Abstract

The aim of this master thesis is the development of a multiagent model for a simulation of two populations whose interactions are strongly influenced by a realistic landscape.
This research will be in line with Consolider-Simulpast (www.simulpast.es), an interdisciplinary project aimed to create simulations designed to be used in archaeological studies of human-environment interaction, decision-making processes and coevolutionary/competition behaviours of past societies.
The work plan will be focused on the development of first-stage models for two societies in the age of agriculture spreading surpassing the hunting and foraging way of living. The simulation will involve a climate engine for seasonality depending primarily on variable rainfall rate. Landscape information will be created from satellite image rasters. Constants, and variable relationship shall be modelled from measures and interviews with the experts. Data analysis tasks will be undertaken to validate the models and detect patterns in the archaeological record. Furthermore a comparison will be established between the classical simple models used in social simulation[1][2][8] and more advanced approaches.

Contents

Chapter 1

Introduction

1.1 Description

1.2 Motivation

Simulation has been following an evolution in the models and paradigms applied to represent its target systems. Dynamical systems, differential equations have been used and an overall simplification of the parts of the systems in the history of simulation to operate with the abstraction and simplification of the problems. Mainly, reusing ideas from physical simulations, social sciences has modelled complex systems with dynamical atomic entities that apply a simple set of centralized rules to move around the environment modelled. It was seen that due to the deeper details in human behaviour the question that could be solved and asked to that kind of models could not go very further as expected.

To solve non-linearity phenomena, heterogeneity, hysteresis[51, p.571–597] and other issues typical of complex systems, Agent Based Models(ABMs) where introduced to gain more insight of the modelled systems achieving good results. But under some conditions of very complex relationships between agents, highly specialized decision taking procedures and issues in the environment led agents to need a sophisticated reasoning and problem solving capabilities that are not specified and not yet introduced in ABMs coming from Social Sciences.

We have found an example of that situation in our case study located in Gujarat from Simulpast project. Our agents must interact with a regular but changing environment to get resources, plan its actions, coordinate with its group and compete with other groups in a **co-evolution** dynamic. Because we want to find out why the Gujarat Hunter-gatherer(HG) way of life lasted more than in other place of Earth in its competition against Agro-pastoralist(AP), we need to embed the behaviours of the survival strategies used by these groups. A simply reactive agent cannot cope with short term plus long term decisions in that competitive environment. So the question stated as topic for this Master Thesis project is **"do we get better results in social sciences simulations adding deeper AI techniques to make richer the**

behaviour or decision making engines of the entities in the system?". As we understand "better results", the outcome from the use of AI should be a more sound validation of the model, a nearer match of modelled behaviours with the real ones, and clearer, richer and robust scientific conclusions.

In order to study such possibilities Sugarscape is a good framework to extend. Sugarscape is an artificial society developed by Joshua Epstein et al [3]. where a number of inhabitants move to collect resources they need to live. Sugarscape models perception, lattice scanning in search of resources, sexual reproduction of the agents in the simulation, market relationships, immunology and spreading of diseases, and feature evolution. Epstein analyses different experiments executed in the Sugarscape offering his conclusions and the dynamics emerging from the simulations. The results and conclusions will feed our AI experiments in order to make the comparisons of classical agents vs AI agents, therefore, giving an answer to one of the topics of the Master Project.

1.3 Simulation

Simulation is a discipline for performing virtual experiments in a computer. Computational techniques are used to build a model that represents your system. The dynamics of that system is codified in an algorithm that computes a calculus imitating the changes of state in the model, hence having a representation of that change along the time of the system modelled. Simulate is to play to "what happens if...?", and it is aimed to discover and explain the dynamics of a system to enhance or guide strategy development, decision taking, management, solving problems without analytical solutions or knowledge discovering and research. Although, we could get other positive benefits from it like theory checking or training through the immersion in virtual worlds responding to our input.
A simulation obeys some direction of experimentation, so a question must be set to drive the selection of features to model from the real system and give a direction to the modelling and the experiments design.These assumptions choice will prune the details not related to the questions to solve. It is not just for the sake of simplicity but for the practical reason that a model too near of the real system will be as hard as the original system to analyse. Simulation, like deduction, starts with a set of those explicit assumptions. But unlike deduction, it does not prove theorems. Instead, a simulation generates data that can be analysed inductively. Unlike typical induction, however, the simulated data comes from a rigorously specified artificial experiment rather than direct measurement of the real world. While induction can be used to find patterns in data, and deduction can be used to find consequences of assumptions, simulation modelling can be used as an aid intuition and hypothesis validation tool. Also as space search mechanism for parameter tuning or optimization. This links with abductive processes.
Just like in a typical Sherlock Holmes case you pick the evidences, scenario for the experiment, and the common knowledge (the initial expert assumptions about

the model). You enter in a refinement cycle where you test hypothesis and readjust them to discover the theory, the "plot", the explanation of what is happening. Following the abductive reasoning schema one looks for the hypothesis that would best explain the relevant evidence [5].

Social simulation is a research field that applies computational methods to study issues in the social sciences. The issues explored include problems in psychology, sociology, political science, economics, anthropology, geography, archaeology and linguistics [46].
Social simulation aims to cross the gap between the descriptive approach used in the social sciences and the formal approach used in the hard sciences, by moving the focus on the processes/mechanisms/behaviours that build the social reality.
In social simulation, computers supports human reasoning activities by executing these mechanisms. This field explores the simulation of societies as complex non-linear systems, which are difficult to study with classical mathematical equation-based models. Most of the times, studying complex systems implies to cope with non reducibility. One of the examples is Gravitational Dynamics. If our assumption is the use of Newton's mechanics, we can predict the state at any time or not, depending on the scenario. For a one dimension world you can predict the state at time t_n from the initial state t_0 without computing all the preceding ones. For two and more dimensions you can only compute directly state t_n if less the three bodies are implied. So in a real environment of many bodies in a 3D world you need to compute all the states from the initial to the one you consider as the last one. The system is non analytically reducible and you are forced to apply simulation to visit all the states and develop the behaviour of the model. It happens in most of the complex systems models, they have a non-linear specification. Non-linear models do not have a simple or computational reasonable analytical solution.

Other of the main issues in complex systems simulation is emergence. While the initial assumptions may be simple, the consequences may not be at all obvious. The large-scale effects of locally interacting entities are called "emergent properties" of the system. Emergent properties are often surprising because it can be hard to anticipate the full consequences of even simple forms of interaction.

There are some models, however, in which emergent properties can be formally deduced. Good examples include the neo-classical economic models in which rational agents operating under powerful assumptions about the availability of information and the capability to optimize can achieve an efficient reallocation of resources among themselves through costless trading. But when the agents use **adaptive** rather than optimizing strategies, deducing the consequences is often impossible; simulation becomes necessary.

1.4 Question

In classical simulation approaches, specifically in the branch of Social Simulation, active entities which model human actors are designed with very simple behaviour engines. The classical hypothesis is that a complex mind for entities in the simulation are not that needed and maybe even could lead to difficult analysis of final results of the simulations (too daring statement?).

Our statement is that, on the contrary, the mind engine of a simulation entity should not be bounded to that limit but special attention must be paid to give any necessary sophistication to give the entity a correct behaviour, real enough, sensible to the changes in the environment and competent to solve the issues that will have to solve along its lifetime in the system. Even more, we think that this entities' capability to respond with complex behaviours is the core that roots the modelling granularity needed to catch the essential of the social systems that we want to model. Applying such premises we will explore the possibility to give or enhance decision making, problem solving capabilities to the entities with the aim to get more accurate simulations and realistic models with higher matching against our job hypothesis and premises. We will take the framework of ABMs to integrate the AI techniques in a decision making schema of action-response dynamics sensible to a modelled world.

1. **Do AI techniques contribute to better simulation results?**
2. **Classic Simple Agent approach vs Rich Agents**
3. **Did Gujarat extreme environmental conditions delayed the HG disappearance?**

Chapter 2

Methodology

2.1 Intro

Modelling is a widely extended methodology to answer the kind of questions we set out. It comes from the natural observation of the world and the curiosity or need to reproduce it.
The modelling activity determines aspects of the world to include or exclude from the details that will conform an abstraction of the world that will allow work the answers [22].

Modelling will be our framework for communication between archaeologists' and sociologists' knowledge and their conceptualizations with our formal representations from computer science practices(simulation, algorithms, AI). The reason is that it is a procedure that will help to communicate the **discursive** nature of Social Sciences with the formal structures from Computer Science. The engineering of model development will allow us to reach a connection from experts' knowledge to a model that comprises the set of detail clearing out the ambiguity that language could filtrate. Also, modelling will help set a picture of the system without inconsistencies, with each fact sound, coherent and consistent from the logical point of view with the whole.

2.2 Why model

The modelling process consists in identifying separable entities, processes, relationships and any relevant information related to the question to solve and the domain of study. This abstraction exercise yields an external and explicit representation of part of reality as seen by the people who wish to understand, to change, to manage or to control that part of reality [21].
Indeed just thinking about something implies an unconscious projection of our mental frame hence producing a set of concepts and relationships that give birth to a model. The missed step is that it was not made explicit through some formal representation. A model is a logical and conceptual prototype.

As Epstein [4, p.1] says *"Anyone who ventures a projection, or imagines how a social dynamic, epidemic, war, or migration would unfold is running some model"*.
Modelling is an introspection exercise where you take into account the domain to elaborate a **formal** representation of the conceptualizations you develop around the problem. Mainly, it will have to do with mathematical expressions from calculus or algebra and logics. That is called **conceptual modelling**. This phase comprises the development of a relevant simplification, which must be complete according to the phenomena that inspires the question. Anything left out will change the outcome of the simulation, and non-relevant added items will produce noise that will difficult posterior analysis. All the involved facts must be correctly well grounded taking into account that any unneeded compound in the model will also be added to the scientific and mathematical justification, adding good-for-nothing effort. For instance, lets consider modelling the dynamics of a restaurant to find an optimum allocation of waiters between interior and terrace tables and their serving policy, so it could be minimized the hired waiters while lowering the waiting time of clients. Variables like client arrival rate, kitchen serving time, number of interior tables, number of terrace tables are reasonable parts of the system to add to the simplification. The colour of the curtains, the outfit of the waiters most probably will not account for the stated optimization objective. Someone could argue about the topology of the tables whether it should be added or not to the model. But if the objective is to model the system to analyse the survival rate when there is a fire and people must exit from the building as soon as possible, table and furniture topology is an unquestionable variable.
This kind of criteria should lead to a preference for simpler models. There are many reasons which force to design consciously with this premise. More complex models require harder effort to work their credibility, verification for the correct implementation of the conceptual model, and from the formal point of view, validation of the model and the scientific conclusions. Considering the system conceptualization and formalization, a more complex model is more open to criticism for the objective or subjective choice of features and modelling decisions. Why the present features were chosen and the missing ones were left out? why one expert point of view, and not other one? Also, as Robinson [22] states, simple models have many advantages, such as they are faster, require less data, are more flexible. But the crucial point is if we better understand them we can better interpret their results. Constituents in models interact each other following the relationships established by the modellers. This produces a network or causality chain that is responsible of the state changes of the model along the simulation. These chains must be inspected to find the origin of the phenomena exhibited in the simulation. As we want to understand a system we must reconstruct the processes that lead to the outcome we observe or check against our assumptions or real world events. If we are designing models where constituents are grouped to conform more complicated constituents, if we design the model to exploit emergent phenomena, analysis of the outcome and the causality chain reconstruction will be a very difficult task to disentangle [26, p.31]. Simplicity is a must. Besides, we are not aiming at a very

rich and complex model that matches its outcomes almost perfectly with some referential real data. As it is explained below we have preference for a model that makes easier the task to explore social processes to answer and propose arising questions. We must find the causal connection between experimental parameters and model dynamics, which parameters under which different initial conditions make the system behave differently and say why. Another concern is that detail and granularity choice are attached to the overall direction that takes the construction of the model in terms of structure. At this point it is interesting to mention how does a model can be grown.
According to Oppenheim's and Putnam's[24, p.1-2][25], theories describe their domain as a corpus of interrelated concepts where the pieces of knowledge are connected with mathematical, hierarchical, structural and logical relationships, to enumerate some. The hierarchical relationships induce a multi-layered organization of knowledge, sometimes called ontology[1]. The hierarchy of layers is related to the level of the abstraction of the contained concepts, going from concrete concepts to abstract ones that contain or subsume the former. Also, the composition relationship makes arise a hierarchy of layers. For instance, in social sciences, considering individuals which are part of families, families which are part of social groups, and groups which conform a society.

Usually, modelling methodology takes either the top, bottom or a middle layer to crystallize the model following the hierarchy in a direction towards the higher concepts or the deeper ones. An ascending crystallization from simpler to more complex is called **bottom-up** and the inverse direction is known as **top-down**. Crystallization could begin in a middle layer and stop before arriving a top or bottom bound layer.
For instance, some simulation would consider a necessity to model individual persons with agents taking decision at that scale. Other simulations would model households with the decision process without having to consider individual persons. Once set this issue, the modellization could keep considering more abstract structures like families or tribes as a composition of households and stop here instead of continuing adding villages or countries to the model.
The point is that the range of concepts that you choose to include in the model is related to the level of detail you want apply to describe the constituents of the model. The simplest chosen concepts that merge as constituents of other complex constituents and phenomena mark the granularity of your model.

Although good modelling of the parts could be accomplished accurately discarding the non-relevant entities and phenomena, let's remember the famous quote of George Box, "essentially, all models are wrong, but some are useful" [29, p.2].

By the way, it can also happen that the modellization requires go further the bound. For instance, consider we are modelling a society to see the emergence of some differentiated groups, in our project, hunters, gatherers, agropastoralists. We could model towns, neighbourhoods, go down to families and then arrive to

[1] We are not talking about knowledge representation from A.I.

the person. Maybe we would like to characterize persons with some inner traits related to their personality as anxiety, generosity, aggressiveness, cooperation aptitude. Now we are entering in a layer belonging to psychology sciences, we have surpassed the bottom conceptualization in sociology. If we go further we could arrive to the brain structures entering the field of neurology. We could continue to molecules, biochemistry, and so on and so forth[30, p.56].
Trespassing these borders and needing the help of experts able to manage and modelize the concepts will be the motivation for multidisciplinarity.

Besides having to cope with implicit ambiguity in each discursive knowledge (as said before, Social Sciences usually represent their knowledge in discursive texts using natural language), all these branches must cooperate in a common framework connecting the different used conceptualizations. Some branches can organize their knowledge in concepts of entities, other use processes or actions, for instance. We cannot collapse this frameworks directly in a formal model. The modelling process will elicit this structures and will match them with the mathematical tools offered by the chosen paradigm(lets say Dynamical System Theory, Agent Based Models, Petri Nets). We will translate the conceptualizations to a common language that will connect the formalizations in a whole, the **conceptual framework**.
Modelling will help to find a consensus for expressing the concepts and properties, will help to elicit knowledge, arrange ambiguities, detect common points. It will allow as to embed the needed rigour to work under the same framework to make every part work together. Modelling shall be an exercise of shared development that can approach positions and circle a communication problem to solve the issues that will arise.

2.3 Modelling in social sciences

Social science refers to the academic disciplines concerned with society and human behaviour[30, p.7,chap.3].

Modelling societies implies being aware of constituents identified by the social theories. Interacting entities form a society. Such entities are observed and abstracted from identifiable individuals, people, and activity units, for instance families, neighbourhoods or job partners, composed also of the same individuals. An individual leaves a trace of participations and interactions in the society through social processes. Such activities occur with other individuals, with some activity units or through them. Populations of individuals flow through the social structures, selectively participating and differentially performing [31, p.8]. Ordinary living involves a participation of people in the social activities of family, leisure and holidays, shopping, work and travel. These activities within units are structured by relationships and choices, rules, rituals and randomness. Ordinary living also involves the participation of cultural ideas and artefacts in social activities.

Social sciences seek to understand not only how individuals behave under the

social influence, but also how the interaction of many individuals leads to large-scale outcomes and global phenomena [31, p.9].
Modelization will be the construction of an **analogy** between these constituents and processes identified by social sciences as elements, procedures, terms or expressions in some **representation language** that allows to manipulate and reason about them.
The main observed issues, individuals, units, processes or actions, flow and dynamics within units, are the front-line of the modellization aspects and motivation for the different paradigms appeared or adapted to solve the modelling objectives. New knowledge to infer from this identified phenomena will be the **descriptive** statement of observed behaviour, quantitative empirical **generalizations**, construction or assessment of **theories** and **prediction** models [32, p.9-53].

