Natural Selection Process for Animals

Davide Monaco Mat. 305133 email: s305133@studenti.polito.it

January 23, 2024

Abstract: This report explains a simulator for wild animals, to discover which features bring a specific specie towards its proliferation. To reach this aim, the simulator is able to manage the events of birth, death , fight and migration among animals.

1 Introduction

The first implementation to build a simulator that mimics the wild life, is the introduction of a ZOO dataset. The dataset contains information for mostly wild animals and specific features about them. This characteristics are important to build a Specie object that has inside parameters of its mobility and lethality. The Specie object also contains information of its lifetime to generate the born events for its child, and also the death time to create the death event. The object Specie is an animal, and during its lifetime is able to move in a map of $1000\,\mathrm{km}^2$. The aquatic animals in the dataset are drop because the map is set for only ground and volatile animals, further future implementation can be used to model each cell of the map to recreate environment for specific needs of the animal. The lifetime of the animals for the first inizialization of birth time is an Uniform distribution $U \sim [10 - 20]$, representing its lifetime in years.

1.1 Specie object

The Specie object inglobes all attributes to build an animal. The first is the attribute "*name*" that contains the typology of the animal. The attribute "*id*" is used to label a unique object inside the simulator. The "*generation*", "*timeborn*" and "*death*" are the parameters of the lifetime and the generation for which it belongs. This parameters are important to create the "*BORN*" and "*DEATH*" event in the simulator. Whenever the simulator matches a "*BORN*" event, the attributes of the "*parentlife*" and the object "*father*", related to its ancestor, are used to estimate "*death*" of that specific animal. The attribute "*position*" indicates the position of the animal in the map, for the first initialization the simulator takes the coordinate (x, y) using an Uniform distribution $U \sim [0 - 10000]$, indicating the position in meters for the map. The attributes "*mobility*" , "*velocity*" , "*lethality*" are explained in details in the next paragraph, to understand the main hypothesis behind them.

1.2 Assumptions for Mobility and Lethality

The mobility and lethality parameters for a specific animal are defined using some features from the ZOO dataset, described in Table 1 and 2. When an object Specie is created then the attributes of mobility and lethality are inizialized for each different animal and assigned a percentage created by the function "*mobility*", it takes the name of the animal and assigns a percentage. The mobility has 4 characteristics, for each of them is defined a percentage based on assumptions. A general animal has different numbers of legs, infact the simulator assigns the main weight using this charateristic. An animal with 0 and 8 legs are defined as slow, because in nature they probably can be molluscs or arachnids, instead for animals with "6" legs insects, and for "2" and "4" the simulator categories as biped and quadruped animals. The same concept is applied for catsize, backbone and airbone. The velocity depends on the size of the animal, a small size corresponds to a lower velocity. If the animal has a backbone the bone structure is solid, in opposite the animal is soggy. The animals with airborne are classified as volatile and in general have an higher velocity and agility. The percentage of the mobility is multiply by 10 km and the result used to compute the maximum velocity, corresponding to the distance travelled per year that an animal reach. When the animal migrates in a new position, it will move among 0 and the maximum distance of travelling. The same idea of the mobility percentage is used to implement a function for the lethality of the animal. It is used when 2 animals are close and they fight for survive. The main contribute is given by the animal that is a predator, by nature this animal is an hunter and dangerous than a domestic one. Toothed and venomous animals increment the lethality of them.

1.3 Measure Object

The Measure object is used to collect data of the birth and death animal for a specific Specie object. This object is used to figure out what is the survival rate of a specific animal. This measure is compute in the

Table 1: *Mobility features.*

legs	catsize	backbone	airborne
0: 0.2 2: 0.5 4: 0.8 6: 0.9 8: 0.1	1: -0.30 0: 0	0: -0.1 1: 0.1	0: 0 1: 0.3

Table 2: *Lethality features.*

predator	toothed	venomous	domestic
0: 0.20 1 : 0.60	0: 0 1: 0.2	0: 0 1: 0.3	0: 0 1: -0.1

simulation to compare the rate of the born and death animal toward a simple counter.

2 Methods

The simulator uses different functions to create or remove animals using the born and death event, or to create an interaction or a fight among animals. Each event generated by the functions are append in a Future Event Set (FES), to maintain trace of future event, instead the animal created by the functions are added in a population variable that contain the list of animal in the simulator.

2.1 Random animal

The Randomanimal() function is used to in the simulator to choice 5 random animals inside the dataset, and create for each of them the Specie object.

