 0.25, 0.50 and 0.75 respectively. Since all 18 runs were considered together, a small standard deviation indicates that allele frequencies were attracted to a particular value regardless of starting values. The internal high-information allele reached no consistent equilibrium; the external high-information allele equilibrated around 0.8; and the relational high-information allele equilibrated around 0.5.

Figures 4, 5 and 6: Sample runs from primary results, with high-information frequencies started at 0.25 (Figure 4), 0.50 (Figure 5) and 0.75 (Figure 6) for all genes. Solid line: internal gene; dashed line: external gene; dotted line: relational gene.

Figure 1

6
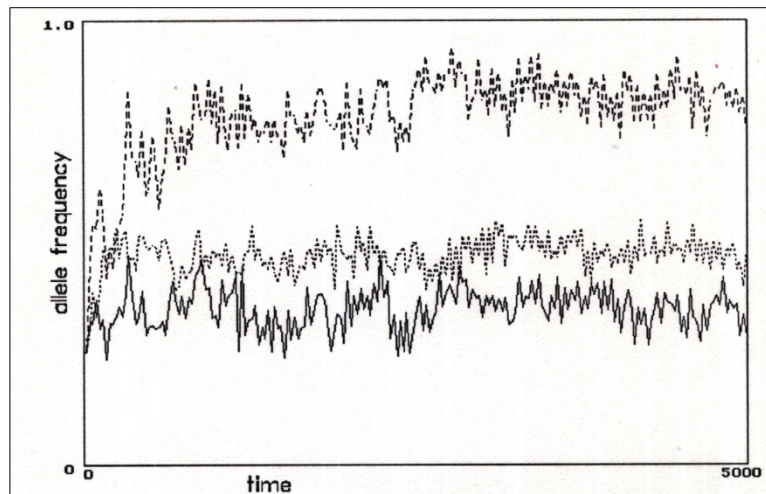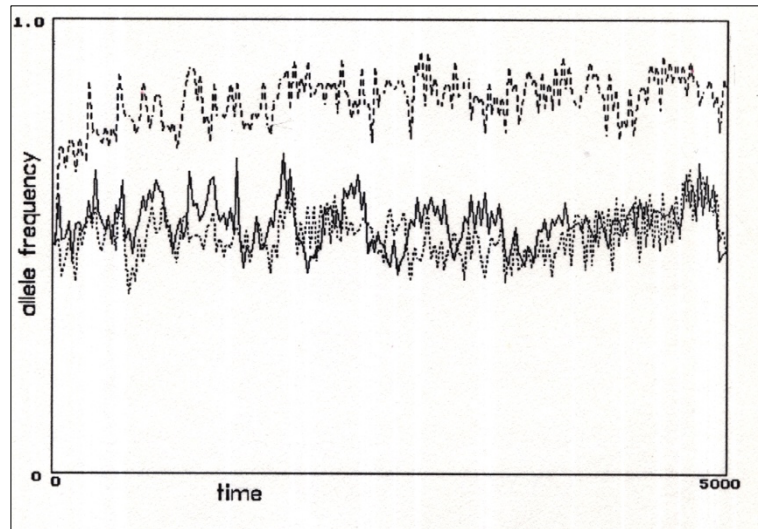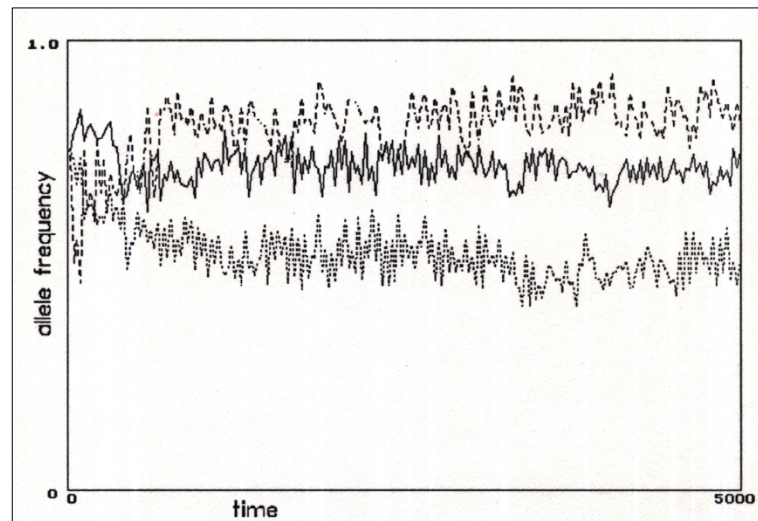
Figure 2

Figure 3

Also, mean allele frequency for each run was regressed against starting allele frequency (0.25, 0.50 or 0.75) separately for each gene. A high regression coefficient indicates that final allele frequencies were heavily influenced by starting frequencies, and therefore were not stable equilibria. Regression coefficients and significance levels are also reported in Table 1.

	internal	external	relational
mean (sd)	0.54 (0.27)	0.80 (0.08)	0.52 (0.04)
regr slope	1.12 ***	0.32 ***	0.12 **

Table 1. Grand means and standard deviations (from 18 runs of 5000 time frames of simulation) of allele frequencies for internal, external and relational high-information alleles. Regression slopes are for regression of mean allele frequency on original frequency (0.25, 0.50 or 0.75 in each run).
* - P < 0.05; ** - P < 0.01; *** - P < 0.001.

Table 2 shows the results of a limited sensitivity analysis. The same 18-run protocol was followed for each variation in model parameters. Variation was considered in the memory coefficient (0.2, 0.5, 0.8), movement cost (2, 5, 10), environmental switching time (25, 100, 5000), and density dependence (present, absent).

Sensitivity Parameters								
Density dependent?	Y	Y	Y	Y	Y	N	Y	Y
Movement cost	2	2	2	5	10	2	2	2
Switch time	25	25	25	25	25	25	100	5000
Memory coefficient	0.5	0.2	0.8	0.5	0.5	0.5	0.5	0.5
Results: Internal Gene								
mean (sd)	0.54	0.48	0.51	0.46	0.53	0.47	0.58	0.60
	(0.26)	(0.28)	(0.27)	(0.33)	(0.29)	(0.27)	(0.24)	(0.27)
regression	1.12	1.06	1.20	1.11	1.36	1.06	1.05	0.87
slope	***	***	***	***	***	***	***	***
Results: External Gene								
mean (sd)	0.80	0.87	0.65	0.86	0.87	0.97	0.79	0.68
	(0.10)	(0.05)	(0.11)	(0.10)	(0.05)	(0.05)	(0.16)	(0.08)
regression	0.32	0.34	-0.06	0.40	0.38	0.06	0.16	0.17
slope	***	***	ns	***	**	ns	**	ns
Results: Relational Gene								
mean (sd)	0.52	0.59	0.55	0.63	0.71	0.69	0.54	0.54
	(0.09)	(0.09)	(0.14)	(0.16)	(0.26)	(0.07)	(0.13)	(0.04)
regression	0.12	0.33	0.32	0.43	0.58	0.26	0.18	0.09
slope	**	***	***	**	**	ns	*	ns

Table 2. Means, standard deviations, and regression slopes for runs in model sensitivity analysis. Details of simulations same as in Table 1. Probability values same as in Table 1.