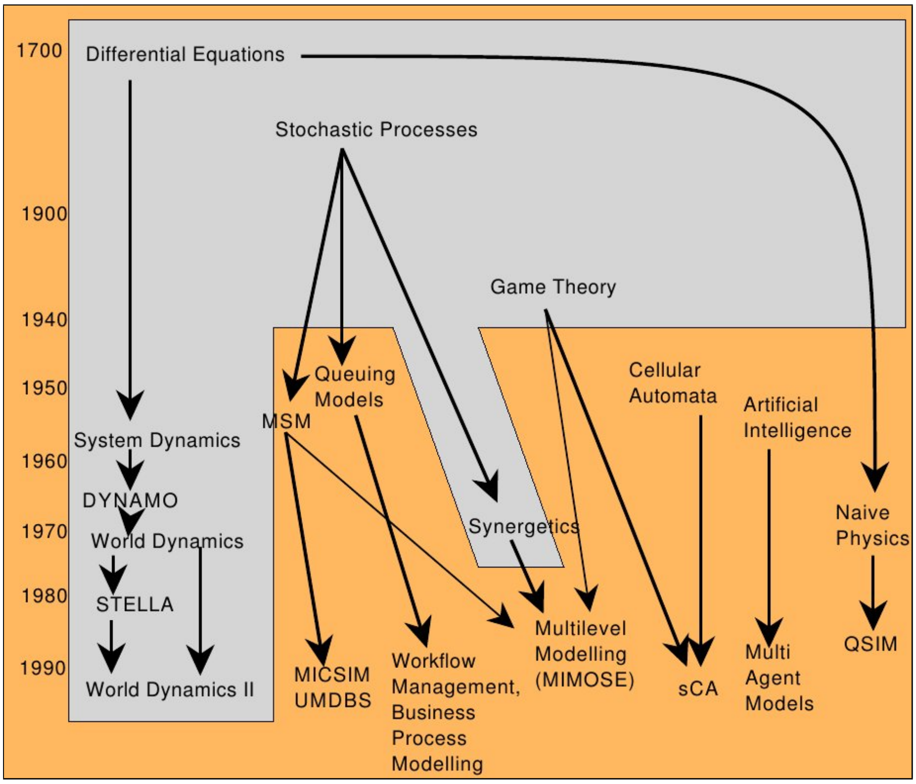

Figure 2.1: The development of the first approaches to simulation in the social sciences (after Troitzsch [12])

The first paradigms to model social processes were borrowed from the fields of physics, operations research, and economics. The first social concepts considered were those related to social units or subgroups and large processes. Also, due to the main use of dynamical systems and differential equations, social phenomena was modelled as a flow between different containers that represent groups or state of individuals.

Figure 2.1: The logic of statistical modelling as a method (after Gilbert 1993)

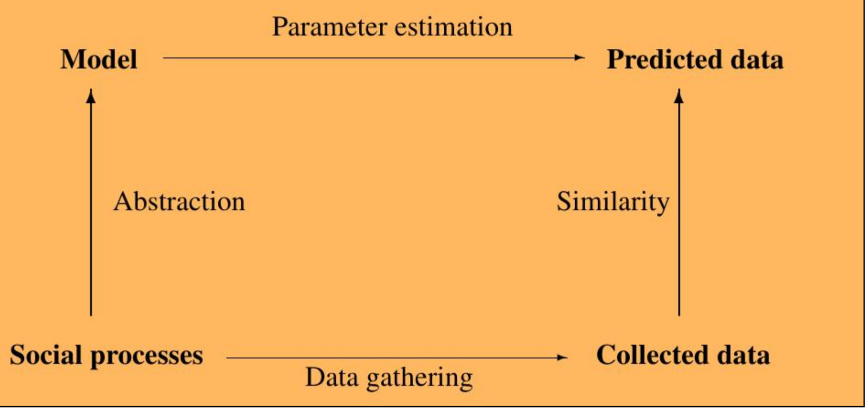

Figure 2.2: The logic of statistical modelling (after Gilbert 1993)[12]

Richer representations to cope with reality led to non-linear specifications and the introduction of heterogeneity present in social systems making them hard to represent or analytically unsolvable, hence, the following years saw the spreading of **simulation techniques**, first AI approximations, cellular automatas and Petri's networks that allowed a finer granularity going from the top abstract groupings inferred in social theories to the individual entities[12, chap.1,p.6-9].

Gradually social modelling began to approach computational sciences keeping its connections to mathematics and statistics. Programming languages are more expressive, less abstract than most mathematical techniques. Programs deal more easily with parallel processes and processes without a well-defined order of actions compared to math equations. There is a quite long experience on studying programs and their properties from Algorithmics, Soft Engineering, and Operating Systems. The engineering of big models benefits from these branches endowing them with the desirable properties of modularity, extendibility, the experience of combining programs to grow huge program systems, error detection and maintenance, to mention some [11].

2.3.1 ABMs in Social Modelling

With its strong component from software engineering, considering the ABM(Agent Based Modelling) methodology to be applied to social modelling there are some striking features that make it stand out from other paradigms or approaches. ABM copes with the simple and bottom constituents of social science implicated in the modellization of a system, hence it can sometimes allow to describe a system naturally. It is not easy to develop an agent, but you are working with a metaphor with a structure and concepts that we have at hand everyday. So it is more manageable

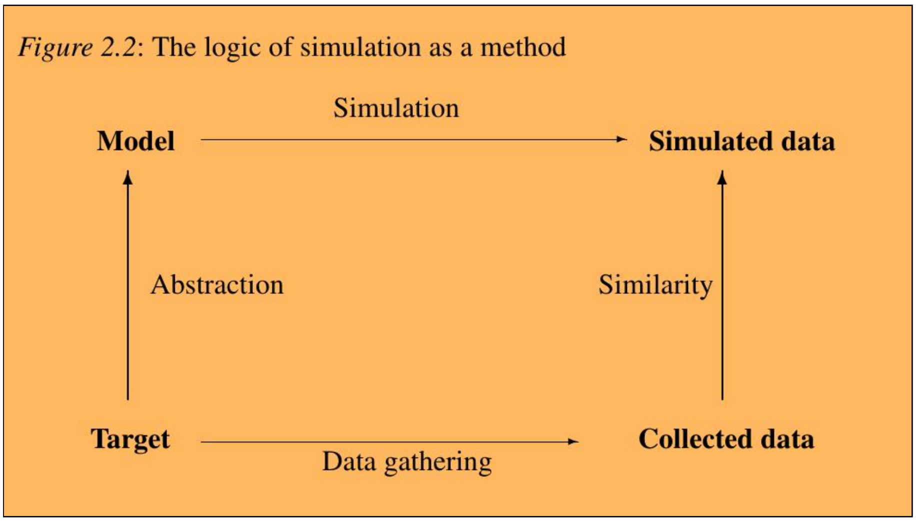

Figure 2.3: The logic of simulation as a method (after Gilbert 1993)[12]

and natural to express things with that "language" giving more flexibility in the modellization.
The use of ABM in social modelling allowed to introduce the bottom layer of the hierarchy of concepts of social science, **people** and all the package of phenomena and issues associated to it. Once you are modelling a person you can work directly with personal or social relationships and their properties, arity, creation and destruction mechanisms, transitivity and interaction rules. This allows to solve models non solved before, like some cooperation, coordination and competition scenarios from game theory where several agents are involved, for instance. Also the direct interaction and feedback effects between the entities and the environment can be represented in the model. Before this step, other paradigms could not cope with the intrinsic phenomena of people interacting in a social scenario. Different issues had to be modelled that were very difficult or impossible to model with raw mathematics. Social entities, let's say people, or social groups, are not like particles which under the same conditions behave the same way. Two distinguishable entities will act differently under the same conditions. People have different perspectives on their social worlds, have a different knowledge corpus or skills [12, p.19]. ABMs allows the introduction of this heterogeneity in the models, and also embedding of the Rational Choice Theory [13], bounded rationality, and complex cognitive processes. We can give each agent an individualized set of traits, features or methods that will model their different performance in the interactions. We can give them or allow them to catch a different picture from the world, the other agents and themselves.
ABM is most indicated for describing and simulating a system composed of "behavioural" entities. Whether one is attempting to describe a traffic jam, the stock market, voters, or how an organization works, ABM makes the model seem closer

to reality. It is more natural to describe how a party of hunters move in a terrain and circle their preys than to come up with the equations that govern the dynamics of the density of hunters. By the way, because the density equations result from the behaviour of hunters, the ABM approach will also enable the user to study aggregate properties[38, p.2]; ABM can manage agents at different levels of aggregation. The decision units can be a person but also social entities like families, couples or tribes with their own rules for interaction and behaviour. You can tune your model easily moving between different layers of abstraction from social sciences. ABM works with models where decision-making and aggregation is clearly separated. The range of complexity of the agent, its behaviour, degree of rationality, ability to learn and evolve, and rules of interactions can be tuned more independently of the range of complexity of the aggregation, individuality, and groups. This allows the modeller to work with different levels of description or complexity in the same model[38, p.2].

Also, modelling with agents from a bottom-up point of view will allow to be near the real causes of macroscopic large scale phenomena non predicted from the microscopic local issues. Lets mention an example from Helbing,[40]; consider a fire escape situation in a confined space: a movie theatre or a concert hall. Let us assume that there is one exit available. How can one increase the outflow of people? Narrowing down the problem, one could ask: what is the effect of putting a column (a pillar) just before exit, slightly asymmetrically (for example, to the left of the exit), about 1 m away from the exit? Intuitively, one might think the column will slow down the outflow of people. However, ABM, backed by real-world experiments, indicates that the column regulates the flow, leading to fewer injured people and a significant increase in the flow, especially if one assumes that injured people cannot move and impede the flow. This result is an example of a **counter-intuitive** consequence of an emergent phenomena: who would think of putting a column in front of an emergency exit? ABM captures that emergent phenomenon in a natural way (see figure 2.1).

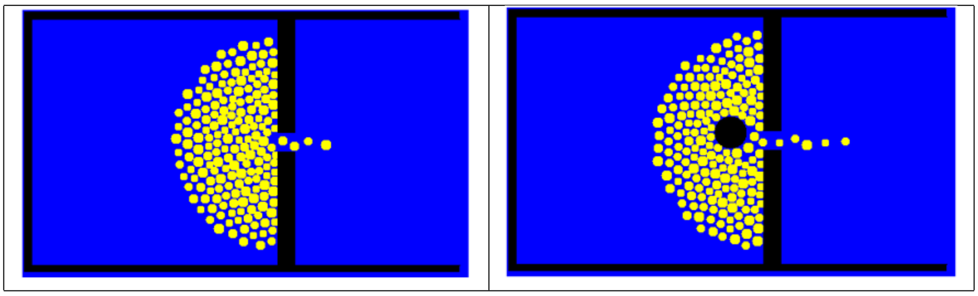

Table 2.1: Unexpected emergence of scape patterns under different situations (from [38]).

Under the next conditions it is advisable, natural and easier to use ABM.

i- The behaviour of individuals cannot be clearly defined through aggregate transition rates.

ii- Individual behaviour is complex. Everything can be done with equations, in principle, but the complexity of differential equations increases exponentially as the complexity of behaviour increases. Describing complex individual behaviour with equations becomes intractable. For instance, the individual copes with hysteresis[2], or there is heterogeneity in the set of behaviours or there are learning procedures and adaptability.

iii- When the interactions between the agents are complex, non-linear, discontinuous, or discrete (for example, when the behaviour of an agent can be altered dramatically, even discontinuously, by other agents).

iv- When space is crucial and the agents' positions are not fixed. Example: fire escape, trade, foraging in a stochastic spatial distribution of resources, traffic.

v- Activities are a more natural way of describing the system than processes.

vi- Validation and calibration of the model through expert judgement is crucial. ABM is often the most appropriate way of describing what is actually happening in the real world, and the experts can easily "connect" to the model and have a feeling of "ownership".

vii- Stochasticity applies to the agents' behaviour. With ABM, sources of randomness are applied to the right places as opposed to a noise term added more or less arbitrarily to an aggregate equation.

A last characteristic feature of agent computing, although not often noted, is that once a model has been created it provides not merely one aspect of the solution — the equilibrium, say, or the stability — but rather entire solution trajectories [39].

Epistemology of ABMs in Social Modelling

Considering specifically archaeology, the introduction of modelling led to the first use of these models as emulation of reality. ABM was used to reproduce the patterns in the material samples found. This led to think ABM as a technique of statistical distribution fitting. ABM models where thought as models that approximated stochastic variables and patterns in the model. This was the scientific use of ABM. With the idea of producing more accurate explanations of the phenomena, ABM models began to be filled with more details. The objective was to approach the

[2] Hysteresis is the dependence of a system not only on its current environment but also on its past environment. This dependence arises because the system can be in more than one internal state. To predict its future development, either its internal state or its history must be known[51, p.571–597].

model to reality to obtain nearer outcomes to the data taken as reference. Those first models were designed to be a mirror of reality, and that is why some experts have been calling them Emulation Models. As it is discussed by Premo [26, p.33] this stalled the usability of models, first due to the difficult of interpreting the causality and processes in a simulation experiment that leads to the arising of phenomena, and second, due to the problem of **equifinality** [26, p.31].

The simulation of a model produces a trace of the states visited by the system. If we execute a stochastic model, the states will be conformed by the exhibited values by the model variables following some distribution. That trace is just only one from all possibles that could appear from the combinations that randomness can produce in the variables. Hence, for a single model and an initial configuration, the stochasticity can lead to many different traces although the system finishes at some attractor state or the same patterns arise. The question is, which trace must be taken into account to explain evidences found in reality? This would be an example of equifinality.

There is also equifinality when given different initial conditions or assumptions on the model the same outcome is reached. Which is the initial scenario that produced the evidences? Due to the long time of the history trace, all initial conditions could have the same probability and data does not make easier the discrimination. Equifinality also appears when the system exhibits sensitivity to initial conditions. Sensitivity is a typical trait of **complex systems** and it is commonly known as the **butterfly effect**. A small perturbation, that is considered **negligible**, in a initial condition can produce a trace with opposite conclusions respect another one without that perturbation. Furthermore, due to sensitivity again, an initial condition in different runs belonging to a same simulation experiment will have usual small perturbations appearing as differences between different traces. Sensitivity to these small inner perturbations can lead to opposite outcomes in the experiment making it difficult to extract and deduce the common patterns or general behaviour we desire to check against data. It will be harder to separate noise from information. The same listed issues hold when instead considering several initial states or assumptions we consider diverse models as candidates for the explanation.

There is another alternative to replicating evidence, another way of exploiting ABMs within another epistemological strategy described by Premo [26, p.33-34] named **Exploratory Models**. Models can be simplified to the bone. This would keep the most essential processes related to the scientific question that motivated the simulation. As it is said above2.2, a simple model allows a clearer insight of what is happening, and of the path of causality between the constituents and the phenomena. Then, taking advantage of equifinality, the use of the model is to explore causality and tendency in the different traces, how happens the connection between assumptions and parameters with phenomena and states. Simplicity in the model allows to design controlled experiments and very explicit and bounded assumptions for hypothesis generation about the model to enhance the knowledge of the processes we model from reality. Then, these experiments will explore the space of parameters under sensitivity to detect ranges, tendencies or configurations

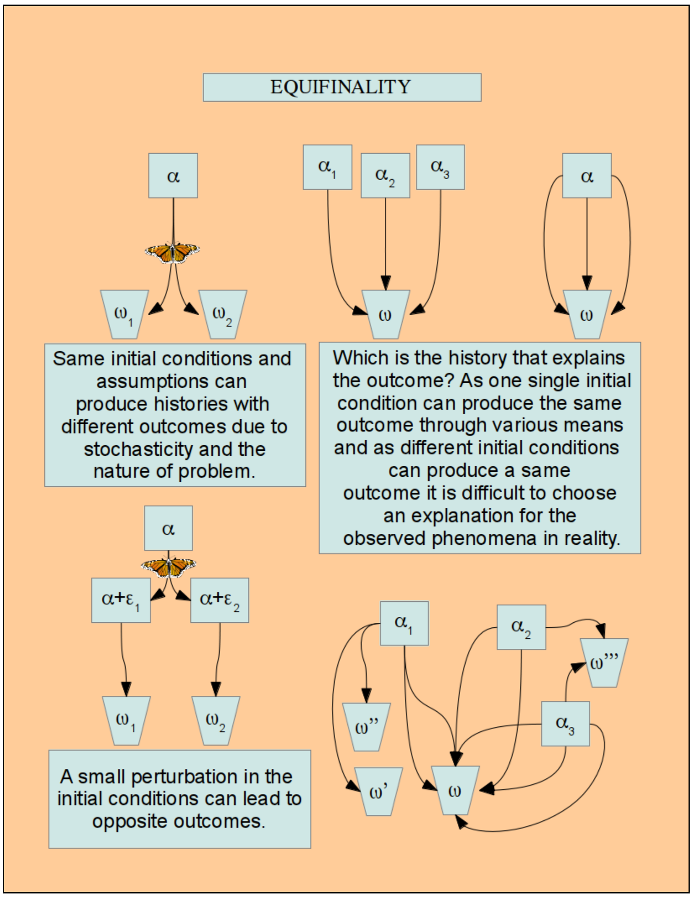

Figure 2.4: Equifinality in stochastic simulation models. For an initial state with its assumptions α a simulation generates a trace of states that lead to a final state or set of patterns and phenomena ω (figure based on [26]).

that produce patterns similar or far from the empirical observations.
The connection between hypothesis, assumptions and outcomes is studied by the modellers applying **retroductive reasoning**[26, p.34]. Retroductive reasoning, also known as **abduction** is an inference frame that connects effects with its causes in a coherent way with an a-priori theory. The best example to catch the concept is to show it compared to other syllogisms.
Deduction makes explicit the facts that some premises entail through some general rules. Given that *"all human are mortal"* and *"Socrates is a human"* I state that *"Socrates is mortal"*. You could see this as a kind of prediction engine. **Induction** produces the general rules, it is an abstraction inference process. For some set of sampled particular cases x_i I observe both properties, *"x_i is mortal"* and *"x_i is a human"*. Then, I could express this correlation inducing the rule *"all mortals are human"* or *"if someone is mortal then it is a human"*, but as we all have seen flies and ants dying, the statement would cover incorrect cases. The final produced gen-

eral rule shall be *"all human are mortal"*. Abduction tries to discover the facts that act as cause for some observed evidences. This inference process produces a causal **explanation**. For instance, I find a dead entity *e*, *"e is mortal"*. I have at hand my a-priori knowledge about the domain, *"all human are mortal"*, and produce my explanation : *"e is mortal"* because *"e is a human"*. That is a plausible explanation, but as shown above, flies and ants are also mortal and it is plausible to say *"e is mortal"* because *"e is an ant"*. Explanation must be refined. Observations and hypothesis trigger new hypothesis to be included in the retroductive inference. New evidences produced by experiments will filter explanations that will allow decide whether *e* is a human or an ant. According to Peirce, retroduction can provide good reasons to pursue a hypothesis but does not, by itself, provide good reasons to believe the hypothesis. In successful applications of retroduction, pursuit leads to the accumulation of evidence that will fix the remaining accepted hypothesis for the explanation [27].
The explanation of the outcome and the evidences from reality arises from the assemble of the knowledge and evidences we gather from experiments. The model becomes part of a pragmatic reductionism frame, what we comprehend from the inner processes is integrated for the comprehension of the whole. This comprehension will induce new questions and hypothesis (or purge them) that will motivate new experiments and modelling tasks producing a loop of model refinement. Generation of these new hypothesis and questions will also be of profit for theory improvement or theory building and self-criticism in a loop of modellization and question-discovering feeding each other.

