AgModel: An ODD Protocol

Isaac I. Ullah[1,2]

[1]San Diego State University, [2] Corresponding author (email: iullah@sdsu.edu)

Purpose and Patterns

AgModel is a landscape-scale Agent-Based Model (ABM) of human foraging that incorporates co-evolutionary feedback between humans and cereal plants. ABM is a modeling formalism that instantiates independent software agents that can interact according to a set of encoded rules in a virtual world, and has been applied in simulation of both contemporary and ancient Social Ecological Systems (SES) (Bonabeau 2002; Romanowska et al. 2021; Rounsevell et al. 2012). This formalism sets AgModel apart from other simulation or mathematical models of the forager-farmer transition. For example, Bowles (2011) employed a comparative statistical accounting model to compare the productivity of foragers and farmers across the transition, Freeman et al. (2015) employ optimality logic in a set of chained dynamical equations to explore the stability of different wild and domestic economies in different environments, and Agourakis et al. (2022) use a dynamical systems modeling approach to understanding human-plant co-evolutionary dynamics. Careful parameterization of AgModel can test scenarios where complex hunter-gatherers engaged in foraging decisions that interacted with plant evolutionary dynamics to artificially increase the prevalence of domestic type plans in the local ecosystem. When the domestic phenotype is most prevalent, the species can be considered to be "domesticated." The temporal dynamics of the process of domestication in AgModel are not likely to be linear, however, and the timeline and sequence of the domestication event may differ under changing environmental conditions or even between repetitions of runs starting with the same initial conditions.

Entities, state variables, and scales

AgModel consists of three types of entities. The first is a software agent that conceptually represents a single hunter-gather community (i.e., a co-residential group that shares in subsistence activities and decision making). The second is an agent that conceptually represents a prey animal population in the surrounding landscape. The final entity is a set of agents that conceptually represent individual patches making up a population of cereal plants in that same landscape.

The forager agent has several state variables, including a running population total, intrinsic birth and death rates, and an annual total energy need (measured in kcal), and an available amount of foraging labor. The cereal patch and prey animal agents have several state variables, such as population totals, density, search costs, and consumption values. Table 1 lists all the state variables for each agent class, their default values, and a brief explanation of what the variable represents.

Table 1. AgModel variables, default values, and explanations.

Variable	Default Value	Description
People	50	Initial number of people in the forager band
MaximumPeople	3000	Maximum human population
HumanBirthRate	0.032	Human birth rate (% per annum)
HumanDeathRate	0.03	Human death rate (% per annum)
HumanBirthDeathFilter	0.005	Width of the gaussian randomizing filter for human birth and death rates (% per annum)
StarvationThreshold	0.8	Human starvation threshold (% of per annum kcal need)
HumanKcal	912500	Yearly per capita kcal need
ForagingHours	4380	Per capita hours of work available per annum
ForagingUncertainty	0.1	Width of the gaussian randomizing filter for foraging returns (%)
Prey	200	Initial number of prey
MaxPrey	500	Maximum number of prey
PreyMigrants	0	Number of prey that migrate into the territory per annum
PreyBirthRate	0.06	Prey birth rate(% per annum)
PreyDeathRate	0.04	Intrinsic prey death rate (% per annum, without human hunting)
PreyBirthDeathFilter	0.005	Width of the gaussian randomizing filter for prey birth and death rates (% per annum)
PreyReturns	200000	Kcal return from an average prey
PreySearchCost	72.0	Hours search time to find a prey
PreyDensity	1000	Density for which the search time is known
MaxPreyEncountered	4	Maximum number of prey per encounter
MinPreyEncountered	1	Minimum number of prey per encounter
PreyHandlingCost	16	Hours of handling time per encounter
Cereal	100	Number of cereal patches (assume 1ha/patch)
WildCerealReturns	0.05	Return rate (kcal) per wild-type cereal seed
DomesticatedCereal Returns	0.1	Return rate (kcal) per domestic-type cereal seed
WildToDomesticated Proportion	0.98	Initial proportion of wild to domestic types
CerealSelectionRate	0.03	Selection coefficient (% per annum)
CerealDiffusionRate	0.02	Diffusion coefficient (% per annum)
SelectionDiffusionFilter	0.001	Width of the gaussian filter applied to selection and diffusion rate (% per annum)
CerealSearchCosts	1.0	Hours of search time to find a cereal patch
CerealDensity	10000000	Initial (and minimum) number of individual cereal plants per patch
MaxCerealDensity	100000000	Maximum number of individual cereal plants per patch, after human influence
CerealCultivation Density	1000000	Number of individual cereal plants that cultivation increases in a patch per annum
WildCerealHandling Cost	0.0001	Handling time (hours) per wild cereal seed