Interactions between the three genes were examined in a series of genetic environment runs (Table 3). Each gene's activity was noted in four situations: 1) fixation of high-information alleles (+) for both remaining genes, 2) fixation of low-information alleles (-) for both remaining genes, and 3) and 4) fixation of the (+) allele of one gene and the (-) allele of the other. Again allele frequencies were started at 0.25, 0.50 and 0.75 and the means of the three runs for each genetic environment were averaged together.

high-information allele frequency set at 0.25		
INT	EXT	REL
0.546 (0.351)	1.00	1.00
0.416 (0.307)	0.00	0.00
0.520 (0.282)	0.00	1.00
0.505 (0.049)	1.00	0.00
high-information allele frequency set at 0.50		
INT	EXT	REL
1.0	0.529 (0.078)	1.0
0.0	0.610 (0.101)	0.0
0.0	0.520 (0.076)	1.0
1.0	0.605 (0.102)	0.0
high-information allele frequency set at 0.75		

high-information allele frequency set at 0.25		
INT	EXT	REL
1.0	1.0	0.513 (0.058)
0.0	0.0	0.674 (0.194)
0.0	1.0	0.509 (0.059)
1.0	0.0	0.583 (0.125)

Table 3. Means and standard deviations of genetic environment runs. Each mean represents three runs with each high-information allele frequency set at 0.25, 0.50 and 0.75.

An additional set of 32 genetic environment runs was carried out in which the number of individuals of each genotype in each site was tested for correlation with the per capita energy at each site in 190 sampled time periods (Table 4). If the number of individuals is negatively correlated with per capita energy, individuals of that genotype visited sites when they were poor choices. Overall, genotypes with the E+ allele were negatively correlated with per capita energy, and those with E- were positively correlated or not significantly correlated.

genotype number	I E R	# runs	significant positive correlations (proportion)	significant negative correlations (proportion)	nonsignificant correlations (proportion)
1	- - +	36	16 (0.44)	4 (0.11)	16 (0.44)
2	+ - +	30	19 (0.63)	3 (0.10)	8 (0.27)
3	- - -	42	13 (0.31)	12 (0.29)	17 (0.41)
4	+ - -	36	11 (0.31)	4 (0.11)	21 (0.58)
5	- + +	30	2 (0.07)	12 (0.40)	16 (0.53)
6	+ + +	24	2 (0.08)	13 (0.54)	9 (0.38)
7	- + -	36	8 (0.22)	10 (0.28)	18 (0.50)
8	+ + -	30	5 (0.17)	12 (0.40)	13 (0.43)

Table 4. Correlations of site densities of each genotype on per capita energy at sites. Negative correlation implies poor choices. Correlations calculated from 190 time periods in each run during 200 frames of simulation.

Discussion of Model Results

Primary results

Table 1 shows that the external and relational genes exhibited stable allele frequency equilibria. At first glance, this result seems counter-intuitive. One would expect increased information access without cost to fixate rapidly. A closer look at individual movements reveals the explanation: crowding of high-information individuals at high-quality sites. As the proportion of individuals with the high-information allele increases, more of them correctly assess site quality. Individuals visit what appears to be the best site, but cannot predict the simultaneous movements of other individuals. Thus a large number of individuals begin cycling between sites, in effect fleeing from one another. Per capita consumption in each site begins to cycle (Figure 9).

In this situation, the uninformed decisions made by low-information individuals actually put them at a competitive advantage, since the two sites not deemed "best" are often sparsely populated. The uninformed genotypes, making what appear to be errors based on site quality, inadvertently choose the best site based on density. Thus high-information allele frequencies decline — until their descent is checked by the real advantage, under non-crowded conditions, of choosing the best site. The two forces combine to produce equilibria.

Figure 7 shows the density in each site for genotype I-E-R-, the least informed genotype. Figure 8 shows site densities for genotype I+E+R+, the most informed genotype, in the same run. The difference in movement pattern between the two genotypes is clear.

Figure 7. Number of individuals at each site over time for genotype I-E-R- during the same simulation run as in figure 8. This genotype shows little movement.

I tested this explanation for the observed equilibria by removing density dependence. In the set of simulations shown in column 6 of Table 2, per capita consumption was made independent of density. The external gene went to fixation, and the internal and relational genes were neutral. It is clear why the relational gene was neutral in a density-independent environment: there was no relational function to be perceived.

Returning to the density-dependent results, why did the internal gene not reach an equilibrium as the others did? I see two possible reasons. First, because the reproduction function used here is monotonically increasing, misjudging one's point on the energy axis produces no ranking errors. It only produces errors in the estimation of differences between sites. Knowing one's true energy state in these circumstances will only be selectively advantageous when it is important to know the difference between site effects, such as when travel time is not identical between sites.

Figure 8. Number of individuals at each site over time for genotype I+E+R+ during the same simulation run as in figure 7. This informed genotype shows crowding cycles.
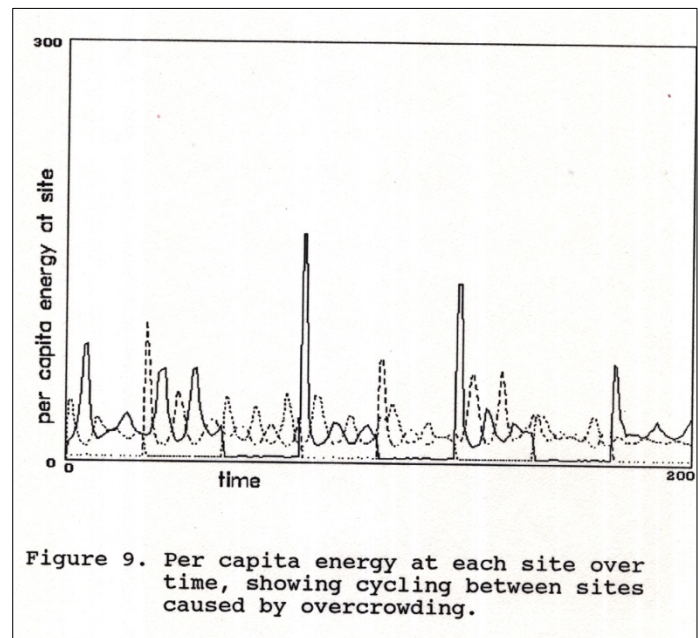
Second, in this model predators never become satiated. If there is no upper bound on feeding, it is not important to be aware of internal state — more consumption is always better. If the model were modified to add an upper threshold for satiation, alleles for increased internal information might respond to selection.
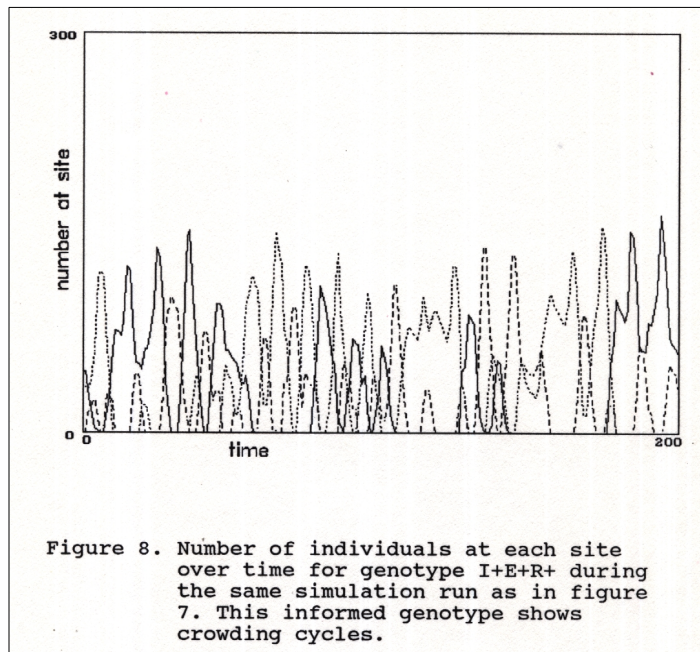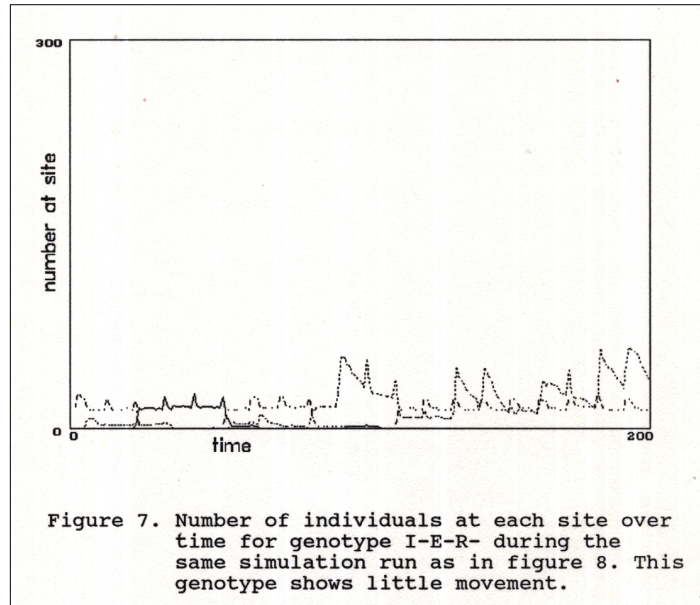
Sensitivity analysis