2.3.2 Summary check-list

The next check-list summarizes the steps visited along the process of development and execution of our model. The ABM methodology fits in the the series of defined stages for simulation based research.

Definition of the target A purpose for the model or a question over the target system is stated. The model will be aimed to prediction or prognosis; diagnosis, to construct an explanation for the dynamics or the state of the system; it also can be aimed to theory validation or theory discovering, or study future possible worlds configurations or attractors in the system.

Observations Data gathering, parameters and initial conditions retrieval is done from the target real system using bibliography, interviews and experts' supervision.

Assumptions Relevant simplifications or principal concepts are considered.

Design model Translation of the experts' conceptualizations to a formal modelling language or structure.

Computer programming Implementation of the model in a computational language. A verification phase will test that the program matches the specifications and features of the formal model.

Run simulation Perform the experiments. The runs will generate the traces that will conform the picture of the exploration space for the stochastic variables in the system and the emergent top level phenomena.

Gather results Extract conclusions from the simulation data, establish correlations and dynamics. This phase will allow to detect evidences for new hypothesis and conclusions or their discarding.

Validation Check that the conclusions are scientifically sound and match the plausible target system behaviour. **Sensitivity analysis** will help detect variables and parameters that produce great oscillations on the simulation results and extract issues for the loop of model, theory and hypothesis refinement.

2.4 Conceptual Framework

This section introduces some definitions and explanations about the main needed concepts to expose the design of the model for the issues and questions stated in chapter one. The text begins explaining what agents consist on. The description of agents from AI covers their architecture and competences, and how different techniques and paradigms endow them with the diverse features and capabilities to solve the problems they are programmed for. An explanation about complex systems follows the section of multi agent systems. It is a necessity due to the nature of the problem and the properties exhibited by models consisting in multiple agents interacting each other. The section links typical complex system phenomena to the arising properties of the models studied for the topics of these master thesis. Emergence is defined and stressed in a dedicated subsection. This part finishes giving some brief ideas about what evolution is. The topic is introduced due to the evolutive component in the model to be run in the simulations. We seek to observe how either of the strategies adapts to the changing environment and how the dynamics favours one instead of the other producing adaptive changes in the agents. As we will study the different social groups in a competitive environment the section will introduce the concept of coevolution, the mutual feedback of evolutive changes between neighbour adaptive groups of entities.After this definitions the next part will retake the topic of agents in the section about Agent Base Models to introduce more ideas and issues from Multi Agent Systems and social simulation.

2.4.1 Multi Agent Systems

Multi Agent System(MAS) is an architecture for software development based on designing a solution for a problem that executes of a set of computational entities called agents that interact themselves in a defined environment[14, chapter.1]. Such entities are active decision making actors in the modelled system. The modelling life-cycle of an MAS will consider a stage where decision making processes must be identified from the system. Usually those decision making actions are carried along by more or less clear individual entities from the system. The modelling process will take the task to set the matching between these entities and the agent that will form the MAS. The concept "agent" condenses a set of features that will specify the modelling metaphor that an agent represents: some enclosed set of mechanisms to be aware of the state of the system, a set of goals to accomplish and the engine to decide from a bounded perception of the world, the actions to apply on it to achieve these goals. Besides the reasoning component of the agents, MASs have a strong component of inter-agent relationship. How one agent interacts with other agents could as important or more as how it reacts to world changes.

Agents

An agent is a computer system that is capable of independent action on behalf of its user or owner, figuring out what needs to be done to satisfy design objectives, rather than constantly being told.[14, ch.1] An agent solves problems applying iteratively the schema of sense, decide and act continuously. Each of the steps can add new targets to fulfil which will keep the agent exploring the environment and interacting with it and other agents to develop its strategies to solve the tasks and achieve goals.

Agents are [15, p.115-152]:

i- clearly identifiable problem solving entities with well-defined boundaries and interfaces;

ii- situated (embedded) in a particular environment—they receive inputs related to the state of their environment through sensors and they act on the environment through effectors;

iii- designed to fulfil a specific purpose—they have particular objectives (goals) to achieve;

iv- autonomous—they have control both over their internal state and over their own behaviour;

v- capable of exhibiting flexible problem solving behaviour in pursuit of their design objectives—they need to be both reactive (able to respond in a timely fashion to changes that occur in their environment) and reactive (able to act in anticipation of future goals).

An example of a very simple agent would be a thermostat. It samples the environment with a probe/sensor, checks if the temperature corresponds to the desired

one. According to the measure it launches one of the two actions, to set the heater to ON or to OFF.
But we could have something as complex as an agent representing a car in a traffic simulation while drives from one point to another in the city. The agent would have accessible a knowledge base of traffic rules that constrain its actions. The agent at each time step of the simulation would control the swarm of other vehicles to avoid collisions and would follow paths and perform actions coherent to the rules of traffic. The traffic rules would act as middle interface consensus of fair driving that coordinates all the cars and allows some prediction of the adjacent cars. The agent can be designed a step further. When congestion increases to a predefined threshold, which could be tuned through automatic learning, the agent recalculates the route to adapt and avoid the traffic jam.
Intelligent agent design considers the setting of Perceptions, Actions, Goals and Environment. These aspects form a general structure where an agent design can grow. Mentioning an example considering a system that simulates hunting practices from ancient cultures will show how features from the agent are attached to these modules in the structure(table 2.2).

Perceptions	Hunger, TerrainSlope, ReadTracks, LookForPrey.
Actions	Eat, LightFire, Cook, ThrowArrow, Walk, Run, Stalk, Hide.
Goals	Survive, AvoidHarm.
Environment	Weather, Plants, Mountain, Valley, Caves, Deers, Rabbits.

Table 2.2: Setting table for a hunter agent.

Agent brief formal description Let E be an environment with a finite set of states

$$\Omega = \{\omega_0, \omega_1, \dots\} \tag{2.1}$$

Each agent has available a set of actions

$$A = \{\alpha_0, \alpha_1, \dots\} \tag{2.2}$$

An action is a function that changes the state of the environment

$$\alpha_i : \Omega \longrightarrow \Omega \tag{2.3}$$

An agent's run is a sequence of states of the environment where the transitions were triggered by actions launched by the agent. Let R be the set of all possible runs over Ω and A, and r^u one sequence of length u.

$$r^u = (\omega_0, \alpha_0, \omega_1, \alpha_1, \dots, \omega_{u-1}, \alpha_{u-1}, \omega_u) \tag{2.4}$$

Let R^A the sequences that end with an action

$$r_A^{u-1} = (\omega_0, \alpha_0, \omega_1, \alpha_1, \dots, \omega_{u-1}, \alpha_{u-1}) \tag{2.5}$$

Let R^{Ω} the sequences that end with an environment state

$$r_{\Omega}^{u} = (\omega_0, \alpha_0, \omega_1, \alpha_1, \ldots, \omega_{u-1}, \alpha_{u-1}, \omega_u) \tag{2.6}$$

Actions in the run are the response of an agent X_i to the states of the environment

$$\forall j \in [0..u-1] : X_i(r_{\Omega}^{j}) = \alpha_j \tag{2.7}$$

An stochastic and historic dependent environment behaviour, τ, can be described as

$$\tau : R^{A} \longrightarrow \wp(\Omega) \tag{2.8}$$

A Markovian environment behaviour would be defined as

$$\tau : \Omega, A \longrightarrow \wp(\Omega) \tag{2.9}$$

and the next would hold for the runs

$$\forall i \in [0..u-1] : \tau(\omega_i, \alpha_i) = \omega_{i+1} \tag{2.10}$$

Let's say that no end condition is described for the runs here because although it could happen $\tau(\omega_i, \alpha_i) = \varnothing$, many other conditions could mark the end of a run. For instance, a run can finish because the population of agents reaches 0 due to some dying process. A run can also stop after a finite number of transitions, or some predefined event appears.

An agent X_i retrieves information from the history of environment states to choose an action to launch

$$X_i : R^{\Omega} \longrightarrow A \tag{2.11}$$

Agent Main Behaviours There is a set of key points that an agent should accomplish to go beyond a simple AI application, a problem solver, or just a system process. AI has compiled some asked mandatory behaviours listed below to a software to be considered an agent [14, ch.2][8, ch.1, ch.2].

Autonomy Other entities do not set the agent objectives nor decisions. With agents, we give a high-level description of the delegated goal, and let the control mechanism figure out what to do, knowing that it will act in accordance with some built-in theory of rational agency to satisfy it.

Reactivity Response to environment stimulus or changes. A reactive agent is one that maintains an ongoing interaction with its environment, and responds to changes that occur in it (in time for the response to be useful). A pure reactive agent, the thermostat can be formally described as

$$X_i : \Omega \longrightarrow A \tag{2.12}$$

Proactivity It means anticipation, taking initiative, detect opportunities. This could materialize in prediction of a future event an realize a set of a priori actions before it occurs. For instance, if you are modelling a population of farmers and agent A1 sees the state of low resources of its neighbour agent A2 at the end of the season. As a social action, A1 makes a present of food to A2 before it begins to starve or asks for help. We also know that this kind of action will produce stronger bonds and could have their pay-off in the future. But the most extended example is when you enter in a bookshop, as you delay a bit in your search, a salesman appears offering its help before you ask for it. It could mean that the agent produces a set of actions that trigger an environment event that will allow the execution of an action the approaches the agent to its goals.

Social capabilities Cooperation, coordination, negotiation, competition, and mind models. Some objectives are not achievable by the only means of the agent. The agent must interact with other entities that can produce the chain of actions to produce the changes in the environment needed. Agents inhabit and interact with the environment applying actions and producing effects in a same medium. Goals, effects and dynamics can clash. Is it possible the appearance of conflict. Goals and planned trend can be contradictory for more than one agent. According to the model of negotiation of the agent it can try to change its plan trying to not interfere with the other agents, or produce a deliberate interference or ignore it and stick to its objectives. Social capabilities will cover the spectrum of communication but also integrate the other agents behaviour. Some sophisticated agents include mind models to add, predict the other agents' behaviour in its knowledge representation engines.

Cooperation Cooperation is working together as a team to achieve a shared goal. Often prompted either by the fact that no one agent can achieve the goal alone, or that cooperation will obtain a better result (e.g., get result faster). That is very easy exemplified with hunt parties or some families of farmers that while one member takes care of the plot the other takes the cattle for grazing.

Coordination Coordination is managing the interdependencies between activities. For example, if there is a non-shareable resource that you want to use and I want to use, then we need to coordinate.

Negotiation Negotiation is the ability to reach agreements on matters of common interest. At the appearance of conflict a solution that benefits the parts is searched. For example: Two farmers arrive at a piece of land good for crop growing. A possible deal: split the land in two.Another solution : both work on the land, but each has assigned different tasks. By the end of the year they divide the harvest. Typically involves offer and counter-offer, with compromises made by participants.

Learning Prediction for future situations,reuse solutions,avoid past errors. The agents stores patterns from the history of environment changes, or other agents' actions. The patterns induce a model of the world or of the task to perform used by the agent for future actions. As a detail, although the learning could be produced in first stages of the run to produce benefits along the life of the agent, it is desirable that the learning should occur along all the run to produce a real adaptation of the agent. The keyword is **incremental learning**.

Intelligent Agents Architecture This paragraph will show different decompositions of how an agent is structured and provide an answer to the question of how the sensor data and the current internal state of the agent determine the actions and future internal state of the agent. The thermostat example contains an environment with a bounded and tractable number of states. Such situations can be solved with a direct implementation setting the bijective function state - action with a table or with a limited number of rules. As complexity of modelled systems grows, the number of states and possibilities become intractable. There is no time nor space to specify each correspondence. The agents apply different techniques for retrieving features and structure from the environment to proceed with the decision process from an abstraction to the action to perform [8, ch.2].

Reactive Architectures

The decision process in Reactive Architectures selects actions only based on the last perception retrieved from the environment. It does not consider any subset of past perceptions. Reactive Architectures encompass the Simple Reflex Agents.

A change in the environment provokes a response from the agent. When changes and deliberations in agent are only motivated by an event in the environment, the agent is pure reactive.

while *true* **do**
  $\sigma \leftarrow$ getNextPercept();
  $rule \leftarrow$ ruleMatch($rules$, σ);
  $\alpha \leftarrow$ ruleAction($rule$);
  execute(α);
end

Algorithm 1: Simple Reflex Agent main loop

- Cognitive Maps
- State Transition Machines
- if-then rules

Deliberative Architectures

A deliberative agent uses symbolic reasoning to deduce the action to launch. The deliberative agent contains explicitly the goal that steers its behaviour. Deliberation is done through an internal formal representation of the state of the world, the state of the agent and other information retrieved by it. A logic engine will produce deductions from the facts stored in the agent memory, the knowledge base. The perceptions from the environment become facts to add to the internal representation of the environment. From this internal world model the agent can deduce trends for prediction besides the next action to launch. A state of the environment or of the agent is specified. The agent must find the way to fulfil this within his reach of perceptions and actions. Desirable situations are sought, called goals, that is environment states or agent states.

Given some systems or situations, goals are non achievable from a single action execution. The agent must apply search and planning techniques to conform a plan that will satisfy, after a limited number of steps, the desired goal.

Goals could be a design feature of the agent, for instance **survive** & **reproduce**, or could be set dynamically in runtime by conditions in the environment or through user commandment.

Logic Deliberative agents will use assert clauses and structures to represent facts in a logic of some level, CP0, CP1. Each new change in the knowledge base will allow to deduce new facts from the status of the system to check against the goals and other issues the agent is considering to produce an action coherent with the planning and the mechanics of the environment to fulfil a new step to get nearer to the goals. The agent will use its knowledge base as a theory of the world and things plus the dynamic facts that represent the volatile states.

Let ρ be a theory of the world. Depending on the architecture this can be a set of rules, or a learned structure along the run.

If Γ is a description for the current state of the world. A the set of possible actions $\{\alpha_1, \alpha_2, \alpha_3, \dots\}$. And $\Gamma \vdash_\rho \Phi$ stands for a successful prove that Φ is deduced from the knowledge base Γ using theory ρ. The Deliberative agent will choose actions according to a schema like that:

forall the $\alpha \in A$ **do**
  if $\Gamma \vdash_\rho doAction(\alpha)$ **then**
    return α;
  end
end
forall the $\alpha \in A$ **do**
  if $\Gamma \nvdash_\rho \neg doAction(\alpha)$ **then**
    return α;
  end
end
return null;

Asserts from Logic will be used to state the facts that are true from environment and the other agents.

BDI BDI stands for Belief-Desire-Intention. Yoav Shoham introduced "agent-oriented programming" in 1990 [17]:"new programming paradigm, based on a societal view of computation". The key idea is about directly programming agents in terms of **intentional** notions like belief, commitment, and intention. Beliefs are used to model the state of the world. Desire allows the selection of possible states of the world and preferences. Intentions are compromises to achieve a given state, they are the commitment of the agent.