DomesticatedCereal HandlingCost	0.00001	Handling time (hours) per domestic cereal seed
Years	3000	The number of years for which to run the simulation

The model does not have an intrinsic spatial scale, but assumes a logistic, central-place foraging strategy (Kelly 1995; Kennett and Winterhalder 2006) in a fixed territory for a two-resource economy of prey animals and/or cereals grains. The territory has a fixed number of cereal patches, and a starting number of prey animals.

Process overview and scheduling

AgModel is conceptualized as proceeding at an annual time interval, with many individual foraging decisions occuring within each time step. Foraging decisions in AgModel are "event" based so that many such decisions will be made in round-robin fashion each year. After each decision, the outcomes are fed back into the general conditions of the model such that prey animal population numbers and remaining cereal patch availability are updated within each annual period. These feedbacks can alter the balance of the DBM, so it is possible for the forager agent to switch between hunting prey animals and harvesting cereals multiple times within or between years. Intra-annual foraging decisions continue until the current annual energy target is reached, the prey and/or cereal population is depleted, or until the annual labor cap is reached.

Design concepts

Basic principles

The human foraging decision component of AgModel takes inspiration from Optimal Foraging Theory (OFT) (Belovsky 1988; Kennett and Winterhalder 2006; Winterhalder et al. 1988). Classical OFT logic assumes the forager to perfectly know the tradeoffs for all foraging choices, to rationally evaluate them, and to never deviate from this decision logic (Kelly 1995). While problematic, this provides a baseline for studying human foraging using straightforward and well-understood logic (Codding and Bird 2015; Gremillion et al. 2014). Models are always simplifications of reality (Box and Draper 1987), but the limitations encountered when distilling a simple model from reality can be useful when model construction is cognizant of those limitations (Evans et al. 2013; Klosterman 2012). AgModel therefore capitalizes on the ABM formalism to ease the strict assumptions of rationality and perfect knowledge of traditional OFT.

In classical OFT, the profitability, P_i , of a potential food item *i* is described as the food energy, E_i, gained (e.g., in kcal) per unit time, h_i, spent “handling” the prey once encountered (Mithen 1988; Smith 1983):

$$P_i = \frac{E_i}{h_i}$$

(1)

The handling time, h_i, includes the time to pursue, kill/harvest, transport, process, cook, and eat the item. When two potential items, *i* and *j*, are encountered simultaneously, a Prey Choice Model (PCM) can be established by:

if: $P_i > P_j$, pursue/process item *i*
else: pursue/process item *j*

(2)

When items *i* and *j* have equal payoffs, the choice is made randomly. When all potential food items are compared iteratively, they are then ranked in descending order of profitability. To choose between more than two food items, or if the food items are not encountered simultaneously, it becomes necessary to establish a Diet Breadth Model (DBM). Here, the profitability of all potential food items must be known, as well as their average search costs, *S*, which is the search time required before the item is encountered (Foley 1985; Hames and Vickers 1982; Smith 1983). To be included in the diet, the profitability of food item *k* must exceed or equal the average potential profitability of all *n* other higher-ranked food items (*i* to *j*) when their search costs are factored in:

if: $P_k \geq \sum_{E_i}^{E_j} \div n$, include item *k* in diet

(3)

For a central place foraging model, like that implemented in AgModel, we can assume that foragers may make foraging decisions *before* departing on forays, and so the search for any food items should be included in the profitability calculations, such that:

$$P_i = ¿ \frac{E_i}{h_i + S_i}$$

(4)

For a two-resource economy, such as that in AgModel, the central-place DBM is:

if: $\frac{E_j}{h_j + S_j} \geq \frac{E_i}{h_i + S_i}$, include item *j* in diet

(5)

Note that this is essentially the same as a PCM when search costs are also included, and that is how the decision algorithm in AgModel is coded.