Memory coefficient. When the memory coefficient was increased from 0.5 to 0.8, causing more rapid fading of memory, the high-information allele frequency of the external gene decreased from 0.80 to 0.65. (All other results were as before.) Thus E+ individuals are even more likely to crowd together when their memory is shortened. Why? To compare different memory coefficients, consider how soon after an event has occurred that its effect on the present decision declines to a negligible amount — how soon it is forgotten. With a memory coefficient of 0.8, the effect is lost after four time steps, while it takes seven time steps with a memory coefficient of 0.5. Because these individuals are constrained to move only once every three time steps, tracking density with a new estimate every four steps is too fine-grained a sensitivity. It is like watching a blinking light and saying, "It's on — no, it's off — no, it's on." Individuals with a memory coefficient of 0.5 were more able to average out fluctuating densities. When the memory coefficient was decreased to 0.2, the E+ equilibrium rose from 0.80 to 0.86. Here, slower memory fade allowed individuals to better track density changes.

Figure 9. Per capita energy at each site over time, showing cycling between sites caused by overcrowding.

The best memory value for any genotype depends on proportions of other genotypes in the population and changes through time. It would be an interesting extension to the model to assign alleles to low and high memory coefficients and see which allele would predominate. McFarland & Houston (1987) suggested that memory should fade more rapidly as the environment becomes more unstable. In this model there are two critical parameters to be tracked: environmental quality, which changes slowly, and site density, which changes rapidly. According to their hypothesis, a small memory coefficient for density and a larger one for site quality would be adaptive. However, that would require individuals to remember site qualities and be aware of conspecifics rather than to remember only past consumption at sites. Most linear operator models have expected individuals to remember only their own consumption (e.g., Shettleworth et al. 1988, Bernstein et al. 1988).

Movement cost. Increasing movement cost increased and destabilized the equilibria of both the external and relational genes. In the impoverished environment, selection for correct choices was intensified and equilibria grew more unstable.

Environmental switching time. Slowing down the rate of environmental change to 100 time frames per site reassignment had little effect on the values of equilibria. Stopping change altogether caused equilibria to depend on the initial distribution of individuals in the sites. Individuals settled into a long-term pattern of shifting between good and medium sites, and those choosing the bad site either moved out of it or died.

Genetic interactions

Table 3 shows three major trends in interactions between the three genes. First, external information proved to be the most crucial. When the E+ allele was fixed (1.0) the relational gene reached an equilibrium, but when E+ was absent (0.0) the relational gene was neutral (Table 3, part c). Therefore in the absence of adequate external information, prediction rules are arbitrary.

Second, the relational gene was important in reacting to conspecifics. Recall that individuals with low relational information replace the density dependence function with a linear approximation. Because of this, R- individuals often make mistakes in site ranking (Figure 10). These mistakes reduce the crowding effect of simultaneous decisions. Thus in a genetic environment of R-, the equilibrium frequency of E+ individuals is higher than that in an environment with all R+ individuals (Table 3, part b). This suggests that rules of thumb not only cost less than complicated decision rules, but stabilize the population's spatial dynamics as well.
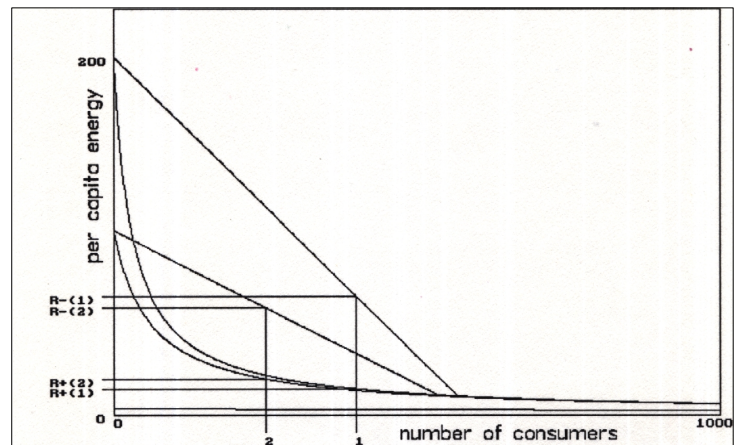
The internal gene reaches an equilibrium in only one genetic environment: fixation of the E+ and R- alleles (Table 3, part a). In this situation, all foragers approximate the density-dependence function. Predicted differences between site consumptions are greater than

Figure 10. Density-dependence function, showiowing site ranking mistakes made by individuals with low relational information (using linear approximations to estimate per capita consumption). R-(1) is the low-relational-information individual's estimate of expected consumption at site 1.

actual differences, except in the short range of the density dependence function (Figure 2) where N is small. The heightened sensitivity causes individuals to overestimate effects of consumption on reproductive probability. Now the difference between I+ individuals (who know their true energy value) and I- individuals (who estimate it) becomes important for the first time. Even though all individuals have good external information (E+), I+ individuals overcrowd, and the others spread out their choices and do better when the informed group is crowded. To a large extent the effect is produced by the chosen linear approximation, which overestimates per capita energy for all density values. The phenomenon does not occur in the environment of E- and R-, because as we have seen above, nothing is important without good external information.

The analysis of site dynamics (Table 4) showed that individuals of E+ genotypes more often chose bad sites (with low per capita energy) than good sites. For E- individuals the opposite was true. Of the E+ genotypes, those with R+ showed stronger negative correlations with per capita energy than those with R-, because of ranking errors made by R- individuals. As a group, individuals of the least informed genotype, I-E-R-, moved practically independently of per capita consumption. However, the behavior of these individuals was not at all random; each made clear, deliberate choices. It was the interaction of their choices with all of the others in the population that produced a random-looking distribution.

To summarize the genetic environment results, the major qualities of each gene are as follows. The external gene is the most important and causes other genes to be nearly trivial in its low-information form. The high-information relational allele increases the accuracy of estimates, adding to whatever synchronous effect is produced by identical external information. The internal gene has little effect in most environments.

General Discussion

These results present the surprising conclusion that alleles for increased information access need not evolve to fixation. But is the conclusion based on a realistic representation of biological systems? How do the assumptions of this model compare to those of other density-dependent foraging models?