BDI belongs to a overloaded kind of logics called modal logics which add meta language operators to the facts to alter with some nuance the meaning or the semantic interpretation of the fact. For instance, let ϕ be a fact that could represent "deer is in the wood". Indeed it is true for the example. Our agent has not seen it and only has some clue that the deer is in the wood; so the agent states $\Box_B\phi$, with an operator $\Box_B$ to indicate "I believe the deer is in the wood". The memory of the agent will contain also the facts below

θ = "I am hungry".

η = "I go hunting".

$\Box_D\eta$ = "I have Desire for going hunting".

$\Box_I\eta$ = "I have Intention for going hunting".

Some of these entail from other. They are governed by the next rules from the knowledge base

- myState(HUNGRY) $\rightarrow \Box_D$ doAction(HUNT)
- $\Box_D$ doAction(HUNT) $\wedge \Box_B$ entityAtPlace(x,y) $\wedge$ edible(x) $\rightarrow \Box_I$ doAction(HUNT)
- $\Box_I$ doAction(HUNT) $\wedge \Box_B$ entityAtPlace(x,y) $\wedge$ edible(x) $\rightarrow$ doAction(go, y)

- entityAtPlace(MYSELF,p) $\wedge \Box_I$ doAction(HUNT) $\wedge$ entityAtPlace(x,y) $\wedge$ edible(x) $\wedge$ distance(p,y) $\leq$ HUNTDISTANCE $\rightarrow$ doAction(HUNT,x)
- doAction(a,x) $\rightarrow$ launchAction(a(x))

This rules would carry the agent to satisfy the goal of feeding going through the goal of going where the prey is, and see that it is indeed there where the agent believed.

Programming agents this way allows a deeper description of the dynamics around the goals an agent self-imposes and also an easier way for the agent to reason about the other agents. This last point is crucial for a better integration of the social dynamics in the planning of goal achieving. The agent can construct through facts of Believing, Desire and Intention a mind model of the other models taking into account their BDI intentions and adapting to them to cooperate or compete.

Continuing the example, the agent would be immersed in social deliberations after retrieving from the world facts like $\Box_I^i \eta$ that stand for "Agent i has the Intention of hunting". The interaction of all these facts, entityAtPlace(MYSELF,P), entityAtPlace(i, P) will make deduce to the MYSELF agent some kind of communication protocol with agent i to solve the conflict. Because they could end to hunting the same deer, before they would launch their HUNT action, they must reach a consensus about what to do each other.

Schema for BDI

> $B \leftarrow B_0$;
> $I \leftarrow I_0$;
> **while** *true* **do**
> $\delta \leftarrow$ getNextPercept();
> $B \leftarrow$ brf(B, δ);
> $D \leftarrow$ options(B, I);
> $I \leftarrow$ filter(B, D, I);
> $\pi \leftarrow$ plan(B, I);
>   execute(π);
> **end**

Algorithm 2: BDI main loop

Utility based Goal-based models are a dichotomic approach that rely on bivalued logics. A goal is considered achievable or not achievable, an agent is committed to a goal or is not committed to that goal. But many fields have shown things usually work in a fuzzy manner or probabilistically or with multivalued assignations. Mentioning some example, robot soccer. In a match, eventually, you could pass the ball to a partner or try to score. If you launch the action "pass" things will happen different from launching action "score" but a priori you cannot predict which will be of more help. And if you stop to think about it the opponent will steal the ball from you. Another example, a fire simulation. There are many escape routes; when a route is saturated due to congestion, people have a variable of internal time-out wait time to leave the route and try another one. There is also fuzzy phenomena in auctions; auction agents have to decide whether raise the "bet" or leave.

An agent under these circumstances will find a set of available goals where each one produce a different benefit for the agent itself. The goals, the states of agent and environment must be weighted in function of probability of success, or benefit for the agent. Also, there is some related optimization trend or related likelihood of achieving positive results 2.10.

A pure logical agent can find a cul-de-sac on its deductions if the possible contingencies overcome the reasoning to deduce the actions to fulfil the goal. These issues are what is called **uncertainty**. Uncertainty is the term used when there are many solutions that lead to the goal or if the goal is something not sure to be achieved and the agent faces a choice with no success ensured.

For instance, considering a hunter agent in a social simulation of ancient societies, a hunt action can be affected by many issues. First you must make a guess about where the preys are. Once you begin the hunting session many things could happen: a weather incident that changes prey grace behaviour, other hunters appear to hunt the preys, or dangerous roaming predators. The pure bi-valued logic agent cannot predict the setbacks so cannot deduce the success of the action, and the hunt action will not be launched leading to starvation. If all the logical paths towards the achievement of the goal are troubled by contingencies, the agent could fall in a loop of inactivity.

When perfect solution is perhaps non achievable, the agent should be allowed to satisfy the goals through a non perfect solution. The decision engine should be changed to accept fallible solutions for its goal seeking. Another hunt area should be eligible in the decision process. But maybe, although the other area is free of contingencies, it cannot offer a good outcome because is almost desert. Some way the agent should express a **preference** and deduce the action planning according to it. Preference over the outcomes of and action in the goal seeking activities of the agent encompass the modelling of success probability and benefit quantification. A particular outcome should be the fact that the hunter returned home without getting injured and a certain amount of Kg. of meat.

Utility theory is used to represent and reason with preferences. Utility theory says that every state has a degree of usefulness, or utility, to an agent and that the agent will prefer states with higher utility [8].

Preference and similar issues are modelled through **utility functions**.

$$u : E \to \mathbb{R} \tag{2.13}$$

Utility functions map desired states to a quantitative dimension, a scoring. Utility weighting allows to break ties in sub-goal selection. Utility functions will give extra heuristic information about the likelihood achieving a goal. If the hunter agent has access to two areas of hunting, *valley* and *wood* and is more likely to find dangerous predators in the *wood* although the same preys can be found on both areas, the utility function would assign greater utility to the *valley*. Utility provides a way in which the likelihood of success can be weighted against the importance of the goals.
Preferences, as expressed with utility functions, are combined with probabilities in the general theory of rational decisions called **decision theory**: [8]

$$Decision\ theory = probability\ theory\ +\ utility\ theory. \tag{2.14}$$

Utility-based agents are rational and follow the principle of **Maximum Expected Utility (MEU)**: *An agent is **rational** if and only if it chooses the action that yields the highest expected utility, averaged over all the possible outcomes of the action.*[8]. This feature is interesting due to its fit in the Rational Choice Theory [**?**] which tries to give a frame for the explanation of behaviours and decisions in an economical domain. It is one of the justifications of using planning strategies in the agents we have developed.

Environment Features The design of the agent is conditioned by what is considered "the environment" in the real system to model. It will facilitate or constraint the features and decisions in the development of the agent. Next, some common features that will guide in the characterization of the environment are enumerated [8, section 2.3].

Accessible vs Inaccessible An accessible environment is one in which the agent can obtain complete, accurate, up-to-date information about the environment's state. Most moderately complex environments (including, for example, the everyday physical world) are inaccessible.
The more accessible an environment is, the simpler it is to build agents to operate in it. Accuracy and completeness about information retrieved is key to decision making. All the missing stretch must be supplied with uncertainty managing techniques.

Deterministic vs non-Deterministic As we have already mentioned, a deterministic environment is one in which any action has a single guaranteed effect

— there is no uncertainty about the state that will result from performing an action. The physical world can to all intents and purposes be regarded as non-deterministic. Non-deterministic environments present greater problems for the agent designer.

Episodic vs non-Episodic In an episodic environment, the performance of an agent is dependent on a number of discrete episodes, with no link between the performance of an agent in different scenarios. Episodic environments are simpler from the agent developer's perspective because the agent can decide what action to perform based only on the current episode — it need not reason about the interactions between this and future episodes.

Static vs Dynamic A static environment is one that can be assumed to remain unchanged except by the performance of actions by the agent. A dynamic environment is one that has other processes operating on it, and which hence changes in ways beyond the agent's control. The physical world is a highly dynamic environment.

Discrete vs Continuous An environment is discrete if there are a fixed, finite number of actions and percepts in it. Russell and Norvig give a chess game as an example of a discrete environment, and taxi driving as an example of a continuous one.

Task Environment	**Observable**	**Deterministic**	**Episodic**	**Static**	**Discrete**	**Agents**
Crossword puzzle	Fully	Deterministic	Sequential	Static	Discrete	Single
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Poker	Partially	Stochastic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partially	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partially	Stochastic	Sequential	Dynamic	Continuous	Single
Image-analysis	Fully	Deterministic	Episodic	Semi	Continuous	Single
Part-picking robot	Partially	Stochastic	Episodic	Dynamic	Continuous	Single
Refinery controller	Partially	Stochastic	Sequential	Dynamic	Continuous	Single
Interactive English tutor	Partially	Stochastic	Sequential	Dynamic	Discrete	Multi

Table 2.3: Examples of task environments and their characteristics.

2.4.2 Complex Systems

Complexity is a discipline that studies systems that cannot be studied through the analysis of its parts and the simple composition of those analysis. Usually, **reductionism** has coped with complicated systems where the suppression of constituents does not change the main trend of its behaviour, and it is used to work with simplified questions. Let's think for instance in a car which more or less keeps running although you remove its seats, the doors, and the lights, to mention some. That is what could be called a complicated system [43]. There are systems where the interrelationship is so rich that suppression of one part dismembers all the coherence of the system. Taking again the car example, if we think about the engine, the suppression of a most tiny gear will change the things, everything will stop. Historically, those systems have been an issue that was of concern to many disciplines, Economics, Physics and Biological sciences, that gave arise through an interdisciplinary effort to complexity science [3] [41]. Complex science tries to find out how the interactions of a set of constituents build a system that exhibits adaptive traits, decentralized organization, and macro-scale behaviour patterns.

One of the most known systems studied by its complexity are ant colonies. The colony is formed by a limited range of roles, a queen, soldiers and workers/explorers. Each role exhibits a limited spectrum of behaviours activated by pheromones or stimulus from the environment. Soldiers attack anything that moves non impregnated by the odour of the nest. Unloaded workers follow explorer's pheromones. When food is found a worker returns home. Dead bodies or garbage found are put near other garbage, generating dump patterns. More or less this is the program of ants. And this executed by a huge amount of ants produces an ant colony that manages food resources, reproduction and care, defence and many other adaptations to the wild life in the forest. It can loose part of its population, can balance workload, a same specific individual group of ants is not needed for an specific task. The colony is ductile, malleable. The ant colony acts has a whole with very rich global behaviours. That is a complex system.

Researchers observed that woods under a fire disappear at different rates not completely dependent to the intensity of the focus, nor the climatic conditions. It was observed that under some tree topologies and certain wood density a small change in the number of trees would mean sometimes the complete combustion of all the wood, and other times the wood would only burn in a small controlled area. This topology dependent condition is called **percolation**. It arises from the spatial relationship between the trees. Considering an ideal situation where topology protects the wood from percolation appearance, slight changes in some trees distribution will make them sensible to fire, but the rest of the wood will not burn. These conditions are an example of robustness in complex systems. But enough changes will collapse it.
The strong relationships that give arise to the complex phenomena act as the glue that holds the system, it gives reason to the system existence.
Percolation was not discovered studying one tree. It was detected taking into account the whole of the wood and the feedback versus the individual trees.

Indeed we observe that there is a system due to the presence of the macro

[3]here we are not talking about the computational concept of algorithm complexity.

phenomena. It is the evidence that something is happening there that relates the constituents.

Just to give a compact definition from Melanie Mitchel [41], “a complex system is a system in which large networks of components with no central control and simple rules of operation give rise to complex collective behaviour, sophisticated information processing, and adaptation via learning or evolution. Summarizing, a complex system exhibits non-trivial emergent and self-organizing behaviours.

The laws that describe its behaviour are qualitatively different from those that govern its individual units. The advances of computational techniques for scientific discovering is changing the way we confront the models and the manipulation of these issues. Computers have allowed new ways of learning about them. Computational techniques allow the modelling of many constituents and its relationships, how they assemble to a whole system. It allows us to understand and play with the experimental manipulation of the behaviour in a easier way. Computers not only help us in studying the surface of the behavioural phenomena but also the internals of complex systems giving insight to our explanations and deductions. Besides, computational experimentation has the advantage of controlling the trace and generating rich databases of events for posterior analyses [44].

Complex behaviour is tied to a non central global decision process. But, on the other hand, the emerged pattern is something that does not belong to a identifiable constituent. It is build from the decentralized dynamic of all the constituents. That feature is called **emergence**.

2.4.3 Emergence

Emergence is a phenomenon where aggregation of individual and local behaviours are the direct cause of a higher level pattern or global behaviour. Emergence is one of the key concepts from Complexity that is studied in Social Simulation. Indeed, emergence is one of the main features exhibited by complex systems.[43]

Although we can give features and descriptions, emergence cannot be fully specified due to the vague concepts of “surprising” and non deductibility [42] . But some signals can be enumerated to circle the concept of emergence.

A phenomenon will be considered emergent when it is repeatable and surprising, non deducible from lower level rules and the relationships of basic constituents of the system. Emergence cannot be predicted, given our current means, from only the constituents and their addition, following the reverse way of reductionism. Reductionism is one of paradigms of science. A system is observed to detect differentiated parts. Then each part is analysed to extract its behaviour. Reductionism states that the behaviour of the system can be understood from the constituents specifications plus a simple or direct aggregation step. This idea can be applied recursively on the same constituents till some atomic element considered the bottom of the process. The thing is that due to non-linear relationships, following the inverse path to reconstruct a sense for the whole system is practically impossible.

Even a small and simple set of rules can make a dynamic system generate emergent phenomena. If these rules induce non linear relationships the proportion of a perturbation in the system will not be paired with the response of it. To use an uninventive example, non-linear emergence occurs when someone **calmly** says “fire” in a crowded room and produces **explosive** panic. Unpredictable results arise from constituents interactions. It is here where emergence can appear. When a pattern is recognizable and repeats usually along experiments, is plausible to be considered emergent [42].

There is potential for emergent phenomena, i.e., when:

- Agent behaviour is non linear (weight sum of variables), or is expressed with discontinuities, if-then rules in a categorical framework or in discrete non continuous manner.
- Under memory phenomena, path-dependence, and hysteresis, non-markovian behaviour, or temporal correlations, including learning and adaptation.
- Heterogeneous interactions between agents.
- When there is instability to perturbations although the system could be defined linearly.

2.4.4 Evolution

Evolution theory is the result of observations along several years made by Charles Darwin in his trips to Galapagos Islands and the study of native species of finche birds.

Darwin proposed that the varieties and specializations of observed species was imposed by the topology of the island and the local conditions of each one. The theory arose from the following points in conjunction of the observations. He based his deductions on the next hypothesis:

- Gradual change over long periods can produce very large effects.
- Population growth combined with limited resources creates a struggle for existence.
- Collections of individuals acting in self-interested ways produce global benefit.
- Life seems to allow almost infinite variation, and a species’ particular traits seem designed for the very environment in which the species lives.
- Species branch out from common ancestors.

Darwin called **Evolution by Natural Selection** to the improvement by mutation and competition process where individual beings produce offspring at a rate greater of the survival rate (otherwise they would extinct). The offspring is almost equal to the parents except from slight variations. At some point the population will saturate the niche and they will compete for the resources. The more adapted individuals are considered those who will satisfy their resource needs and have higher reproduction success. This will imply that a great number of offspring that inherited features from their successful parents will populate the environment. These traits will persist through the time from generation to generation. This would explain why individuals are as they are and not other way. Adaptation comes from the small perturbations or changes between parents and offspring. This is an open door to the appearance of an improved trait that would increase the adaptation of the new generation. Hence, change after change the Evolution sculpts step by step the organic beings making them more adapted to the environment. Traits that does not allow to survive nor reproduce will not appear in the next generation, except due to some rare mutation but that will be purged again in the competition game against the environment.

To summarize the major ideas of Darwin's theory:

- Evolution has occurred; that is, all species descend from a common ancestor. The history of life is a branching tree of species.
- Natural selection occurs when the number of births is greater than existing resources can support so that individuals undergo competition for resources.
- Traits of organisms are inherited with variation. The variation is in some sense random—that is, there is no force or bias leading to variations that increase fitness. Variations that turn out to be adaptive in the current environment are likely to be selected, meaning that organisms with those variations are more likely to survive and thus pass on the new traits to their offspring, causing the number of organisms with those traits to increase over subsequent generations.
- Evolutionary change is constant and gradual via the accumulation of small, favourable variations.