Emergence

The main capacity for emergence in AgModel arises through the interaction of feedbacks and stochastic variables to affect the temporal trajectory of domestication. The capacity for emergence is sensitive to the combination of values of the input variables in that some combinations will always lead quickly to domestication, and other will always never lead to domestication, but still others will create variability across repetitions in the timing of the onset of domestication and/or cycling/hysteresis in domestication and reversion back to foraging.

Adaptation

AgModel allocates a finite annual per capita labor budget against which accrued search and processing costs are balanced to set a limit on the number of foraging activities that can be done in a year. The forager agent calculates its annual labor budget by multiplying its current population by a per capita labor availability set at model initialization. If the agent chooses to hunt prey, it will expend time from their labor supply searching for and processing prey animals. If the agent chooses to harvest cereals, then they expend time from their labor supply searching for a cereal patch and processing the cereal grains in that patch. AgModel employs a “satisficing” strategy of achieving the minimally successful outcome, and no more (Ward 1992), so the agent cannot spend excess labor towards accumulation of a surplus that could roll over to a subsequent year.

Objectives

AgModel was designed to study how the dynamics of human foraging decisions can feedback into human-domesticate coevolutionary processes to affect the probability of a domestication event under difference scenarios of population growth.

Learning

The forager agent learns from past returns, but does not have perfect knowledge of current conditions. More details of the forager agent learning are described in the “Stochasticity” section, below.

Prediction

The model simulates long-term human-domesticate coevolutionary dynamics, and in a general sense provides output that can be used to interpret archaeological and paleoecological evidence from real world case studies. In short, the model is designed to be a hypothesis generating device derived from theory about human foraging behavior and evolutionary processes.

Sensing

The forager agent senses prey animal and cereal patch agents in that it is aware of the general number and estimated average productivity of each within their territory.

Interaction

The forager agent kills prey animals and harvests and reseeds cereal patches. Human harvesting impacts reseeding rate and selection for domestic phenotype individuals in harvested patches, so that the proportion of domestic type increases in regularly harvested patches over time. Cereal patches interact by cross pollination that can reintroduce wild phenotypes once humans cease to reseed.

In AgModel, the rate of human selection for domestic-type plants, r_s, and the rate of introgression of wild traits, r_d, are set at model initialization and interact to produce co-evolutionary dynamics over time. Both rates are subjected to an annual gaussian filter that is similar to those represented in Equations 9, 10, and 11 (see *Stochasticity*, below). The selection rate is the percentage per year at which human harvesting and processing behaviors alter the ratio of wild-type to domestic-type cereals, $R_{w:d}$, in utilized patches, up to the maximum ratio of 1.0. At the same time, there is diffusion of wild type traits in the harvested patches, so the ratio of wild to domestic types is updated according to the following formula:

$$R_{w:d} = R_{w:d} + r_s - r_d \tag{6}$$

Evolutionary changes only occur if human selective pressure is greater than the background diffusion rate, and the ratio of wild to domestic plants will reduce in the harvested patch. If humans do not harvest a patch, then r_s is 0.0, and diffusion of wild type traits will begin to increase the ratio of wild to domestic types in a previously used patch. There are many physical traits that are related to cereal yield, caloric payoff per kernel, and handling times that can change during domestication, including individual seed mass, number of kernels per seed head (infructescence), and number of seed heads per plant (Preece et al. 2017). Other characteristics can increase the overall yields by making processing easier or more efficient, such as non-shattering seed attachments, thinner, fewer, or less tenacious glumes/hulls, alterations to the structure of the seed head, and increased seed viability (Abbo et al. 2014; Preece et al. 2017). Changes in human husbandry behaviors such as tilling, weeding, sowing, and irrigating can also increase overall yield per unit area and further reduce handling times (Allaby et al. 2021; Spengler 2020). AgModel abstracts all of these changes into three areas: caloric payoffs, handling times, and patch density. Wild-type cereals are parameterized with one set of caloric payoffs (e_w kcal/kernel) and handling time (h_w hrs/kernel), and domestic-type cereals can be parameterized with another set of payoffs (e_d kcal/kernel) and handling times (h_d hrs/kernel). These values are balanced for the population of cereals according to the ratio of wild to domestic types in a patch (Equation 12) such that the total energetic payoff (kcal/hr) of a particular patch, E_i, is calculated as:

$$E_i = \text{¿} \frac{\left(\left(R_{w:d} * D\right) * e_w\right) + \left(\left(1 - R_{w:d} * D\right) * e_d\right)}{\left(\left(R_{w:d} * D\right) * h_w\right) + \left(\left(1 - R_{w:d} * D\right) * h_d\right) + S} \tag{7}$$

Here, D is the total patch density (kernels/patch) and S is the search time to find a patch of cereals. D changes from a lower initial value (i.e. a “wild state”) to a theoretical maximum value (i.e. a “fully cultivated state”). Density changes according to a cultivation factor (d_c kernels/patch) that is added each time a patch is harvested, and removed each time it is left unharvested:

$$D_{t2} = D_{t1} \mp d_c \tag{8}$$

Thus, the total payoff of harvesting any particular patch of cereals is a function of the ratio of wild to domestic plants and the amount of time that humans have been managing that patch (or not). The model assumes that the agent consistently chooses cereal patches in the same order, which allows for changes to accumulate in patches even if cereals only make up a portion of the annual subsistence plans. Each patch will develop a unique handling time and caloric return rate over time according to the balance of evolutionary dynamics of human plant selection in that patch and consistently harvested patches can become dominated by the domestic phenotype.

Stochasticity

AgModel employs stochasticity in a few of it’s core algorithms. Firstly, stochasticity is incorporated in the foraging decision algorithm to allow for a *potentially fallible* agent to make a series of choices over time even when the agent does *not* have perfect knowledge about returns or costs. To this end, AgModel incorporates flawed knowledge and the potential for mistakes by stochastically perturbing the agent’s knowledge of food item profitabilities at each decision point via a gaussian random function that draws values from a probability density function where the calculated profitability P_i of food item i is the mean of a normal distribution with standard deviation equal to some percentage w of the mean:

$$P_i = f(x \mid \mu, \sigma^2) \text{ where: } \mu = P_i \text{ and: } \sigma = P_i * w \quad (9)$$

There is therefore only a small and finite probability of the agent exactly estimating the numerically correct average profitability of food item *i*., but will be within *w* percent of the actual average profitability 68% of the time, within 2*w* 95% of the time, and within 3*w* 99.7% of the time.

Another stochastic component of the foraging model affects the encounter rate of prey animals. The actual number of individual prey animals that are/would be encountered in each foraging decision point is drawn from a uniform random distribution between the lower and upper boundary that are specified at the start of the model run.

AgModel exposes intrinsic per capita birth and death rates for humans and prey animals as variables in the modeling interface. Although these rates are entered as average birth and death rates, the demographic modeling engine introduces variability in year-to-year population changes in two important ways. The first is through a gaussian random filter that is applied each year when births and intrinsic (natural) deaths are calculated. The filter takes the form of:

$$n_b = f(x \mid \mu, \sigma^2) * p_{t1}, \text{ where: } \mu = r_b \text{ and: } \sigma = w \quad (10)$$

$$n_d = f(x \mid \mu, \sigma^2) * p_{t1}, \text{ where: } \mu = r_d \text{ and: } \sigma = w \quad (11)$$

Here, n_b and n_d are the number of individuals that are born and died in a given year, r_b and r_d are the intrinsic birth and death rate as entered at the start of the simulation, *w* is the standard deviation (width) of the gaussian distribution used in the demographic filter, and p_{t1} is the population at the current timestep. Although the birth and death rates stay fixed during a model run, this stochastic filter ensures that the actual number of births and deaths can vary by a small amount from year to year. Births and deaths are balanced each time step to determine the rate of population growth or decline over time:

$$p_{t2} = p_{t1} + n_b - n_d \quad (12)$$

Here, pt_1 is the population size at time one, p_{t2} is the population size at time two, n_b is the number of births from Equation 7, and n_d is the number of natural deaths from Equation 8. An optional maximum population size limit can cap growth at a specific number (e.g., to account for extrinsic forces not included in AgModel). If the forager agent depletes all food resources or expends all labor, it will not achieve its caloric target for a year. If this deficit is greater than a starvation threshold as set at model initialization, then a negative demographic feedback loop is enabled:

$$p_{t2} = p_{t1} - (2n_d) \tag{13}$$

As the simulation proceeds, Equations 9 and 10 work to balance births and deaths in response to subsistence feedback so that it is possible to achieve dynamic equilibrium, hysteresis, or collapse.

Prey demography is also governed by intrinsic annual per capita birth and death rates with stochastically introduced variability, with the addition of n_h deaths caused by human predation and a random number of up to n_m new prey animals that may "migrate" into the territory each year. Prey demography is therefore annually updated as:

$$p_{t2} = p_{t1} + n_b + n_m - n_d - n_h \tag{14}$$

Collectives

AgModel simulates farmers as a collective "village" of farmers. Prey animals are simulated as collective population within the site catchment area. Individual cereal plants are collected as patches, and there is a fixed number of patches within the site catchment areas.

Observation

Summary statistics are exported in the csv file format in three files. A "general stats" file includes annual values for the following variables: Total Human Population, Human Kcal Deficit, Total Prey Animals Population, Number of Prey Animals Eaten, Total Cereal Population (*10^3), Number of Cereal Patches Exploited, Proportion of Domestic-Type Cereal, Average Cereal Patch Density (*10^3). A "cereal patch density stats" file includes annual values for the density of cereal plants in each patch, and a "cereal patch domestic proportion stats" file includes annual values for the proportion of domestic type cereals in each patch. The graphical interface includes the ability to display real-time plots of these values.

Initialization

AgModel is written in pure Python 3 with the hope of better integration to scientific Python (e.g., pandas, matplotlib) and the open-science movement. The model has a simple graphical interface, or can be run "headless." AgModel requires Python 3 and the following Python modules: NumPy, Pandas, MatPlotLib, seaborn, and EasyGUI. Once all dependencies are installed, AgModel is launched from the commandline. If the graphical interface version of the model is initiated, the graphical interface will deploy to guide selection of variable values. Optionally values can be loaded and saved to a configuration file. Once parameters are set, the model can be launched and an optional plotting canvas window will display plots of human, prey animal, and cereal plant populations, the number of harvested prey animals and cereals, and the proportion of domestic type cereal plants in the total population. At the end of simulation, stats files and plots can be saved. The headless version requires a configuration file to be specified on the command line at launch, and will run with no user interaction, saving stats files for later analysis. Optionally, experiments using parameter sweeps with repetition can be set up to run in parallel on as many available processors as desired using the included helper script "*parallelizer.py*" with the headless version of the model.

Input data

All together, AgModel exposes 35 variables in the setup interface that can be customized to create unique modeling experiments. These variables, their default values, and brief descriptions are listed in Table 1 (above). Where possible, AgModel uses "real world" values instead of abstracted numbers in an effort to help align modeling experiments with scientific literature about foraging and early farming behavior, cereal and prey species ecology, physiognomy, and behavior, and with agronomic and dietary data. This approach makes it simpler to derive numerical values from the literature or from experimental or observational data, but at the expense of making it more difficult to balance feedback and values in any particular experiment. Further, users should be cognizant that the use of "real world" values in this model does not necessarily mean that the specific model outputs are directly translatable to the archaeological record or to other real world cases. The model is still a simplified and abstracted representation of real world processes, and to ensure validity of experimental results, it is important to carefully balance the input values, and to ensure that the values are all within reasonable ranges of real-world cases using empirical data sources. In particular, the numerical values that are used in the foraging decision logic of the model need to be within a reasonable range of the "real-world" values that are expected by the OFT equations that are employed (see *Basic Principles* and *Interaction*, above).

Submodels

There are no submodels contained within the AgModel codebase, but there are two custom functions related to determining the number of births and deaths, respectively, in human and prey animal populations. See equations 10, 11 and 12, above.