The ideal free distribution

The ideal free distribution (IFD) describes the spatial distribution of optimal consumers in a patchy, density-dependent environment (Fretwell and Lucas 1970). It defines optimal consumers as being "ideal," or perfectly

accurate in their assessment, and "free," or equally able to enter all areas being considered. (Specifying free consumers prevents the IFD from applying to despotic populations, since subordinate individuals are not as free as dominants to take advantage of resources.) According to the IFD, optimal competitors will distribute themselves so that no individual receives greater resources than any other. Individual site selection produces an overall distribution in direct proportion to resource availabilities in patches. The IFD has been confirmed qualitatively in bees (Pyke 1980), dungflies (Parker 1970, 1974), toads (Davies & Halliday 1979), sticklebacks (Milinsky 1979), ducks (Harper 1982), and wading birds (Goss-Custard 1977a).

The distribution of consumers in this model should fit the IFD. These individuals are supposedly ideal in their knowledge and equally free to move. However, fluctuating site numbers show that there is no stable distribution here, even when the environment is stable. Why does this model not conform to the IFD? How is it different from previous studies? It is different in two ways: first, all individuals move simultaneously; and second, sites are chosen based on a definite ranking of sites and not on a probability of movement. I believe that while these conditions are unrealistic, they point out hidden assumptions in previous studies of the IFD that are also unrealistic.

Theoretical treatments of the IFD can be broken into analytical and simulation models. Each type of model typically contradicts one of the two conditions of this model. Analytical models contradict the first, that of simultaneous movement.

Time and the ideal free distribution

In analytical studies of the ideal free distribution, animals are assumed to move one at a time between sites (Comins & Hassell 1979, Sutherland 1983, Pulliam & Caraco 1984, Pulliam 1987). These models speak of some animals "first" filling the best site, then "additional" animals filling the second best site, and so on (Fretwell & Lucas 1970). Sibly (1983) was the first to state the restriction explicitly: "...each choosing to feed where it obtains maximum individual advantage, subject to the constraint that decisions are only made by one individual at a time...." Davies (1982), in describing the "behavioural assessment ESS" or "spatial ESS" (Parker 1978), goes further. He states, "The stable distribution would come about by each individual ... sampling the various habitats and choosing the most profitable one, which will depend on where the other individuals have decided to go.... The strategy would therefore be something like, 'if most of the population are at A, go to B.'" How can each individual decide where to go after all the others "have decided?" This is clearly a problem.

The representation of time is an important conflict between analytical and simulation models. Simulations assign a common time interval for the decisions of all individuals, and do not allow any individual to act outside of that simultaneity. Analytical solutions assign an exclusive time interval to each individual decision, and do not allow any two individuals to make decisions at the same time. Both approaches place limits on our representation of reality, which must lie somewhere between these extremes.

I argue that simultaneity has been under-represented in foraging analyses. Imagine the following scenario. An individual is at this moment deciding whether or not to join an existing group. What are the group members doing? They are reevaluating their options, as they may do at any time. They may be aware of the individual's proximity, and if so, the group's coherence may be affected by the behavior of the deciding individual as it decides. Because of simultaneity an individual will always carry an element of uncertainty in its decision. Because of simultaneity neat distributions of even "ideal" foragers cannot be realistic.

On the other hand, does completely synchronous decision making pose a serious difficulty for simulation models? First we must examine what makes time discrete or continuous. Discrete time is only time in which an important factor changes abruptly after a fixed interval of (continuous) time. In most situations encountered by foraging animals, time is not discrete. To an animal watching for predators, every second has the potential to change important conditions. But there are other situations in which time can more usefully be described as discrete, because what happens between changes in the most important factor is immaterial. Three factors suggest themselves: 1) dependence on an ephemeral resource (the resource changes abruptly), 2) long travel

time (the condition of resources changes during travel; it may not be an abrupt change, but to the returning animal it appears so), and 3) high travel cost (the condition of the forager changes during travel; again, the change is abrupt in the experience of the traveling animal). Each theoretical approach should design a plausible representation of time for the situation it studies.

For situations in which complete synchrony is unrealistic, simulations can be designed in which individuals make decisions in partial synchrony. Typically decisions represent one mass activity that takes place in a simulation's time loop. The decision process could be subdivided into smaller pieces so that any number of individuals could decide at the same time, from one to the entire population. Actually, the procedure only subdivides time into very tiny pieces, so that its discrete nature is hidden. Designing a simulation to allow partial synchrony would not be a trivial task, but would greatly improve the reality of the model over any approach to date.

Choice and the ideal free distribution

Since simulation models of the ideal free distribution have been reported, why do they not agree with these results? Because they contradict the second condition of this model, that animals make unambiguous choices. In Parker's (1978) model of dungflies searching for females, males arrive at the breeding site simultaneously, but leave according to a "distribution of emigration thresholds for departure." Lester (1984) and Regelmann (1984) both assign their model individuals to new sites based on a probability of movement to each site. Bernstein et al. (1988) randomly reassign predators to sites; the decision is how long to stay at a site once there. In each of these cases, even though individuals decide simultaneously, their movement is broken up so that decision-dependent crowding does not take place. Surprisingly, Bernstein et al. (1988) do describe a modification of their model in which consumers are "best informed" — that is, they no longer move randomly between sites, but choose one based on an estimation of quality. The site dynamics of their best informed consumers resemble those of my high-information individuals. They comment, "The fact that our simulations show that the distribution of 'omniscient' predators lies farther from the IFD than a population of learners is not equivalent to claiming that learners would be favored by selection. In an evolutionary game where most predators are learners, an informed predator ... would surely spread, although we have not investigated what the stable state for the population would be."

What difference does it make whether choices consist of movement probabilities or imprecise site ranking? After all, both mechanisms counteract overcrowding problems. The difference is in awareness. In order for animals to assign probabilities to movement in response to uncertain environmental states, as in the above models, they must be aware of the fact that the environment is unstable. In effect, individuals must be aware of their ignorance. This model illustrates a more plausible situation. I propose that each individual assumes its chosen strategy to be the correct one, but that some individuals are wrong for the particular situation they are in. Selection then supports a range of behaviors that persists in a fluctuating environment.

The idea that selection upholds variability in behavior has been suggested by other authors. Nur (1987) reviewed studies of clutch size to explain the consistent variability in the clutch size of some bird species. He supported the hypothesis that a range of clutch sizes is maintained by unpredictable fluctuation in the payoff matrix. He suggested that clutch size cannot be described by a mixed or conditional ESS; rather, it follows no equilibrium at all because of constantly changing optima.

Clark and Ehlinger (1987) also suggested that individual differences in behavior persist when environmental conditions change so rapidly that the optimal behavior cannot be anticipated. In particular, Ehlinger (1986) found that bluegill sunfish can be separated into two types based on a tendency to switch to a higher-quality site earlier or later in foraging bouts. Simulations showed that early switchers did best in conditions of moderate variability in prey density, while late switchers did best with either high or low variability in prey density. Since the variability in prey density can itself change unpredictably, a mix of the two types in the population is supported. Finally, Slater (1981) stated that "...where the environment changes in a manner which is more

probabilistic than deterministic,... the behavioral option taken by the individual is more of a gamble, and high variability is to be expected."

The ideal free distribution was intended to represent a population-level compilation of individually optimal decisions. Dealing with simultaneous decision making by having individuals move randomly or based on probability makes the IFD not a distribution rising out of optimal individual decisions, but individual foraging behavior suited to fit a distribution determined as optimal. This defeats the entire purpose of the IFD as a description of the population effects of optimality.

Conclusions

Let us return to the two questions posed at the beginning of the thesis. The first was: what can an analysis of cost-free information use tell us about the value of information? Simulation has shown that having more information does not always confer a selective advantage, even when the information is free. The fitness value of free information is not monotonically increasing. Rather, it covers a complex surface of adaptive peaks and valleys. Any solution must take into account the accessibility of peaks as well as their height. Individuals may be unable to maximize fitness through knowledge of the decisions made by conspecifics. That knowledge may involve a considerable investment in observation and cognition, or perhaps manipulation. Instead, they give up some information and climb another adaptive peak.