Darwin theory had some points to be clarified. How did parents pass their traits to the offspring? The process of refining these theories comprised the addition of Mendel's theories lasting till the past century where everything was unified with the discovering of DNA and genomics. DNA is the way in which are encoded the traits that the organic being will probably exhibit in his life, and also DNA is the transmission medium of the traits in the reproduction process. We will take **Richard Dawkins** position as our final work metaphor for evolution and natural selection[49].
As it is mentioned, part of the working framework of our simulation is based on the

ideas of evolution and "natural selection". We understand "natural selection" as a process that "rewards" adaptive solutions and penalizes those less adapted. Darwinian Evolution is the result of the continued application of this screening and the persistence of adaptive patterns and what comes off of them. Persistence is produced through the reproductive process of agents. Regularly an agent generates a copy of itself with some disturbance in their features and takes charge of it for a period of time. The extraction capacity of system resources, adaptability, marks their survival and that of their offspring. If the offspring survives, the configuration is maintained over time and gets another chance to be perpetuated when this new generation begets his sons / daughters. We use evolution as a tool for selection of configurations to respond more adaptively than others, working with the hypothesis that selected ones would correspond to reality. We simulate systems plus the evolutionary process. Our utility functions, birth & mortality are filters to keep or remove agents of the system depending on its performance against its life-cycle.
We are applying a parallelism between DNA information and configuration or features in the agent. The features establish the policy of the agent. For instance, the Gujarat project would differentiate between Hunter-gatherer way of living from Agro-pastoralist way of living through this features.

Coevolution *When two or more species form an interdependent ecosystem the evolutionary progress of part of the ecosystem will generally induce co-evolutionary changes also in the other species* [47, p.154].

Long ago, bacteria and plankton dominated the seas. There was only microscopic life. Then appeared photosynthesis based microbes. With Sun light and CO_2 chemistry they spread at a greater rate than their neighbours. Photosynthesis microbes filling the seas plus millennia produced the oxygen filled atmosphere of the planet. Changes in the atmosphere changed weather and many other chemical properties in the planet. The atmosphere contained ozone which barred the UV rays of the Sun which forbid life in the surface. Then some individuals appeared with capabilities to get out of the water and breath the air of the surface. It is an example of mutual feedback between the environment and the evolving life beings. But this feedback can also happen between these beings, and then it is called coevolution[49]. When the adaptive process of an species produces a change that motivates selection of new adapting traits in other species which in turn affects the same way the first species we have coevolution. For instance, the relationship between a prey and a predator will induce coevolution. The hunting activities produce death in the prey species, hence motivating the selection of traits that give more probabilities in front of a predator attack. Predators feast on preys with feeble traits. Old feeble traits disappear. Then predators must hunt the individuals without the feeble traits. Incompetent predators will starve and a new elite will appear. A mutual selection pressure is established between both species feeding changes of new "attacks" and "counter-attacks and "counter-attacks to the "counter-attacks.

Although we will not wire explicitly coevolution in the two studied social models we expect to see this phenomena. In Gujarat project we expect to observe coevolution between Hunter-gatherers and Agro-pastoralist as they compete for resources and land. The cultural traits are considered an analogy of adaptive strategies.

Cultural Evolution Behavioural Sciences trying to explain the cultural idiosyncrasy of human beings have traced a parallelism between its dynamics and Darwinian Evolution theories. Cultural Evolution [49, ch11.][50](socio-cultural evolution, CE) states an analogy with the selection, adaptation and innovation processes described in the theories of darwinian evolution. Culture endows humans with a mental frame of assumptions and behaviours that affect their response to the environment. This has an impact in human performance to the treats of the environment and hence adaptation to it. Culture is a trait that like the physical features that conforms an individual will be crucial to its resilience.
Another important dimension of culture is transmission. Culture like genes is something parents can transfer to their offspring. Also, besides receiving a vertical endowment from parents, culture can also be acquired from other peers in a oblique way, maybe from brothers and sisters, or grandparents. This process is called cultural transmission. Just happens like in replication of parental genes, there exists the possibility of errors in the copy process. Either the communicator of cultural traits or the receiver can alter the information leading to errors or improvements. There could exist also improvement in knowledge due to revision and update of the results of the behaviours through experience. Next a table is presented to show the parallelisms between evolution theory and cultural evolution theory.

CE theory assumes a morphism with contemporary evolutionary theory; CE theory states that the current state of culture for some social group is a result of natural selection upon the individuals exhibiting some cultural traits that changes its resilience in the environment. Environment pressure will be a selection and scanning mechanism of cultural traits.

Applying these ideas, concerning the Gujarat project, we will try to explain the change from one adaptation strategy of hunter and gathering to an agropastoralist one based on the assumptions of selection and resilience applied to both strategies.

We study the skills and strategies for resource exploitation. The successful strategy, hunting and gathering versus agropastoralist activities or sophisticated AI versus classical agent, will yield a greater probability of success to the agent. The evolutive-coevolutive framework will trigger a dynamic of selection and gradual change. In the long term we will observe the equilibrium properties reached by coevolution.

In our study of transition from HG to AP, both groups will be modelled as two different typologies of agents. Each typology will account as a differentiated group where members will belong to it along all their life. When an agent emancipates from its parents takes the decision of being HG or AP according to an evaluation

Modern Evolution Theory (NeoDarwinism)	**Cultural Evolution**
Information is encoded in DNA strands, chains of genes, the genome. It will be expressed in the physiological traits, structure and metabolism, the phenotype.	The cultural corpus, containing ideas, mental frames, patterns and know-how, is added to the behaviour of the individual.
The physiological traits achieve some level of fitness to the environment setting the adaptability and resilience of the individual.	Behaviour and know-how conditions the response and adaptability to environment(physical world and other entities) challenges. How well the individual responses defines its fitness and resilience.
Higher resilience gives higher opportunities for offspring generation. The fittest individual retrieves more resources and avoids best the treats. The fittest reproduces more.	The individual endowed with a culture that enhances its resilience survives to transfer it to its offspring / partners / peers. Higher resilience gives higher opportunities for offspring generation. The fittest individual retrieves more resources and avoids best the treats. The fittest reproduces more.
Parents transfer their gene information to offspring : vertical transmission.	Cultural transference can happen through multiple channels, parents, peers, authorities : oblique transmission.
Copying dna although it is a process with correction mechanism has a ratio of error that introduces mutations in the copy and hence new information that entails new traits, which can report benefit or lead the individual to a lesser fitness than its parents.	Learning is a process where the absorbed corpus can be altered by the communicator, the receiver, or due to added new knowledge. The error in transference or execution of learned skills and knowledge can lead to innovation.
Offspring inherits traits. There is a correlation of parents' fitness respect offspring's fitness. Usually , strong parents produce strong offspring which will find easier to develop in the environment and hence reproduce, repeating the process from generation to generation.	The individual endowed with a culture that enhances its resilience survives to transfer it to its offspring, partners or peers. The individual acts as the medium through which culture is transferred.
Fit gene combinations persist in time displacing less adapted ones.	Fit culture or knowledge persists in time displacing less adapted ones.

Table 2.4: Summary of parallelisms between contemporary evolution theory and cultural evolution theory.

of which of both groups seems more successful. When an agent takes the same role of the parents it can be seen as vertical transmission of cultural traits. When an agent takes a role different from the parents we can assume that it learned these skills from neighbours in environment. It must be stressed that we pay attention to transition to agriculture, not to emergence of agriculture in a only HG environment. Somehow agropastoralist yet exists in the environment. We only want to know how different environmental conditions favour one or another strategy and why HG kept being a plausible strategy in Gujarat when it had disappeared in other places. Each group has particular actions representing their way of making their living. HG execute Forage actions and AP execute Crop actions (with subtasks of preparing, cropping and harvesting fields). But there societies where both strategies coexists inside a group. Individuals show competence and skills to survive applying a mix of HG and AP actions. So the final aim is to remove groups. If we understand strategies as knowledge, knowledge will be contained in the agents in a fuzzy and dynamic way instead of wired eligible actions. There will be mixed agents that will execute HG actions and AP actions with different proficiency on them. We will call *skills* to this construct. And these skills represent the "genetic" traits that we want observe evolving from this framework of CE. Each agent will have proficiency in the four skills from the merge of HG and AP groups : hunter, gather, crop, shepherd. A typical Hunter Gatherer considered in the scope of this master-thesis work would be an individual with a near 100% proficiency in hunting and gathering while having 0% on the other actions. And a pure AP individual would be endowed with the complementary assignation, 0% for hunting and gathering with almost 100% for actions Crop and Shepherd2.13. In the long term we want to observe the gradual transformation from one strategy to another through the change of profiles of proficiency. As knowledge is something where skills are not mutually exclusive, we consider reasonable that agents can execute the same set of actions. The dynamics of selection and environment pressure will benefit some configurations from others. This can lead to profile attractors or cyclic emergence of profile patterns according to the periodicity of climatic fluctuations.

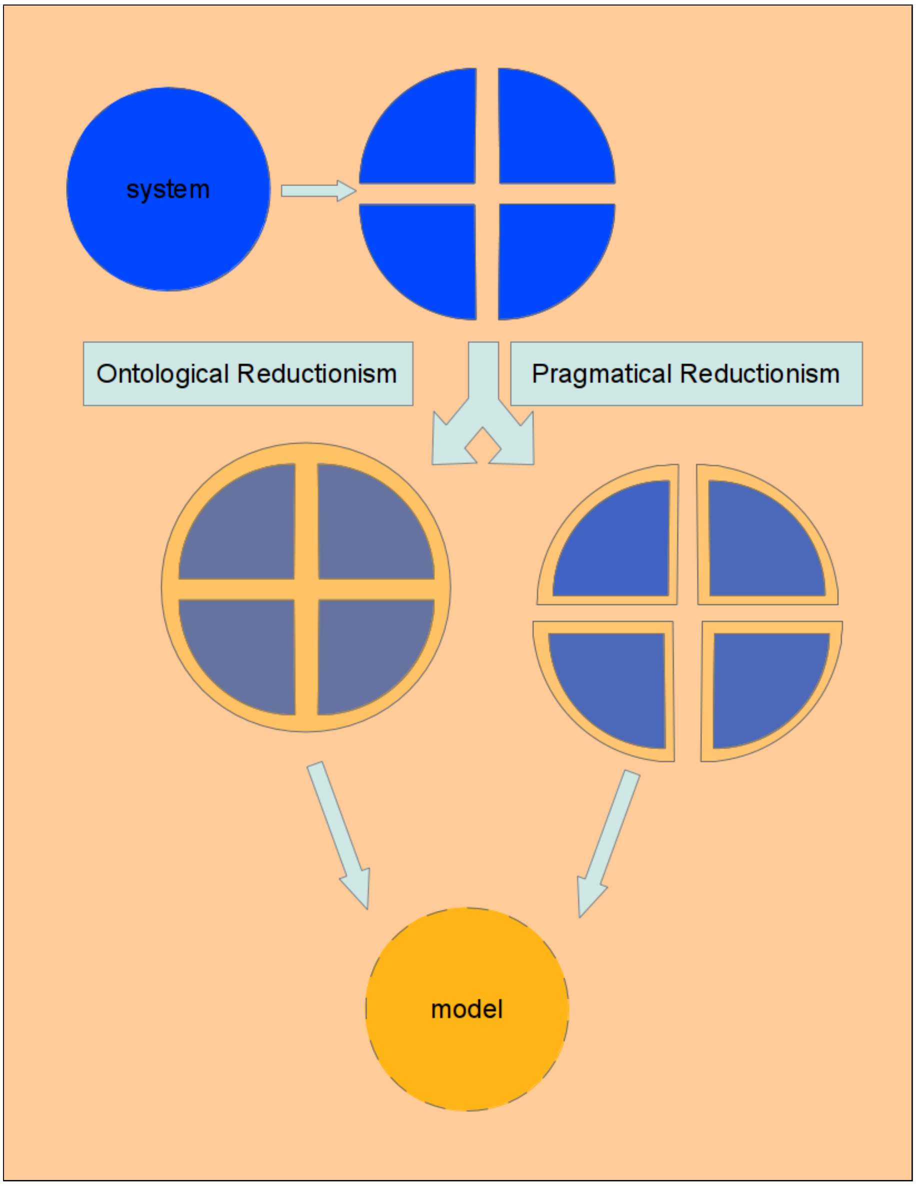

Figure 2.5: Ontological Reductionism vs Pragmatical Reductionism.

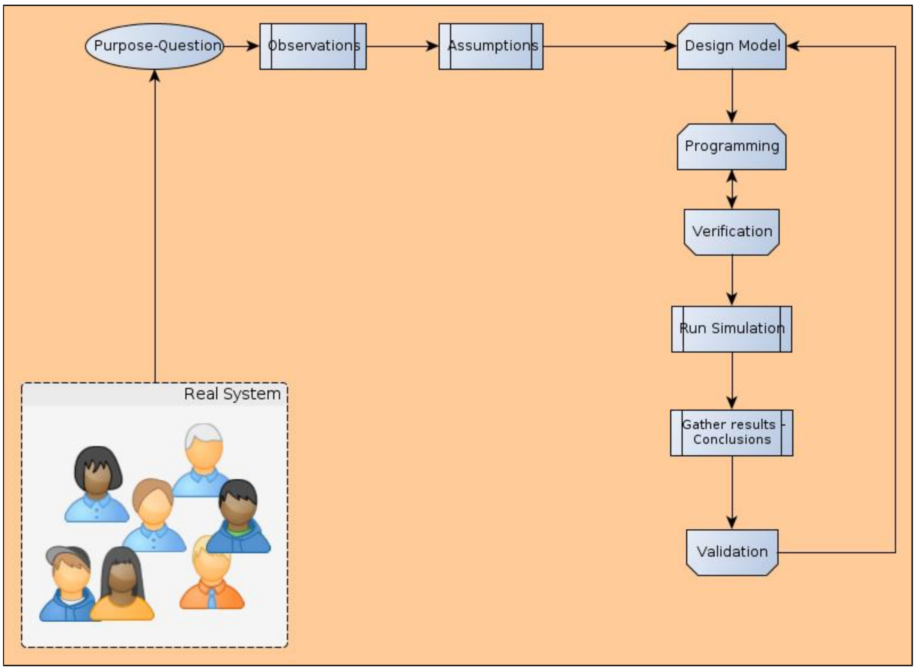

Figure 2.6: Stages defined in simulation-based research [8]

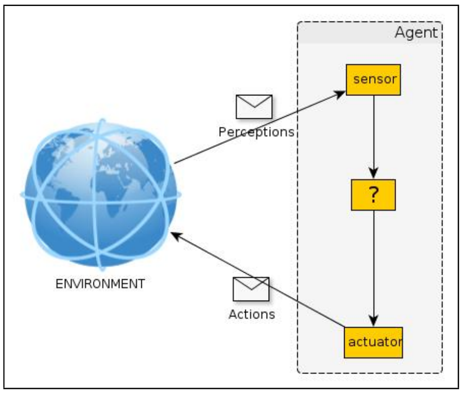

Figure 2.7: Agents interact with environment through sensors and actuators.

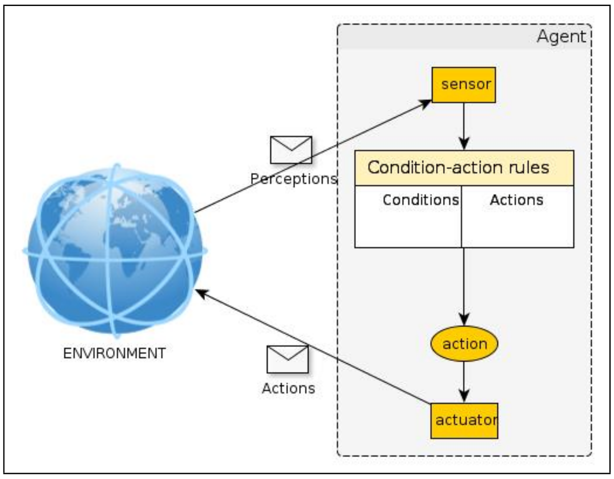

Figure 2.8: Simple Reflex Agent Architecture

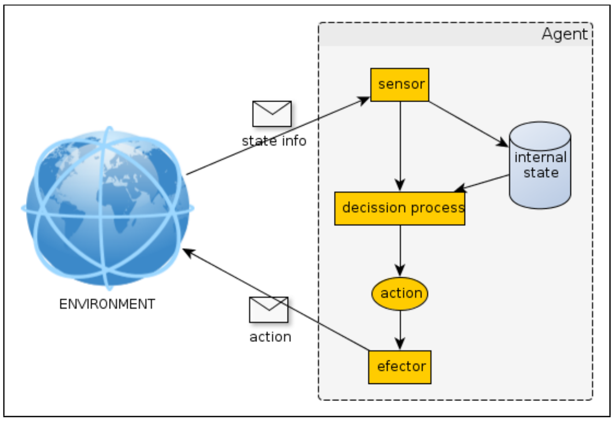

Figure 2.9: State Machine Reactive Agent Architecture

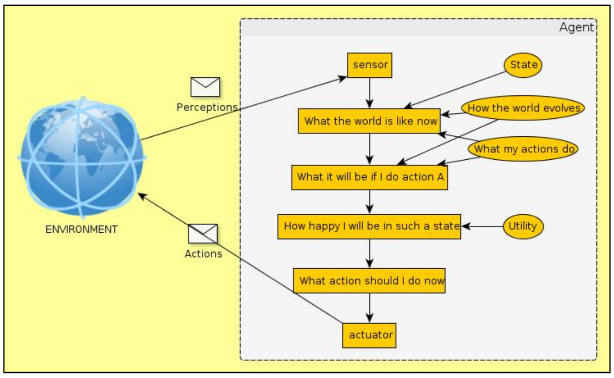

Figure 2.10: Utility Based Agent Architecture

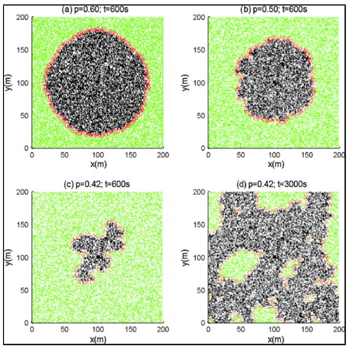

Figure 2.11: Percolation phenomena in fire spreading.