2.2 Born

The Born() function is used to create in a specific time the Specie object. This function uses the information of the father's object Specie, to estimate the lifetime of the animal born. In the same time, a Poisson distribution generates a random number used to create the future child. The lifetime of these future child can be: better of the father lifetime, following the equation **1**; or having a worst child lifetime, following the equation **2**. For both case, the features of mobility and lethality can improve of 1%.

1 $ChildLT(t) = \text{Uniform}\,(LF(t), LF(t) \cdot (1 + \alpha))$
with prob. improvement p

2 $ChildLT(t) = \text{Uniform}(0, LF(t))$
with prob. improvement 1-p

the estimation time of birth and death are used to create future events for the FES.

2.3 Death

The Death() function simply finds the death animal in the population and removes it.

2.4 Movement

The Movement() function is used to move the animal in a new position in the map, the function compute the new position of the animal using its velocity. On the coordinates (x, y), is added or subtract the distance choosing randomly with a Uniform $U \sim [0 - max(velocity)]$. The function also checks that the animals don't exceed the border of the map, If the animal exceeds the border in one axis or both, the movement for that specific axis is channeled towards internal of the map.

2.5 Interaction

The Interaction() function uses the position of each animal in the population to compute the Euclidean distance on them. The interaction happens if the Euclidean distance among 2 animals is less than 500 m, saving their interaction. In the same time is applied fight_escape() function to interact 2 close animals for fighting.

2.6 Fight & Escape

The Fight_escape() function is applied on 2 animals to recreate a fight. The animal has 2 options, fight or escape from another one. This idea reproduces the escape-attack behaviour that each animal has inside its reptilian brain, using the theory of triune brain model. The interaction of 2 animals has 4 options described in the Table 3,

Animal X	Animal Y	Result
Attack	Escape	Attack/Escape
Attack	Attack	Attack
Escape	Attack	Attack/Escape
Escape	Escape	Escape

Table 3: *Results of interaction among 2 animals.*

When animals fight, if they are able to escape, then its mobility attribute is decreased of 1%, otherwise are compered the lethality attributes to decree the survivor. If two animals are bound to fight, one animal die, and the survivor increases its lethality of 1%. For the last case, if both escape, then their mobility attributes are decreased of 1%.

2.7 Simulation

The Simulation() function runs the entire simulation. The simulation time is set to a fixed number of seconds, the reason of this choice is that the behaviour of the population is an exponential, the population tends to grow for infinite time. Using as threshold, an internal timer produces some issues about the termination loop, for which it can never be reach due to an explosion of born event in a small amount of time. The simulation

Table 4: *Features of 5 animals.*

Animal	Lethality	Mobility
	[predator - toothed venomous - domestic]	[legs - catsize backbone - airborne]
buffalo	[0 - 1 - 0 - 0]	[4 - 1 - 1 - 0]
lynx	[1 - 1 - 0 - 0]	[4 - 1 - 1 - 0]
flea	[0 - 0 - 0 - 0]	[6 - 0 - 1 - 0]
chicken	[0 - 0 - 0 - 0]	[2 - 0- 1 - 1]
mongoose	[1 - 1 - 0 - 0]	[4 - 1- 1 - 0]

reach its saturation in few seconds. The result of the explosion of born is strictly correlated with the equation **2** and the first initialization of the lifetime of the 5 random animals. The function extracts each time an event from the FES, and whenever the internal timer reach a new year, then is applied the functions movement and interaction.

3 Results

The following simulation are conducted using different values of improve probability [0.4 – 0.5], and the mean reproduction rate [2 – 3] imployed in the Poisson distribution to generate the random number of child. For each couple of parameters the simulator runs 30 time. The simulation uses the same 5 random animals generated by previously function described, and is computed the average survival rate, its confidence interval and its accuracy. The time window is 20 s. The graphs show that the buffalo has an higher survival rate than other animals. This result can be drive from the assumption that it has an higher value of lethality, instead little and quick animals like flea, can use its features to escape from the fight. Another important consideration is given by the value of the reproduction rate, infact in the case of lower reproduction rate and improve probability, showed in the first graph, the accuracy of all animals is lower than others, probably given by sparse survival rate of each animal. The survival rate for an higher improve probability makes the results more stable. In the following Table **4** are described the features. The lynx and mongoose share the same features, infact its values are similar but this result can be alterate from the migration of the animals. Instead, the chicken and flea can benefit on good mobility features to escape by the combat. In conclusion, the simulator affirms that every time it can produce vary results, conducted by different inizialization for the first position, and the typology of the animal.