The second question posed was: how can the great variety of incomplete information problems be unified? Three major types of information used in the decision-making process were described. Internal information concerns internal state; external information concerns the state of the environment external to the animal; and relational information concerns interactions between internal and external states. The three types of information outlined here may help to evaluate similarities in information use between different situations. Simulation did not produce conclusive results, but suggested interesting ideas for further exploration.

Intuition is given a high status in foraging theory. We often declare that something "is intuitively obvious" (Stillman and Sutherland 1990) or "fits with intuition" (Stephens 1989), and label results that do not match our expectations as "paradoxical" (McNamara and Houston 1980). What exactly is intuition? The Oxford English Dictionary defines it as "immediate insight by sense." The important word here is sense. Our sense is one of reasoning, cultural, educated animals. Our sense is not that of other species, and it is certainly not that of natural selection. The conclusions presented here are not obvious according to current intuition, nor are many of the results produced by game theory analyses and individual-based population models. Perhaps a new "intuition" which takes account of factors above the level of the individual is appropriate. In the future, an integrative perspective may not be considered a special approach to behavioral biology or of special concern to social behaviorists, but as a framework within which all behavior must be viewed.

Literature Cited

Bernstein, C., A. Kacelnik and J.R. Krebs. 1988. Individual decisions and the distribution of predators in a patchy environment. J. Anim. Ecol. 57:1007-1026.

Charnov, E.L. 1976b. Optimal foraging: the marginal value theorem. Theor. Pop. Biol. 9:129-136.

Clark, A.B., and T.J. Ehlinger. 1987. Pattern and adaptation in individual behavioral differences. In P.P.G. Bateson and P.H. Klopfer, eds., Perspectives in Ethology, vol. 7, pp. 1-39. Plenum, New York.

Comins, H.N., and M.P. Hassell. 1979. The dynamics of optimally foraging predators and parasitoids. J. Anim. Ecol. 48:335-351.

Davies, N.B. 1982. Behaviour and competition for scarce resources. In King's College Sociobiology Group, eds., Current Problems in Sociobiology, pp. 363-380. Cambridge University Press, Cambridge.

Davies, N.B., and T.R. Halliday. 1979. Competitive mate searching in common toads, Bufo bufo. Anim. Behav. 27:1253-1267.

Ehlinger, T.J. 1986. Learning, sampling and the role of individual variability in the foraging behavior of bluegill sunfish. Ph.D. dissertation, Michigan State University.

Fretwell, S.D., and H.L. Lucas. 1970. On territorial behavior and other factors influencing habitat distribution in birds. I. Theoretical development. Acta Biotheor. 19:16-36.

Goss-Custard, J.D. 1977a. Optimal foraging and the size selection of worms by redshank, Tringa totanus, in the field. Anim. Behav. 25:10-29.

Gould, J.P. 1974. Risk, stochastic preference, and the value of information. J. Econ. Theory 8:64- 84.

Green, R.F. 1984. Stopping rules for optimal foragers. Am. Nat. 123:30-40.

Harper, D.G.C. 1982. Competitive foraging in mallards: 'ideal free' ducks. Anim. Behav. 30:575- 584.

Janetos, A.C. & B.J. Cole. 1981. Imperfectly optimal animals. Behav. Ecol. Sociobiol. 9:203-210.

Krebs, J.R., A. Kacelnik, & P. Taylor. 1978. Test of optimal sampling by foraging great tits. Nature 275:27-31.

Krebs, J.R., J.C. Ryan, & E.L. Charnov. 1974. Hunting by expectation or optimal foraging? A study of patch use by chickadees. Anim. Behav. 22:953- 964.

Lester, N.P. 1984. The 'feed:feed' decision: how goldfish solve the patch depletion problem. Behaviour 89:175-199.

Lima, S.L. 1983. Downy woodpecker foraging behavior: efficient sampling in simple stochastic environ ments. Ecology 65:166-174.

Mangel, M. 1990. Dynamic information in uncertain and changing worlds. J. Theor. Biol. 146:317-332. Mangel, M. & C.W. Clark. 1986. Towards a unified foraging theory. Ecology 67:1127-1138.

Mangel, M. & C.W. Clark. 1988. Dynamic Modeling in Behavioral Ecology. Princeton University Press, Princeton, NJ.

McNamara, J.M. & A.I. Houston. 1980. The application of statistical decision theory to animal behavior. J. Theor. Biol. 85:673-690.

McNamara, J.M. & A.I. Houston. 1987. Memory and the efficient use of information. J. Theor. Biol. 125:385-395.

Milinsky, M. 1979. An evolutionarily stable feeding strategy in sticklebacks. Z. Tierpsychol. 51:36- 40.

Nur, N. 1987. Alternative reproductive tactics in birds: individual variation in clutch size. In Bateson, P.P.G. & P.H. Klopfer, eds., Perspectives in Ethology, vol. 7. pp. 49-76. Plenum Press, New York.

Orians, G.H. 1981. Foraging behavior and the evolution of discriminatory abilities. In A.C. Kamil and T.D. Sargent, eds., Foraging Behavior: Ecological, Ethological and Psychological Approaches. pp. 389-405. Garland STPM Press, New York.

Parker, G.A. 1970a. The reproductive behavior and the nature of sexual selection in Scatophaga stercoraria L. II. The fertilization rate and the spatial and temporal relationships of each sex around the site of mating and oviposition. J. Anim. Ecol. 39:205- 228.

Parker, G.A. 1974c. Assessment strategy and the evolution of fighting behavior. J. Theor. Biol. 47:223-243.

Parker, G.A. 1978. Searching for mates. In Krebs, J.R. & N.B. Davies, eds., Behavioural Ecology: An Evolutionary Approach, 1st ed. pp. 214-244. Sinauer Associates, Sunderland, MA.

Pulliam, H.R. 1987. On the evolution of density-regulating behavior. In Bateson, P.P.G. & P.H. Klopf er, eds., Perspectives in Ethology, vol. 7. pp. 99-122. Plenum Press, New York.

Pulliam, H.R. & T. Caraco. 1984. Living in groups: Is there an optimal group size? In Krebs, J.R. & N.B. Davies, eds., Behavioural Ecology: An Evolutionary Approach, 2nd ed. pp. 122-147. Blackwell Scientific Publications, Oxford.

Pyke, G.H. 1980. Optimal foraging in bumblebees: calculation of net rate of energy intake and optimal patch choice. Theor. Pop. Biol. 17:232-246.

Raiffa, H. 1968. Decision Analysis. Addison-Wesley, Reading, MA.

Regelmann, K. 1984. Competitive resource sharing: a simulation model. Anim. Behav. 32:226-232.

Schoener, T.W. 1971. Theory of feeding strategies. Ann. Rev. Ecol. Syst. 2:369-404.

Shettleworth, S.J., J.R. Krebs, D.W. Stephens, & J. Gibbon. 1988. Tracking a fluctuating environment: a study of sampling. Anim. Behav. 36:87-105.

Sibly, R.M. 1983. Optimal group size is unstable. Anim. Behav. 31:947-948.

Slater, P.J.B. 1981. Individual differences in animal behavior. In Bateson, P.P.G. & P.H. Klopfer, eds., Perspectives in Ethology, vol. 4. pp. 35- 49. Plenum Press, New York.

Stephens, D.W. 1989. Variance and the value of information. Am. Nat. 134:128-140.

Stephens, D.W. & J.R. Krebs. 1986. Foraging Theory. Princeton University Press, Princeton, NJ.