Figure 2.12: Development of spiral waves after hydrodynamic breaking of a concentric wave (Zhabotinsky and Zaikin, 1971).

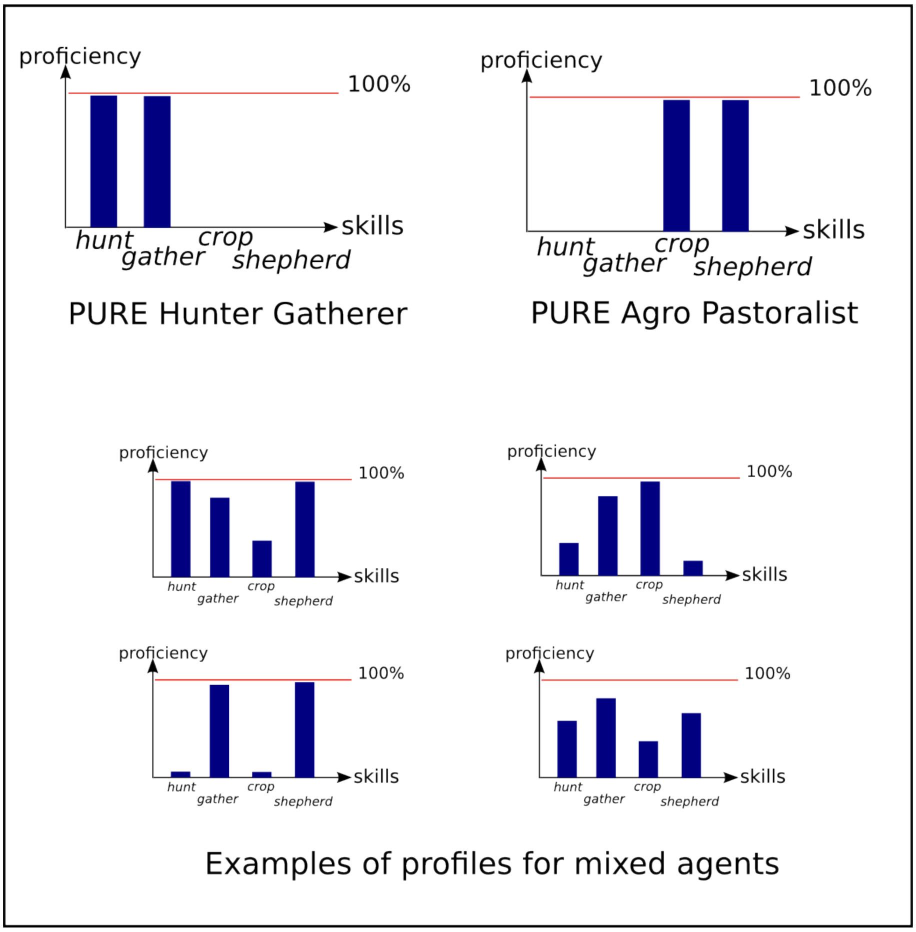

Figure 2.13: Skill Profiles.

Chapter 3

Gujarat

3.1 Introduction

This chapter exposes the work in case study one in the Simulpast project. The text follows the structure of ODD documents we have been using to register our decisions and models and as a communication report in the group. ODD is a document structure with an overall look of high-level specification but open to detailed explanations [23]. ODD documents have been thought to be able to yield enough details so another researcher can replicate the system and produce the same results and conclusions. We think this feature is a good point for expert-modeller communication. It helps to keep an account of the details and the refinement of the discursive part of knowledge from social science experts into formal components of the model. We know ODD is not perfect. It does not follow Object Oriented methodology, structure, or entities and its relationships, but we wanted to ensure the communication part for elicitation applying ODD philosophy following a growing procedure of a seed system based in Sugarscape [3].

The first section of the chapter introduces the observations and hypothesis of archaeologists that have been studying the scenario of Gujarat during the transition and settlement of agriculture. This part states the motivation that leads to the purpose of the model. The following sections get deep in the details of the system, environmental features, resources, weather conditions and dynamics, the agents and the features for matching the relevant system dynamics to solve the question. After the modelling details the next section exposes the solution for solving the decision making of the agents. A planning strategy is needed to achieve maximum adaptability of the agents under the different weather conditions scenarios for stress experiments. Markov Decision Processes and UCT algorithm will be introduced. Once the decision making is explained, different solutions are presented to enhance efficiency and throughput of steps per second in the simulation. The last sections cover the experimental part of the project. It will be centred about the resilience of the current social group modelled, hunter gatherers.

3.2 Target Description

Northern Gujarat is a marginal environment between the Thar Desert and the more fertile area of Saurashtra. This region is an ecotone, characterized by the seasonal influence of the monsoon where contrasting ecological niches are in tension and small climatic shifts can generate significant environmental changes, eventually affecting resource availability. Archaeological evidence points to the presence and possible coexistence in the area of groups of people with different resource management strategies and mobility behaviours: hunter-gatherers (HG); agropastoralists (AP). The aim of this study is to model resource management and decision making among hunter-gatherer groups in this region to explore adaptive trajectories and performance in relation to a) environmental variability and b) the appearance of other specialized groups. What factors play a role in HG persistence or disappearance in arid margins? Is the advent of agro- pastoral behaviour a big enough change to explain the disappearance of HG behaviour? Does climate variability affect HG behaviour? The agent-based simulation here proposed explores the potential for the persistence of hunter-gatherer (HG) communities relative to climate-driven environmental change during the Holocene in N Gujarat, a semi-arid region in NW India.

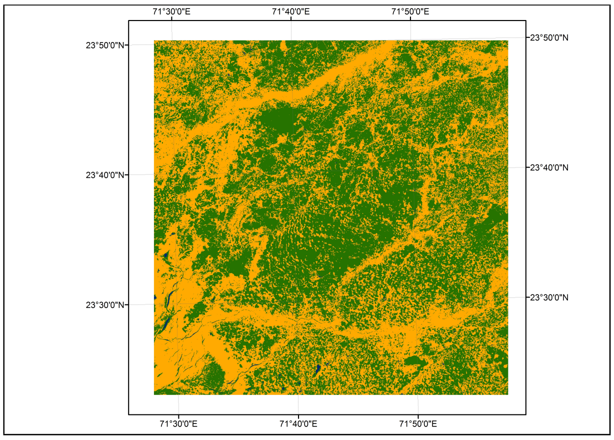

Figure 3.1: Gujarat area for HG and AP resilience simulation.

Note: The modelling work-flow is following these steps: 1. Environmental settings and climate engine 2. Resources and energy flow 3. Socio-ecological behaviour of HG 4. Socio-ecological behaviour of AP 5. Cultural transmission

This scaling approach includes three main theoretical and methodological research aspects: 1. Human behavioural ecological approach and socio-ecological co-evolutionary framework 2. Model based vs rule based action planner of the agents Up to the date, steps 1, 2 and 3 have been developed . Work regarding AP models and cultural transmission is in progress.

3.3 Purpose

In our starting hypothesis HG groups are adapted to marked seasonality (represented by the monsoon) in the arid margins of northern Gujarat. We intend to explore HG resilience (Holling 1973, Carpenter et al. 2001) considering climate variability.

Future : We intend to explore HG resilience considering the appearance of AP

3.4 Entities, state variables, scales

The model is designed to explore separately socio-ecological behaviours of two isolated populations: 1) Hunter-Gatherers (HG) agents and 2) Agro-Pastoralist (AP) agents. This separate simulation of the two groups is needed to obtain independent, coherent and consistent models of HG and AP decision making. Once the dynamics of the modelled systems are understood for each population, agents from the two populations will be combined in a single simulation execution. These will be considered as independent groups, and interaction between agents will be limited to other agents belonging to the same population. But the scope of this work is focused on exploring the socio-ecological behaviour of HG populations.

Agents from one population will interact within a given territory. It is characterized in terms of: a) geographical information derived (height, slope), and b) landscape (soil types, resources). This territory and its characteristics will generate an environment that will allow to portray differences in strategies regarding settlement, mobility and resource use.

3.4.1 Scales

Agent Scale The basic agent is defined as a couple (one woman and one man). This is considered to be the entity engaging in all decision-making processes and actions modelled in the simulation.

Time Scale Time Scale for the simulation is one day. This time step is coherent with the granularity of agents' planning. The decision making of HG happens at the level of their daily action.

Space Scale The spatial resolution of the proposed simulation model is constrained by the resolution of available relevant geographic data and the nature of the agent mobility and resource gathering activities being modelled. Hence, it was decided to use 31.5m x 31.5m cells, corresponding to ca. 1000 square meters. This is the level of resolution of the most detailed geographical information available for the area. This surface fits the type of settlements recognized from archaeological surveys.

3.4.2 Environment

The simulation environment is large enough to develop all potential processes defined by the model according to the experts' advice. It extends over an area of 50 Km x 50 Km (2500 Km2). Space is represented as a regularly spaced grid of cells (a raster map). Each cell is a square of 31.5 m per side, and the total size of this environment translates into a space of 1600 x 1600 cells (50,400 m x 50,400 m). The ground model includes elevation and land features. Elevation is determined by a Digital Elevation Model (DEM), a raster map containing the elevation value for each cell calculated from contemporary satellite imagery. Land characteristics are reduced to three elemental categories:

1. Water: represents rivers and lakes.
2. Dune: represents the top area of the dune, which can be settled. Home location of the agent will always be in a dune cell.
3. Interdune: represents the interdune(between water and dune) area where resources grow. The different land features do not seasonally change in extension but their productivity (in terms of moist content and therefore resources supported) does.

The cornerstone of our environmental modelling is the climatic 'engine'. The climate module determines the quantity of rain that precipitates evenly on the landscape on every time step. Precipitation is used in conjunction with the terrain model to calculate the amount of biomass for each cell and season. The climate model is based on historical data, as well as Holocene monsoon models.

Climate

The focus on resource utilization strategies within a particular environment requires to make explicit the potential variations in the landscape. In particular for our case study, the presence of the monsoon generates a strong seasonality (asymmetrical precipitation patterns). Monsoon seasonality determines the presence of three critical "moments" in simulation time, each spanning 4 months. Therefore, the seasonal subdivision in three periods will be repeated in a cyclical way as follows below. Identification of months from western calendar covered by each of the seasons is done taking the first letter of the month name and using the concatenation to label the season. So "JJAS" stands for the season that covers June, July, August and September, for instance. "ONDJ" is the label for the season covering October, November, December, January. "FMAM" is for February March April May.

1. JJAS (rain season: high precipitation, high temperature, low evapotranspiration)

2. ONDJ (post-Monsoon: low precipitation, cool temperature, medium evapotranspiration)

3. FMAM (dry season: low precipitation, high temperature, high evapotranspiration)

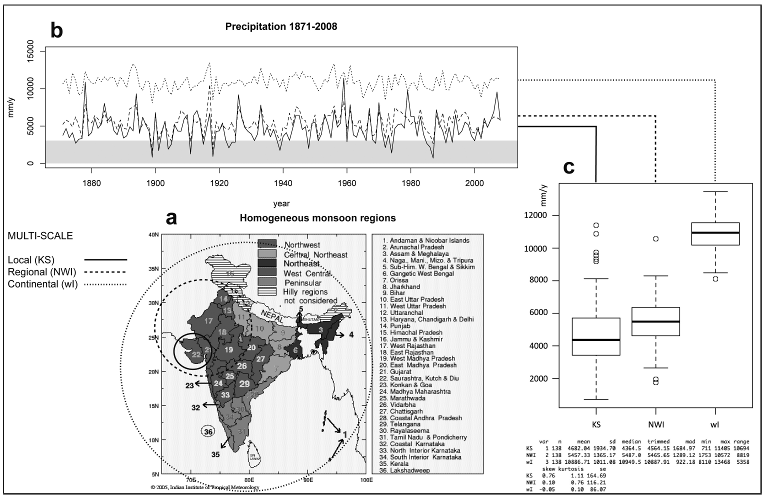

Figure 3.2: .

It is important to note that any given "year" in the model starts with the beginning of the rain season (June). In fact, virtually all rain in the region is carried by the monsoon that falls between June and September (JJAS). Therefore, it is during the JJAS season that the totality of the generated yearly precipitation value is calculated (following a Gamma distribution). No additional precipitation is considered for the remaining eight months of the year (ONDJ and FMAM).

Resources

Each cell has a finite number of resources. Resource availability for each cell is calculated from the following variables:

1. Yearly precipitation (rainfall, using a Gamma distribution)

2. Type of cell (Water Body, Interdune, Dune)

3. Mean yearly Biomass per cell and type.

4. Cell history (e.g. whether the cell was used for crop in a previous step or part of the resources were consumed before).

Resources include the total biomass that can be found in a cell (fauna and flora). They are exploited by HG agents engaging into foraging activities. Foraging includes activities such as hunting animals and gathering plants, fruits, seeds, etc. Indeed, from a literature review it is clear that secondary biomass production (animals) is directly related to primary biomass quantity. Moreover, as there is no interest in our simulation to explore gender-based labour division we decided to consider hunting and gathering as a single activity (foraging) without distinguishing between plant or animal utilization. In the light of this, it was decided to consider a value for cell (dune vs interdune) based on published information of primary biomass production in desert (dune) and savannah (interdune) biomes (after Kelly 1983 – Table 3).

Cell Type	Yearly Primary Biomass Production	Efficiency	Energy	KCal
Dune (desert)	$700g/m^2$	13%	$1820KJ/m^2$	$435KCal$
Interdune (savannah)	$4000g/m^2$	23%	$18400KJ/m^2$	$4395KCal$

Table 3.1: Parameters for resource parametrisation according to Kelly (1983; Table 3, p. 284).

1. Cell area = 1000m2
2. 1 g Primary Biomass = 20KJ
3. 1 kcal = 4.184 KJ

Distance to water This average value of resources is modified based on the distance of a cell from the closest water body. This weight decreases linearly to the distance, and models the heterogeneity of biomass generated by a higher presence of flora and fauna near the zones where needed water can be collected. The update due to resource growing will depend on the matrix of distances to the nearest water body representing this information.

Efficiency The total primary biomass value does not constitute the entire primary biomass available for consumption to animals and humans. This value represents the entire biomass production including both edible (fruits, tubers or some roots) and non-edible (wood, stems, branches for instance) parts of the plants. The ratio of profitable biomass versus whole biomass will be the efficiency value specified in the above table that allow to calculate the energy effectively available for humans.

Domesticated cell (work in progress) The model is prepared to tag cells whether they are domesticated or wild when we will introduce AP agents in the runs Domesticated plant resources define the density of domesticated plants found inside the cell. It is important to note that wild and domesticated types of resources are mutually exclusive: i.e. wild resources are not found on cultivated plots.

The basic idea for a domestication cell is that of small scale shifting cultivation, in which a plot is cultivated following a cycle that includes abandonment so to allow soil properties to recover. A domesticated cell can be planted for a maximum of three consecutive years. It then needs to be abandoned for a minimum of three years (during which it will be considered as Fallow). After three years of abandonment the cell becomes wild and can be used again for agriculture. If a plot is abandoned before being cultivated for three consecutive years it will have to remain abandoned for the same amount of years it was worked. During these years it will remain fallow, after that it will become wild again.

Crop cells change temporary their state to Fallow during the FMAM season, where no agricultural uses can be executed. It turns to Crop again during JJAS season, and will be ready to sow at the end of it.

3.4.3 Agents

The following attributes have been chosen to account in our model.

Age - A numeric variable that keeps track of how many time steps a given agent has been active in the simulation.

Children - A collection of human entities. The number of children is bound by resource availability. Birth and mortality rates depend on resource availability and hence regulate the number of children.

Home location - the cell where the agent resides and the spatial centre of the activities it carries out. Any number of agents can share a given cell.

Home range - Maximum distance an agent may travel in one day. This attribute restricts foraging and social activities from taking place in cells too far from the agent Home location. The area enclosed in the circle of radius "Home range" is divided in six equal sectors. Such division models the idea of direction of exploration for the agent actions, while simplifying the decision-making process (an agent will choose to forage in one of the six sectors, independently of single cells).

Food needs - Value that sets the minimum calories a given individual needs in each time step in order to survive. The total amount of food needs for an agent is computed as the sum of the food needs of the individuals that form a given agent. The probability of death increases for all the individuals that form a given agent when this basic quantity of resources is not foraged within a day (due to starvation). Needs are defined by the following table:

Available forage - Daily time that an agent can spend on foraging. The total amount of foraging time for any given agent is computed as the sum of the foraging time of the individuals that compose it. Foraging time increases from infancy to adult life, modelling learning processes as defined in the table3.4.

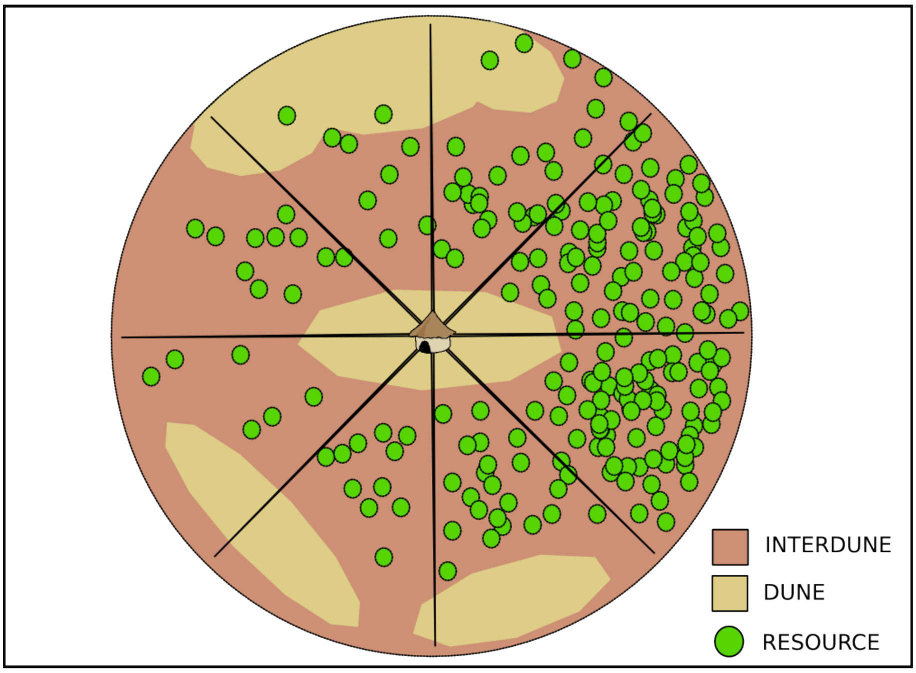

Figure 3.3: Figure. Home range division for foraging and moving home actions

3.5 Process overview and scheduling

Execution follows two time-scales. On the one hand, three processes ('yearly precipitation', 'biomass yearly production' and 'population size adjustment') are executed once every year. On the other hand, agents decision-making processes are updated on a daily base. The simulation follows this schedule, beginning the first day of the JJAS season: For each year:

1. 1. Precipitation calculation
2. 2. Biomass yearly production
3. 3. For each day of the year:
   (a) a. Daily biomass availability
   (b) b. Agent planning:
      i. i. Knowledge update
      ii. ii. Choice of actions
   (c) c. Execution of agents actions
4. 4. Population size adjustment

Details for each simulation phase are given hereafter.

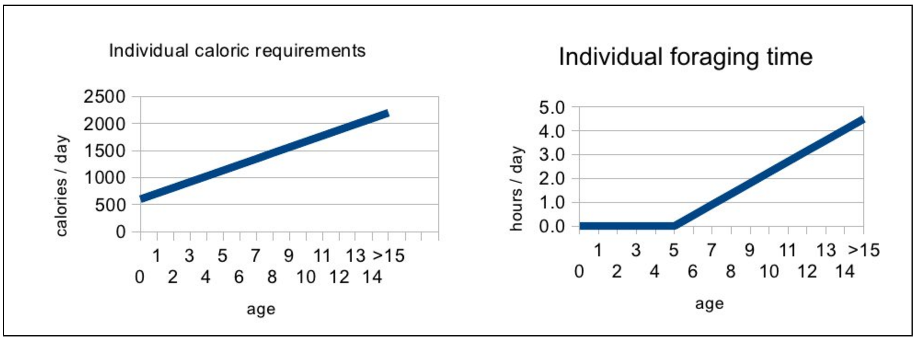

Figure 3.4: Tables. Individual caloric requirements (left), Individual foraging time (right)

3.5.1 Precipitation calculation

The total amount of rain is calculated as a random number following the Gamma distribution defined in section 7 (Input Data)

3.5.2 Biomass Yearly Production

The biomass that a cell will produce in an entire year is calculated from rainfall and mean year production for its particular type, provided by historical records. We consider a linear relation between rain and biomass production. The deviation of rain in a given year from the period mean allows interpolating the amount of biomass deviation from the yearly mean biomass. That is, if the mean of rain is 100 litres and the climate model produces 80 litres the deviation to apply is 20%, and for that year the biomass will be a 80% of the mean yearly production for the period.

3.5.3 Daily processes

Biomass availability

Yearly biomass production does not appear immediately in the cell in the first day of JJAS season. Resources increase gradually, following a cumulative pattern that accounts for the progressive accumulation of water through JJAS, until the beginning of ONDJ. From then on, resources decrease linearly to the end of the year, then, they reach a percentage of the highest peak defined by the ‘end- of-year minimum residual resources’ parameter (EMR). Variations in EMR does not affect overall yearly biomass production so that the higher the EMR, the lower the maximum peak of resources at the JJAS-ONDJ boundary.

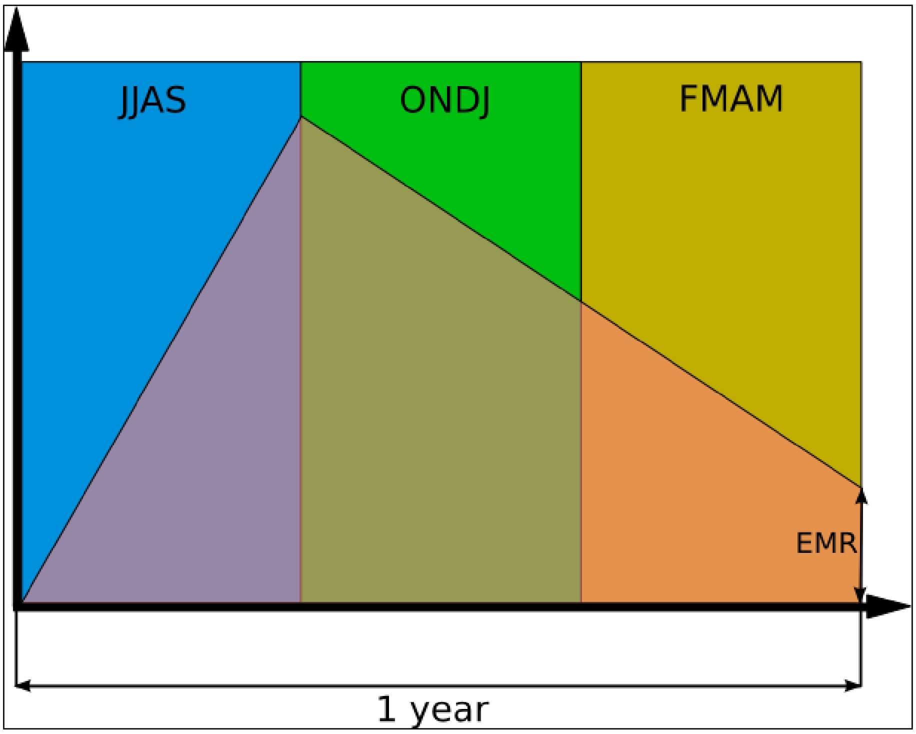

Figure 3.5: Figure. Modelled biomass availability through the year.

Agent planning

Each day the agent will update its knowledge about environment and choose an action to execute (the decision-making process is defined in the Submodels section 8). The list of available actions is:

1. Forage - The agent takes multiple walks of a bounded length computed from available foraging time. Walks are limited to the agent's Home range. From the visited cells, resource reward is retrieved based on biomass of the cells. The agent will halt the walk when reward achieves food needs.

2. Move home - The agent moves from its current home location to a new one within Home range. The new home settlement is chosen randomly between the dune cells situated in the richest sectors (containing the highest amount of resources) within Home range. Afterwards, a Forage Action is executed using half the available daily Foraging time of the agent in order to include the time spent on movement.

Adjustment of agent population size

This processes are executed for each agent:

1- Age. Agent ageing (increment human objects age).

2- Death. Every individual inside an agent will have a probability of dying. At the end of the year every individual within an agent must pass two tests to survive:

   (a) Natural death. Every individual has a 1.5% annual death probability, except during the first four years of life, when this probability is 10%.

   (b) Starvation. Depends on the capability of an agent to fulfil its caloric requirements. Every day the agent computes the percentage of needed resources that it was unable to collect. This 'starvation value' is accumulated through the year. The cumulative starvation value is translated into the overall percentage of full days of the year in which the agent was unable to gather sufficient resources. This percentage is translated at the end of the year as the probability for the agent to die of starvation.

3- Removal. If all the individuals that form an agent are dead, the agent will be removed from simulation.

4- Reproduction. At the end of the year every agent where both adults are still alive will have a 50% chance of having a new child.

5- Emancipation. An agent with individuals coming of age will seek suitable matches among agents within its social range. When two individuals coming of age from different agents join a new agent is created.

3.6 Design concepts

3.6.1 Basic principles

The behaviour defined in this model is derived from the Optimal Foraging Theory (OFT)[48], developed within behavioural ecology and Rational Choice Theory(RCT)[13] from economics. The main principle of OFT is the maximization of long-term energy gain. In other words, it is usually assumed that animals attempt to maximize the benefit to cost ratio. Evidence exists e.g. among great tits(Parus major), birds that show relatively successful strategies in terms of OFT. Although it is doubtful whether humans attain the optimal rate of energy gain, they do succeed in improving their foraging efficiencies, or 'memorising'. Also, RCT framework backs the decisions of not taking a choice that implies some negative outcome. The term Rational stands for balancing cost of choices in a way that maximizes personal advantage.

3.6.2 Emergence

The model explores the emergence of stable HG populations under different climatic conditions.

3.6.3 Adaptation

At the present moment the model is not interested on the emergence of individual adaptive traits, and for this reason adaptive options for the agents are limited to the decision making process. The different agents try to respond to the dynamics of environment choosing Home locations and Foraging actions depending on their particular situation. The aim is not policy discovering, just adaptive planning. Adaptive options for the agents are based on the complexity of the decision making process. Actions costs and outcomes are predefined following the model's configuration, but the order and the use of them is entirely an agent's choice. Following this reasoning, the different agents will try to adapt the dynamics of environment (both ecological and social) planning a different set of orders each time step. In a future step we will introduce in this model agents with mixed pool skills that are capable of choosing actions typical of HG and AP populations. Each agent will have different efficiency values for different actions, that will vary in time. In this way, the agent will also adapt specializing its actions.

3.6.4 Objectives

Following the basic principles stated before, the objective for any agent is the survival of its individuals. This assumption is clearly from optimizing the system, as the different populations won't be guided by the mission of 'colonizing' the entire landscape. Anyway, this outcome will be seen following evolutionary mechanisms and positive selection. Well adapted agents will have more possibilities to survive, thus creating more children and agents with similar cultural traits.

3.6.5 Learning

Foraging learning process is modelled using vertical transmission. A child gradually learns to forage in an efficient way, and for this reason the available foraging time contributed by children increases until adulthood.

3.6.6 Prediction

An agent does not keep track of previous rainfall values, so it is not able to predict the future state of the environment.

3.6.7 Sensing & Information Retrieval from Environment

An issue seldom addressed in the literature of ABM applications into Social Sciences is the fact that agents do not have perfect information on their environment. Home range limits the zone that agent know around its home location. Other issues about bounded rationality is treated in the smart agent section.

3.6.8 Interaction

The interaction between agents is currently limited to the *fission* process that is executed when two agents with adult children are inside social range, and to *information sharing*3.10.2 at the end of the day.

3.6.9 Stochasticity

Stochasticity is used in three different concepts:

1. Environment. Precipitation is calculated as a stochastic process following a Weibull distribution.

2. Outcomes. Some actions have different outcomes depending on stochastic processes, like forage and harvest. It encapsulates the complex process of resources collection (i.e. risk, variability, etc.), and it is important due to the fact that Actions will be chosen depending on their outcomes and risk of failure.

3. Life events. Death and reproduction are stochastic processes following realistic distributions.

3.6.10 Collectives

The agent, atom of the decision-making process, is itself a collective of different related individuals.

3.6.11 Observation

Population dynamics are the most important concepts to derive from the model.

3.7 Initialization

Initial state of the model is divided by entities:

3.7.1 Climate

Rainfall yearly precipitation is a stochastic value calculated from historical data[45] Calculated values depend on the initialization seed used in the random number generation, that is stored as a parameter of the model's configuration. Next parameters can be modified during initialization time:

EMR End-of-year minimum residual resources

AYP Average Yearly Precipitation

VYP Variance in yearly precipitation

3.7.2 Environment

Ground Model and land features are raster maps created from real data. The model is able to load any raster map with correct values. This process is done during init time from the file specified in the configuration.

3.7.3 Resources

The conversion functions that create available biomass from landscape and rainfall for each cell use parameters specified in the configuration. They are based on published research; nevertheless they can be modified in order to explore different plausible scenarios.

3.7.4 Agents

Several parameters can be changed from the configuration. These values are loaded during init time, and remain stable during the entire execution. This is the list of parameters used for this model:

1. Life-event related:
   (a) Adulthood age: 15
   (b) Dying age (HG/AP): Stochastic distribution
   (c) Number of agents
2. Resource related:
   (a) Home Range: 300 cells
   (b) Number of sectors: 8
   (c) Forage time cost: 30 minutes
   (d) Walking Speed: 3 km/hour

3.8 Input data

3.8.1 Rainfall

Rainfall (yearly precipitation) is the 'environmental engine' of the model. Data for precipitation rate are extracted from historical data (1871 - 2008). The climate engine is defined as a probability distribution, from which the total precipitation during a year is derived. The Gamma distribution was the best fit for the available rainfall dataset. The first approximation was the Weibull distribution. But we wanted to explore the parameters of mean and stddev in the rain fluctuations to experiment with stressful scenarios and verify the resilience of the agents. The characterization of the scenarios was prefigured by the mean and the standard deviation of the yearly rain. Weibull distributions are parametrized with an alpha and a beta parameter that would be found with a fit function of the platform R from the historical data we have. The connection of these parameters with the mean and the standard deviation were not simple formulae in the straight way we wanted. The second option, with an excellent match for the shape of our data, was a Gamma distribution3.6. Again, its parameters alpha and beta are found with a R command. This time, alpha and beta can directly be altered with perturbations derived from the mean and standard deviation explored in the experiments. Table 3.2 a comparison of the relationship of distributions gamma and weibull with the mean and variance parameters. The function Γ(Gamma) appearing in the column of weibull distribution has no relationship with "Gamma", the statistical distribution.

–	*Gamma*	*Weibull*
Mean	α/β	$\lambda\Gamma(1+1/k)$
Variance	α/β^2	$\lambda^2\Gamma(1+2/k)-\mu^2$

Table 3.2: Statistical parameters of Gamma and Weibull distributions. $\Gamma(n) = \int_0^\infty e^{-x}x^{n-1}dx$.

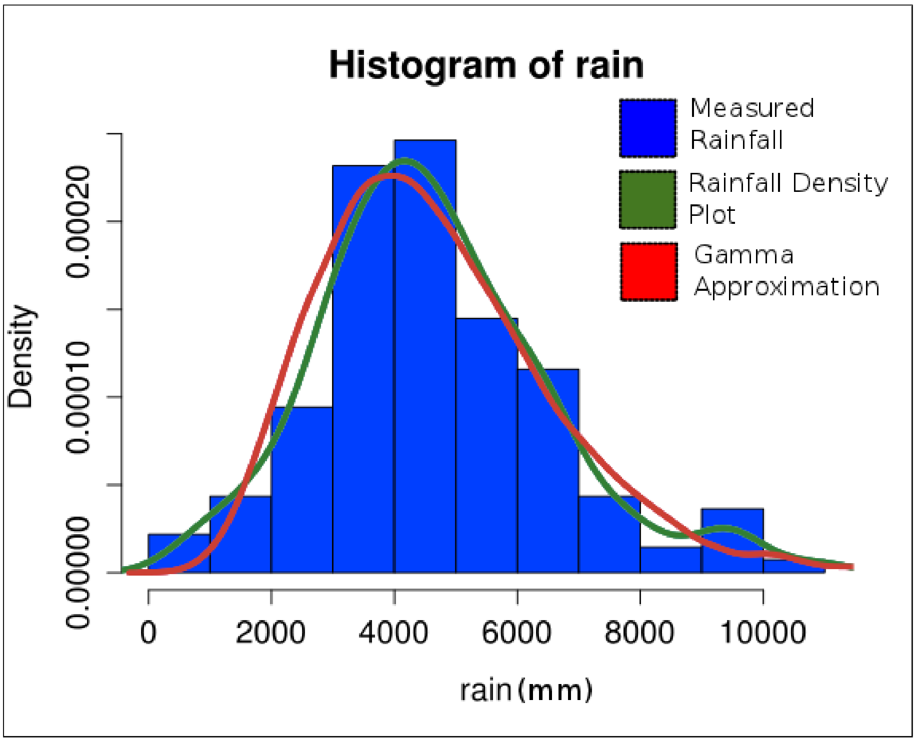

Figure 3.6: Matching of yearly rain distribution shape against gamma distribution shape.