Stillman, R.A. & W.J. Sutherland. The optimal search path in a patchy environment. J. Theor. Biol. 145:177-182.

Sutherland, W.J. 1983. Aggregation and the 'ideal free' distribution. J. Anim. Ecol. 52:821-828.

Sykes, J.B. 1976. The Concise Oxford Dictionary of Current English, 6th ed. University Press, Oxford.

Appendix: Program Code

```pascal
PROGRAM inform;

{ Notes : LSP = low sampling period, NL = nonlinear, PV = personal }
{ value. These are the high-information alleles.                   }
{ Coeff = a, the coefficient of the reproduction function.        }
{ Qual# is the quality of each site.                              }

USES crt;

CONST startsize =  300;          coeff = -0.005; {-0.000175;}
      qual1 = 500;               memory = 0.5;
      qual2 = 15000;             maintcost = 10;
      qual3 = 29500;             babycost = 80.0;
      maxindv = 1000;            startfood = 450;
      threshold = 10;            T = 6000;
      K = 884000;                kick = 0;
      P = 1.0;                   refractory = 10;
      move = 2.0;

TYPE real3array = array[1..3] of real;
     int3array = array[1..3] of integer;
     three = 1..3;
     eight = 1..8;

     onebody = record
               speriod : boolean;        nonlinear : boolean;
               persvalue : boolean;      SE : real;
               howlong : integer;        estimateSE : real;
               estimateQ : real3array;   estimateN : real3array;
               where : three;            alive : boolean;
               count : 0..10;            age : integer;
               moved : boolean;          end;

     place = record
          kind : three;                  quality : real;
          attendance : integer;          totNL : integer;
          totLSP : integer;              totPV : integer;
          addSE : real;                  end;


     placearray = array[1..3] of place;
     bodyarray = array[1..maxindv] of onebody;
     eightarray = array[1..8] of 0..1000;
     threeeight = array[1..3] of array[1..8] of 0..1000;

VAR indv :                        array[1..1000] of 1..1000;
    body :                        bodyarray;
    site :                        placearray;
    know :                        eightarray;
    gensite :                     threeeight;
{   trans :                        array[1..3] of array[1..3] of integer;}
    qual :                        array[1..3] of integer;

    freqLSP, freqNL,
    probdeath,
    freqPV, seed,
    expectedkids, justanumber,
    startLSP, startNL, startPV :     real;

    totLSP, totNL, totPV,
    i, j, q, time,{ oldtime,}
    printyear, totaltime,
    popsize, babies, dead,
    whom, run, jk, this, kid, switch :       integer;

    decided, extinct, screenupdate : boolean;
    out{, error} :                text;
    ans :                         char;

FUNCTION UniRand(Seed:real; Var TinySeed: real): real;
const a = 16807.0;
```

```
      m = 2147483647.0;
var this : real;
begin
     this := a*Seed;
     Seed := this - m*(trunc(this/m));
     UniRand := Seed/m;
     TinySeed := Seed;
     end;

FUNCTION smintrand(most: integer; seed: real; var newseed: real): integer;
var bottom, top, where: real;
    i : integer;
begin
     top := 0;
     smintrand := 0;
     where := unirand(seed,newseed);
     for i := 1 to most do begin
         bottom := top;
         top := top + 1.0/most;
         if (where > bottom) and (where <= top) then smintrand := i;
         end;
     end;

FUNCTION intrand(most: integer; seed: real;
        var newseed:real): integer;
begin
    intrand:= trunc(most*unirand(seed,newseed)) + 1;
    end;

FUNCTION NormRand(mean: real; variance: real; seed: real;
                                    var newseed: real): real;
var j : integer;
    total : real;
begin
    total := 0;
    for j := 1 to 12 do total := unirand(seed,seed) + total;
    normrand := ((total - 6)*sqrt(variance)) + mean;
    end;

FUNCTION max(var efft: real3array): integer;
var i, this: integer;
begin
     this := 1;
     for i := 2 to 3 do if efft[i] > efft[this] then this := i;
     max := this;
     end;

PROCEDURE changesites(var sites: placearray;
        seed: real; var newseed: real);
{ reassigns identities of sites }
var q: integer;

begin
    sites[1].kind := intrand(3,seed,seed);
    if sites[1].kind = 1 then begin
         sites[2].kind := intrand(2,seed,seed) + 1;
         if sites[2].kind = 2 then sites[3].kind := 3;
         if sites[2].kind = 3 then sites[3].kind := 2;
         end;
    if sites[1].kind = 2 then begin
         sites[2].kind := intrand(2,seed,seed);
         if sites[2].kind = 2 then sites[2].kind := 3;
         if sites[2].kind = 1 then sites[3].kind := 3;
         if sites[2].kind = 3 then sites[3].kind := 1;
         end;
       if sites[1].kind = 3 then begin
         sites[2].kind := intrand(2,seed,seed);
         if sites[2].kind = 1 then sites[3].kind := 2;
         if sites[2].kind = 2 then sites[3].kind := 1;
         end;

     end;

PROCEDURE NLeffect(energy: real; estqual: real3array;
```

```
                  estatt: real3array; var wheretogo: three; mve: real);
{ calculates effect on probability of reproduction of eating }
{ at each site based on estimates of Q and N }
{ using real density-dependent function }

var Pnow,difference : real;
    addSE,effect: real3array;
    i,j : integer;
begin
      Pnow := 1 - exp(coeff*(energy - threshold));
      for j := 1 to 3 do begin
          addSE[j] := T/(P*estatt[j] + K/estqual[j]);
          if j <> wheretogo then addSE[j] := addSE[j] - mve;
          end;
      for j := 1 to 3 do effect[j] :=
          (1 - exp(coeff*(energy + addSE[j] - threshold))) - Pnow;
    { if time <> oldtime then begin
          for j:=1 to 3 do write(error,effect[j]:12:3); writeln(error);
          end; }
    {  difference := 0;
      difference := effect[max(effect)] - effect[wheretogo];
      if difference > effect[wheretogo]/move then} wheretogo := max(effect);
      end;


PROCEDURE Leffect(energy: real; estqual: real3array;
          estatt: real3array; var wheretogo: three; mve: real);
{ calculates effect on probability of reproduction of eating }
{ at each site based on estimates of Q and N }
{ using linear approximation to density-dependent function }

var Pnow,difference : real;
    addSE, effect : real3array;
    i, j : integer;

begin

      Pnow := 1 - exp(coeff*(energy - threshold));
      for j := 1 to 3 do begin
          addSE[j] := (((T/(1+ K/estqual[j]))-maintcost)/
                      (1 - (T/maintcost-K/estqual[j])/P) )*
                      estatt[j] + (T/(1+K/estqual[j]));
          if j <> wheretogo then addSE[j] := addSE[j] - mve;
          end;
      for j := 1 to 3 do effect[j] :=
          (1 - exp(coeff*(energy + addSE[j] - threshold))) - pnow;
    {   difference := effect[max(effect)] - effect[wheretogo];
      if difference > effect[wheretogo]/move then} wheretogo := max(effect);
      end;


BEGIN


{FOR RUN := 2 TO 9 DO BEGIN
totaltime := 4000;
screenupdate := false;
seed := 12.3456789;
writeln('RUN NUMBER ',run);
case run of
    1 : begin
              assign(out,'d:\cynthia\ge6a');
              startLSP := 1.0; startNL := 1.0; startPV := 0.5;
              end;
    2 : begin
              assign(out,'d:\cynthia\ge6b');
              startLSP := 0.75; startNL := 0.0; startPV := 0.0;
              end;
    3 : begin
              assign(out,'d:\cynthia\ge6c');
              startLSP := 0.25; startNL := 0.0; startPV := 0.0;
              end;
    4 : begin
              assign(out,'d:\cynthia\ge7a');
              startLSP := 0.5; startNL := 1.0; startPV := 0.0;
              end;
```

```pascal
      5 : begin
                assign(out,'d:\cynthia\ge7b');
                startLSP := 0.75; startNL := 1.0; startPV := 0.0;
                end;
      6 : begin
                assign(out,'d:\cynthia\ge7c');
                startLSP := 0.25; startNL := 1.0; startPV := 0.0;
                end;
      7 : begin
                assign(out,'d:\cynthia\ge8a');
                startLSP := 0.5; startNL := 0.0; startPV := 1.0;
                end;
      8 : begin
                assign(out,'d:\cynthia\ge8b');
                startLSP := 0.75; startNL := 0.0; startPV := 1.0;
                end;
      9 : begin
                assign(out,'d:\cynthia\ge8c');
                startLSP := 0.25; startNL := 0.0; startPV := 1.0;
                end;
     10 : begin
                assign(out,'d:\cynthia\ge4a');
                startLSP := 1.0; startNL := 0.0; startPV := 0.5;
                end;
     11 : begin
                assign(out,'d:\cynthia\ge4b');
                startLSP := 1.0; startNL := 0.0; startPV := 0.75;
                end;
     12 : begin
                assign(out,'d:\cynthia\ge4c');
                startLSP := 1.0; startNL := 0.0; startPV := 0.25;
                end;
     end;}
assign(out,'inform.out');
rewrite(out);
{assign(error,'trans.out');
rewrite(error);
oldtime :=1;   }
qual[1]:=qual1; qual[2]:=qual2; qual[3]:=qual3;
extinct := false;
clrscr;
writeln('Enter starting frequencies of Internal, External, Relational HI alleles.');
readln(startPV, startLSP, startNL);
writeln('Enter total time');
readln(totaltime);
writeln('Switch the sites how often?');
readln(switch);
if totaltime < 200 then printyear := 1 else
   printyear := trunc((totaltime + 1)/200);
{writeln('Enter cost of movement (maintenance cost is ',maintcost:4,')');
readln(move);       }
writeln('Enter seed');
readln(seed);
writeln('Do you want a screen update? (Y or N)');
repeat
     ans := readkey;
     if ans in ['Y','y'] then screenupdate := true;
     until ans in ['y','Y','n','N'];
writeln(out,'Frequencies ',startPV:4:1,startLSP:4:1,startNL:4:1,' Time ',totaltime:6,' threshold
',threshold:5);
    writeln(out,'Memory ',memory:4:1,' seed ',seed:16:12,'  maintcost ',maintcost:4);
writeln('*** RUNNING SIMULATION. PRESS ANY KEY TO SUSPEND PROGRAM. ***');

                         { initialize arrays }

for i := 1 to 1000 do indv[i] := i;
for i := 1 to 1000 do body[i].alive := false;

for i := 1 to startsize do begin
    with body[indv[i]] do begin
        SE := normrand(startfood,startfood/2,seed,seed);
        estimateSE := SE;
        for j:=1 to 3 do begin
            estimateQ[j] := qual[2];
```

```
                estimateN[j] := startsize/3;
            end;
        alive := true;
        where := smintrand(3,seed,seed);
        howlong := smintrand(3,seed,seed);
        count := 0;
        age := intrand(4999,seed,seed);
        end;
    end;

                { assign allele frequencies }

for i := 1 to startsize do body[indv[i]].speriod := false;
if startLSP <> 1.0 then begin
for i := 1 to round(startLSP*startsize) do begin
    repeat this := intrand(startsize, seed,seed)
          until body[indv[this]].speriod = false;
    body[indv[this]].speriod := true;
    end;
    end else for i := 1 to startsize do body[indv[i]].speriod := true;

for i := 1 to startsize do body[indv[i]].nonlinear := false;
if startNL <> 1.0 then begin
for i := 1 to (trunc(startNL*startsize) + 1) do begin
    repeat this := intrand(startsize, seed, seed)
          until body[indv[this]].nonlinear = false;
    body[indv[this]].nonlinear := true;
    end;
    end else for i := 1 to startsize do body[indv[i]].nonlinear := true;

for i := 1 to startsize do body[indv[i]].persvalue := false;
if startPV <> 1.0 then begin
for i := 1 to (trunc(startPV*startsize) + 1) do begin
    repeat this := intrand(startsize, seed, seed)
          until body[indv[this]].persvalue = false;
    body[indv[this]].persvalue := true;
    end;
    end else for i := 1 to startsize do body[indv[i]].persvalue := true;

                { calculate initial site attendance }

for i := 1 to 3 do site[i].attendance := 0;
for i := 1 to startsize do site[body[indv[i]].where].attendance :=
    site[body[indv[i]].where].attendance + 1;

time := 0;
for j := 1 to 3 do site[j].quality := 0;
popsize := startsize;
time := 0; babies := 0; dead := 0;
extinct := false;

WHILE TIME <= TOTALTIME DO BEGIN

   if keypressed then begin
      writeln('Press s to stop and g to go on');
      repeat
         ans := readkey;
         if ans in ['s','S'] then halt(1);
         if ans in ['g','G'] then writeln('Press any key to stop again.');
         until ans in ['s','S','g','G'];
       end;

    time := time + 1;
    babies := 0; dead := 0;

                    { switch sites around }

    if (time = 1) or ((time/switch) = trunc(time/switch)) then
        changesites(site,seed,seed);

         { select site qualities for time step from normal dist }

    for q := 1 to 3 do site[q].quality := normrand(qual[site[q].kind],
       qual[site[q].kind]/10.0,seed, seed);
```

```
                        { update estimates }

  for i := 1 to popsize do begin
      whom := indv[i];
      if body[whom].persvalue then
         body[whom].estimateSE := body[whom].SE
         else begin
             if body[whom].SE > 0.5*ln(0.001)/(-1*coeff) then
                 body[whom].estimateSE := 0.75*ln(0.001)/(-1*coeff)
                 else body[whom].estimateSE := 0.25*ln(0.001)/(-1*coeff);
         end;
      if (body[whom].speriod) and (body[whom].howlong = 1) then begin
         for j := 1 to 3 do begin
             body[whom].estimateQ[j] := (memory*site[j].quality)
                 + (body[whom].estimateQ[j]*(1-memory));
             body[whom].estimateN[j] := (memory*site[j].attendance)
                 + (body[whom].estimateN[j]*(1-memory));
             end;
         end;
      if (not body[whom].speriod) and (body[whom].count = 10) then begin
         for j := 1 to 3 do begin
             body[whom].estimateQ[j] := (memory*site[j].quality)
                 + (body[whom].estimateQ[j]*(1-memory));
             body[whom].estimateN[j] := (memory*site[j].attendance)
                 + (body[whom].estimateN[j]*(1-memory));
             end;
         body[whom].count := 0;
         end;
  end;

              { individual evaluations and movements }

  jk := 0;
{ for i:=1 to 3 do for j:=1 to 3 do trans[i,j] := 0;}
  for i := 1 to popsize do begin
         whom := indv[i];
         jk := body[whom].where;
         body[whom].moved := false;
         with body[whom] do begin
             if howlong >= 3 then begin
                 if nonlinear then NLeffect(estimateSE, estimateQ,
                    estimateN, where, move)
                    else Leffect(estimateSE, estimateQ, estimateN,
                    where, move);
                 howlong := 0;
                 if not speriod then count := count + 1;
                 end;
             end;
         body[whom].howlong := body[whom].howlong + 1;
         body[whom].age := body[whom].age + 1;
         if jk <> body[whom].where then body[whom].moved := true;
        { trans[jk,body[whom].where] := trans[jk,body[whom].where] + 1;}
         end;
{ writeln(error,time:5);
  for i:=1 to 3 do begin
      for j:=1 to 3 do write(error,trans[i,j]:8);
      writeln(error);
      end;
  writeln(error); }

                    { calculate site attendance }

  for i := 1 to 3 do site[i].attendance := 0;
  for i := 1 to popsize do begin
      for j := 1 to 3 do if body[indv[i]].where = j then site[j].attendance := site[j].attendance +
1;
      end;

                { calculate per capita consumption }

  for j := 1 to 3 do site[j].addSE :=
      T/(P*site[j].attendance + (K/site[j].quality));
```

```pascal
                        { calculate site totals for output }

    totLSP := 0; totNL := 0; totPV := 0;
    freqLSP := 0; freqNL := 0; freqPV := 0;
{   for j := 1 to 3 do site[j].totLSP := 0;
    for j := 1 to 3 do site[j].totNL := 0;
    for j := 1 to 3 do site[j].totPV := 0; }
    for i := 1 to popsize do begin
        whom := indv[i];
        if body[whom].speriod then begin
            totLSP := totLSP + 1;
          {  site[body[whom].where].totLSP :=
             site[body[whom].where].totLSP + 1; }
            end;
        if body[whom].nonlinear then begin
            totNL := totNL + 1;
          {  site[body[whom].where].totNL :=
                site[body[whom].where].totNL + 1; }
            end;
        if body[whom].persvalue then begin
            totPV := totPV + 1;
          {  site[body[whom].where].totPV :=
                site[body[whom].where].totPV + 1;  }
            end;
        end;
    freqLSP := totLSP/popsize; freqNL := totNL/popsize;
    freqPV := totPV/popsize;

          { calculate genotype frequencies for output }

    for i := 1 to 8 do know[i] := 0;
    for i := 1 to 3 do for j := 1 to 8 do gensite[i,j] := 0;
    for i := 1 to popsize do begin
        with body[indv[i]] do begin
            case speriod of
                false: begin
                    case nonlinear of
                        true: begin
                            case persvalue of
                                true: begin
                                    know[2] := know[2] + 1;
                                    for q := 1 to 3 do if
                                        where = q then
                                        gensite[q,2] := gensite[q,2] + 1;
                                    end;
                                false: begin
                                     know[1] := know[1] + 1;
                                     for q := 1 to 3 do if
                                        where = q then
                                        gensite[q,1] := gensite[q,1] + 1;
                                    end;
                                end;
                            end;
                        false: begin
                            case persvalue of
                                true: begin
                                    know[4] := know[4] + 1;
                                    for q := 1 to 3 do if
                                        where = q then
                                        gensite[q,4] := gensite[q,4] + 1;
                                    end;
                                false: begin
                                     know[3] := know[3] + 1;
                                     for q := 1 to 3 do if
                                        where = q then
                                        gensite[q,3] := gensite[q,3] + 1;
                                    end;
                                end;
                            end;
                        end;
                    end;
                true: begin
                    case nonlinear of
                        true: begin
```

```pascal
                     case persvalue of
                         true: begin
                             know[6] := know[6] + 1;
                             for q := 1 to 3 do if
                                 where = q then
                                 gensite[q,6] := gensite[q,6] + 1;
                             end;
                         false: begin
                             know[5] := know[5] + 1;
                             for q := 1 to 3 do if
                                 where = q then
                                 gensite[q,5] := gensite[q,5] + 1;
                             end;
                         end;
                     end;
                 false: begin
                     case persvalue of
                         true: begin
                             know[8] := know[8] + 1;
                             for q := 1 to 3 do if
                                 where = q then
                                 gensite[q,8] := gensite[q,8] + 1;
                             end;
                         false: begin
                              know[7] := know[7] + 1;
                              for q := 1 to 3 do if
                                 where = q then
                                 gensite[q,7] := gensite[q,7] + 1;
                             end;
                         end;
                     end;
                 end;
             end;
         end;
     end;
  end;


                         { output }


if (time > 1000) or (time/printyear = trunc(time/printyear)) then begin
    writeln(out,'Time = ',time:4);
    for i := 1 to 8 do write(out,know[i]:5);
    writeln(out);
    writeln(out, freqLSP:8:4, freqNL:8:4, freqPV:8:4, popsize:5);
 {  writeln(out,site[1].totLSP:4,site[1].totNL:4,site[1].totPV:4);
    writeln(out,site[2].totLSP:4,site[2].totNL:4,site[2].totPV:4);
    writeln(out,site[3].totLSP:4,site[3].totNL:4,site[3].totPV:4);   }
    writeln(out,site[1].attendance:6,site[2].attendance:6,
        site[3].attendance:6);
    writeln(out,site[1].quality:12:1,site[2].quality:12:1,
        site[3].quality:12:1);
    writeln(out, site[1].addSE:6:1, site[2].addSE:6:1,
        site[3].addSE:6:1);
    for i := 1 to 3 do begin
        for j := 1 to 8 do write(out,gensite[i,j]:6);
        writeln(out);
        end;
    end;
if screenupdate then
    if time/(4*printyear) = trunc(time/(4*printyear)) then
        writeln(time:4,'  |', freqLSP:8:4, freqNL:8:4,
        freqPV:8:4, popsize:5);

                          { eat }

for i := 1 to popsize do body[indv[i]].SE := body[indv[i]].SE +
  site[body[indv[i]].where].addSE - maintcost;
for i := 1 to popsize do if body[indv[i]].moved then body[indv[i]].SE :=
    body[indv[i]].SE - move;
if time = kick then for i := 1 to trunc(popsize/10) do body[indv[i]].SE := body[indv[i]].SE + 6000;

                          { die }
```

```
    for i := 1 to (popsize-dead) do begin
            if popsize <= 2 then extinct := true;
            whom  := indv[i];
            probdeath := exp(coeff*body[whom].SE);
            justanumber := unirand(seed,seed);
            if justanumber < probdeath then body[whom].alive := false;
            if body[whom].age >= 5000 then body[whom].alive := false;
            if not body[whom].alive and
               (dead < popsize - 2) then begin
               body[indv[i]] := body[indv[popsize]];
               dead := dead + 1;
               end;
            end;
     popsize := popsize - dead;
  {   write(dead:5);  }
                            { have babies }

    for i := 1 to popsize do begin
        whom := indv[i];
        if (body[whom].age/refractory = trunc(body[whom].age/refractory)) and (body[whom].alive) then
begin
        expectedkids := 1 - exp(coeff*(body[whom].SE - threshold));
            justanumber := unirand(seed,seed);
            if (justanumber < expectedkids) then begin
               body[whom].SE := body[whom].SE - babycost;
               babies := babies + 1;
               kid := indv[popsize + babies + 1];
               body[kid].where := body[whom].where;
               body[kid].speriod := body[whom].speriod;
               body[kid].nonlinear := body[whom].nonlinear;
               body[kid].persvalue := body[whom].persvalue;
               with body[kid] do begin
                    SE := normrand(startfood,startfood/2,seed,seed);
                    estimateSE := SE;
                    for j:=1 to 3 do begin
                        estimateQ[j] := qual[j];{site[j].quality;}
                        estimateN[j] := popsize/3;
                        end;
                    alive := true;
                    howlong := 0;
                    count := 0;
                    age := 0;
                    end;
                end;
            end;
        end;
    popsize := popsize + babies;
{    writeln(babies:5);  }
    if extinct then begin
       writeln(out,'Population crashed.');
       time := totaltime + 1;
       end;
    END;                                          {time loop}
    close(out);
  { END;         run loop }
   { close(error); }
END.
```