3.8.2 Ground Model

This model is derived from LANDSAT and ASTER satellite imagery (combining pre- and post-monsoon imagery) and includes DEM and land features. Satellite data are transformed using unsupervised classification and clustered in the 3 classes (water, interdune and dune). The model is exported as a Raster map.

3.8.3 Hunter-Gatherer behaviour

Archaeological data are incomplete and limited in terms of derivable behavioural patterns. HG behaviour for the model was derived from published studies of historical and present-day populations in similar ecological settings. There are groups of HG that live near N Gujarat (the Van Vargis, see Nagar 2008). However, these communities have a high degree of interaction with and dependency from settled agricultural communities for their subsistence strategies. This occurrence constitutes a strong bias towards the use of these groups to model our HG agents. Instead, we used as surrogates of our HG population, African groups of the San communities. Among living and historical HG communities, the San (especially the G/wi and G//ana groups of Botswana) represent the best-fitting parallel in terms of ecosystem (Tanaka and Sugawara 1996). These groups are found on a flat plateau in the central part of the Kalahari desert. The landscape morphology is characterised by fossil rivers and traces of sand dunes. Rainfall is concentrated in the summer months with c. 400 mm annual average precipitation. The vegetation of the area is dominated by plants of the Gramineae family (grasses) and a mixture of shrubs of the same genus/families that are found in North Gujarat.

3.9 Submodels

3.9.1 Agent execution cycle

The majority of Agent-Based Models mix knowledge acquisition, decision-making and execution in the same phase of an agent's execution. This choice is useful if we deal with agents with simple decision- making processes where the choice of behaviours is predefined. However, this classical approach to ABM has a major drawback, and is the fact that the agent will have scripted strategies, and for this reason it won't be able to choose strategies different from the ones defined there. The model proposed here splits the different phases. During each time step every agent updates its knowledge about the environment (possibly in the future including other agents state). This action is combined with a set of possible actions, in order to choose which plan of actions will be executed. Several factors can be used to enrich the process:

1. Agent's goals and agent's preferences referred to the choice of particular actions

2. The information that the agent perceives from the environment and the reliability of such information.

3. The information collected from other Agents, as well as its reliability.

4. The feedback the agent receives from engaging into a given activity. In this particular model the goal of every agent is to maintain alive its individuals, and the potential actions are the ones defined in the document.

   This approach will allow to integrate more sophisticated Artificial Intelligence techniques into the current decision-making process, depending on particular research around this first model.

The execution of the agents during each time step is divided in three different phases:

Knowledge update The agent collects information from the environment, and creates an individual representation of the world using its preferences and objectives. Agents will calculate the amount of biomass available in each directional sector (utility score), as well as potential settlement zones.

Action choice The agent decides which actions to execute once knowledge has been collected, based on the following steps:

(a) Agents checks whether there is any sector inside its home range where resources can be obtained. This is calculated based on available foraging time and resources on cells.

(b) If needed resources can be obtained the agent will choose to Forage in one of the Sectors where this is possible.

(c) If this is not possible, the agent will choose to Move Home. A collection of possible new homes is created based on the quantity of resources inside the Home range from this new location. The final location is chosen amongst the ones that fulfil resource requirements.

Action After every agent has defined a plan, all of them are executed sequentially following a randomized order.

Information Sharing Once actions of the day are executed the pipeline proceeds with the information sharing phase.

3.10 Knowledge

According to anthropological registers, HGs explore the area around their homes following a eccentric direction. Just like a compass divisions, a HG has set of choice directions. Each direction leads to areas where some could be good for foraging and other ones could be poor in resources. Also, anthropological sources describe hunt parties. Sometimes males of the familiar groups join in a group of hunters and spend several days apart from the home hunting in the whereabouts of the home range. This first stage of our models is not considering yet agent coordination and cooperation, so we stick to simple hunt actions with one hunter (and grown children) with a few hours duration, which are subsumed under ForageAction.

We decided to endow agents with a map of cells of the world bounded by their home range. The circled area by the home range is divided in equal slices called sectors. The number of sectors is parametrizable though a configuration file. For each sector their cells are stored and an utility scoring is assigned. For foraging decisions the utility is the sum of resources of the cells that compose it and for the MoveHome action the utility is

Each time step the amount of resources in each cell changes due to the growing dynamics of biomass and to the possible foraging practices of other agents. The updating of utilities must be done each time step. The default update of utilities is omniscient inside the home range. The agent access directly to all the information inside the area of its home range, outside of it the agent does not map more cells. We assume the agent can read the last updated value in resources for any cell concerning the update procedure.

At this stage the agent does not store knowledge about the other agents.

3.10.1 Partial Information

Although real HGs have a direct access to their domains they cannot traverse it as quickly as to say they have last minute updated information from the environment. The model for HG considers a home range of 10km, a large extension for keeping an updated account of the resources in it. Each foraging exploration is a bet for the HG. Although he can know that in average one zone has enough resources to cover the survival threshold it does not ensures that this estimation is not going to change. There is uncertainty mainly due to other agents actions. The partial-information add-on emulates this bounded rationality and uncertainty feature. The update of the knowledge of the agents is calculated with two different sources of information. When an agent visits one sector in some time step the process will read the cells of the high resolution raster (direct access to the world) corresponding to the sector.

Utility values have a timestamp. Each low resolution cell in the private raster of the agent has associated a timestamp of *one week*. Once the timestamp expires the low resolution cells of the private raster receive an utility value equal to the statistical mean value of resources in the world raster, that is the second source of information. Summarizing, the utility in the knowledge structures of the agents is initialized to the mean, and as the agent visits sectors the overall information in the knowledge structures is renewed; information expires and then receives as utility scoring the statistical mean of resources. Figure 3.7 shows the pipeline of flow of information and utilities.

PHASE	DESCRIPTION
1	The planner accesses to the private low res knowledge structures of the agent.
2	The planner proposes one action and one sector where the action will occur.
3	The agent has a high resolution version of its sectors, these lists of high resolution coordinates to actuate in the world.
4	The agent explores the world which will be materialised as a walk over the matrix of resources of the world.
5	The procedure of updating the low resolution rasters registers the changes happened in the high resolution raster.
6	The agent can copy directly the utility of a cell in the low resolution raster to its private low resolution raster if such low resolution cell was contained in the sector that was visited in the last action.

Table 3.3: Explanation for steps in figure 3.7

3.10.2 Information Sharing

Information sharing and knowledge exchange is a crucial activity for performance enhancing on everyday challenges. In our study of resilience we try to explain the conditions that trigger or inhibit the change of social groups moving from hunting strategies to agropastoralist strategies. The most successful strategies will be passed between agents of our simulation in a cultural transmission modelized framework2.4.4. Our first results of agent communication are based on sharing environmental information concerning resource allocation in the environment. Our simulations show that simulating individuals with partial information3.10.1 of the environment, but sharing it with other neighbour agents, allows them to achieve a resource retrieval as good as if they would be endowed with complete perfect information of the resource allocation (omniscience)**??**.

We consider the introduction of social networks as an important factor for resilience as it acts as a safety network for unlucky agents affected by detrimental variability of environment conditions.A social network will be a source of help in the form of resources, environmental information and transmission of knowledge for survival strategies. For the first modelling of information sharing we stick to two anthropological evidences. HGs usually share information with other HGs living around a distance we have called social range, and many times they do that at the end of the day. Second, the information token is some resource information concerning some patch area which would correspond to the abstract mental con-

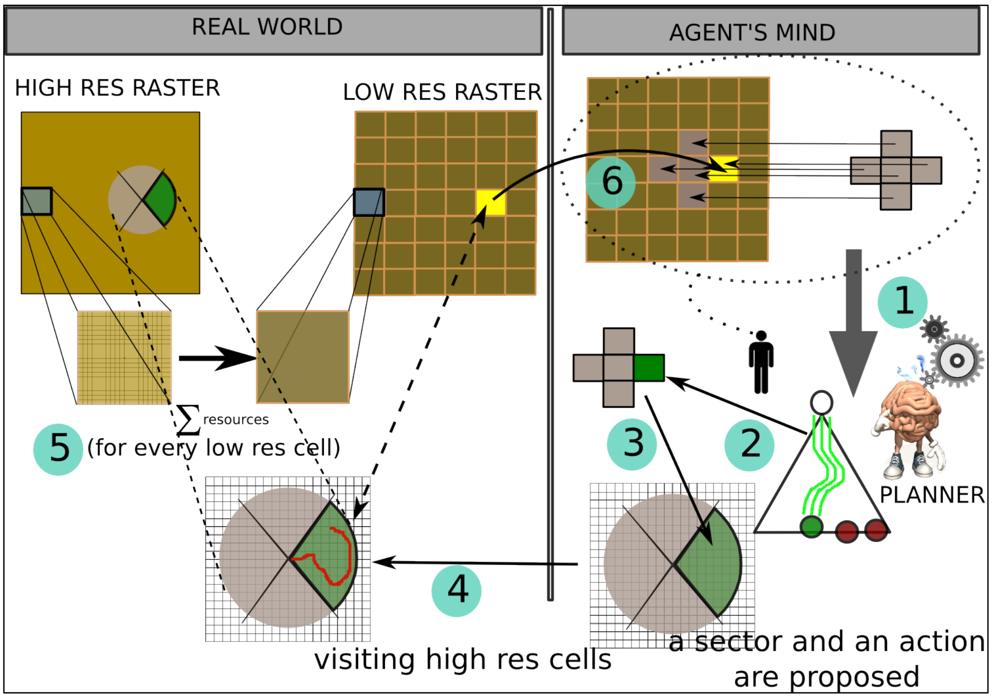

Figure 3.7: Pipeline of partial mind updating.

struct that inspired the low resolution cells for the mind knowledge representation of the agent. The agents will execute a procedure of information sharing at the end of the simulation step. For every agent inside the social range of an agent, each one will give the central agent an information token which is composed of a low resolution coordinate and an amount representing the utility of the low resolution cell shared. Each agent will send and will receive an information token in the interaction with the other agents. The information is accepted and written over the private raster only if the timestamp is newer than the owned one.

Results of the Information Sharing can be seen in the chapter of experimentation**??**.

3.11 Decision Making

3.11.1 Rule Based Agent

The rule based agent was developed as a representation of the classical agents in social simulation. The objective of having rule based agents is to compare its performance against the AI approach. The rule based agent follows a decision schema constructed by the modellers with advice from experts. The ingredients of decision are the set of actions (ForageAction and MoveHomeAction) and the set of zones where actions are performed, the eight sectors. The agent at each time step evaluates if there is enough resources in some of its sectors. Enough stands for allowing its family avoid starvation during one day. There is a survival threshold based on food-need parameters 3.4. The rules are expressed in the decision schema appearing in figure 3.8.

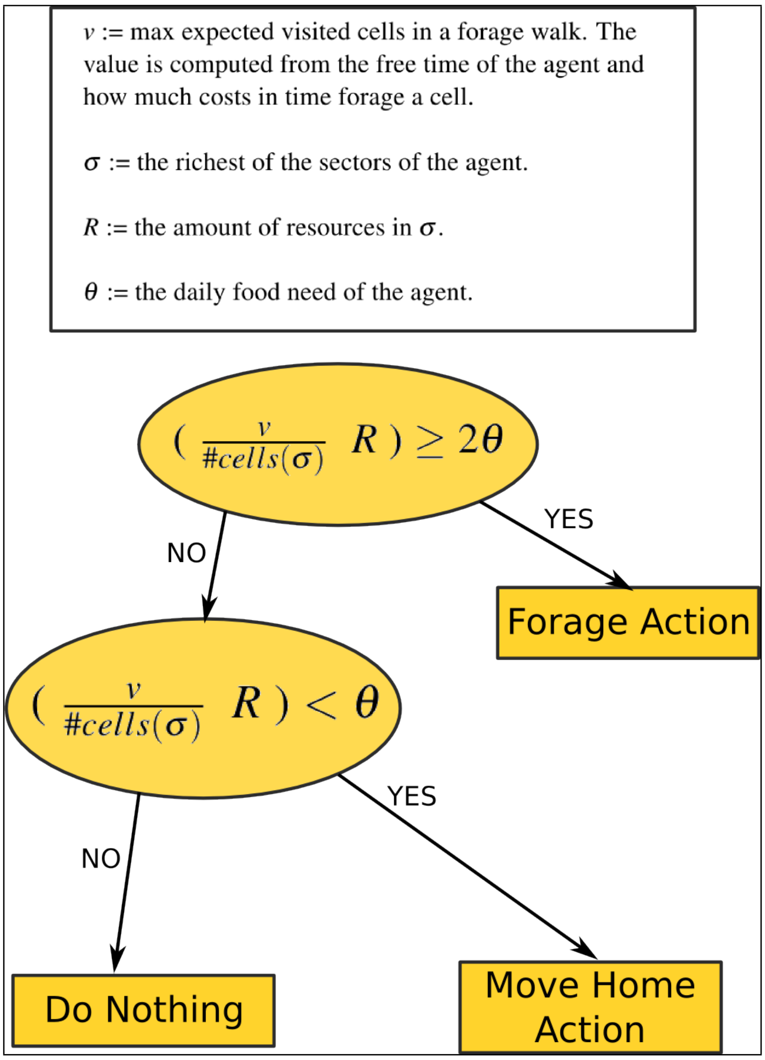

Figure 3.8: Decision tree followed by the rule based agent.

3.11.2 Planning

Previous sections have introduced the necessity of applying planning3.6.3 in the decision making of the agents. According to the aim of the project the agents face changing conditions that compromise their adaptability. Rule based decision making is associated to a hard wired knowledge. Something that works for a situation. If conditions change rules serve no more. A static and closed knowledge encumbers the agent's adaptability. The dimensionality of the world due to climate and agent interaction would make appear non predicted states making the agent receive a lower benefit from its actions.

Rules are a mechanism that is applied blindly. Action is executed after conditions on the world are evaluated. For adaptability, when thinks change in the world, rules expire. We need to now the new conditions. So, the new correspondence action - state. Although we are talking of change we assume that some bottom-essential mechanics remain unmovable. Agents will adopt an exploratory framework. Agents will use a hypothetical-deductive frame for decision. This is the "what-if?" question. An agent will test an hypothesis (launch some action a) and will try to predict the outcome or the state produced by the action. The nature of the problem of resource retrieval in the raster lattice does not allow to develop a logical model to reason about states and actions. We cannot provide facts about state of the system and a logical model of the transitions of the system.
We will follow an utility based architecture as said before. The agent has to tackle with resource optimization, large number of uniform possible cells for exploration and with associated uncertainty. We think it is more reasonable to take an optimization and heuristic search approach instead of a symbolic reasoning based one to synthesize the agent plan.

The uncertainty of the world, the fact that agents make their decision on current time step world state and decisions are taken sequentially defines the decision making as a Markov Decision Problem(MDP) [8, chapter.17].
MDPs are characterized by the markovian property, by some utility function and a transition of states. A MDP is represented by a directed graph3.9 where nodes are states of the world and edges denote and action performed. Traversing an edge means that an action transforms the state of the world from the state associated to the origin node to the state associated to the target node. MDP are solvable through solving the Bellman equations. The solution is an association of states to executable actions that shall be the policy that an agent will have to follow. A Bellman solution is too near to a rule based approach. The dynamic of events and indirect competitiveness for resource acquisition is an evidence to find a technique that solves scenarios like board games. State transition in board games depend on the current state. They work launching sequential actions. And boards exhibit a large number of configurations.

UCT

UCT [52] [53] is an algorithm that finds an expected optimal action to launch from some state of the system that is explored by the algorithm. UCT receives as input a state of the system and explores the outcome of applying actions to the input state. UCT expands a search tree applying actions to nodes that represent states. The strategy of UCT is to find the trace of actions that in long term returns maximum profit. UCT applies CPU power to promising traces. Traces are non deterministic sequences of actions and states. MDPs have probabilistic transitions and one action applied to a state can lead to different states with some probability distribution. Hence, UCT must apply a sampling and produce a statistical scoring for states and traces. The further a state is from the initial node the lesser its statistical significance will be. For the sake of statistical relevance and bounded resources the search tree has a maximum exploratory depth; this is a parameter for UCT, the horizon. The number of samples for a trace is called width. The winner trace indicates which is the next action to perform. UCT has provided good results for board games and a very respectable scoring in the japanese game *Go*[57].

UCT(s,d): s is current state; d is remaining steps to depth bound; G is a explicit graph, initially empty; π is base policy; C is exploration constant;

if *d=0 or s is terminal* **then**
  return 0;
end
if *node(s,d) not in G* **then**
  add node (s,d) to G;
  $N(s,d)$:= 0;
  $N(a,s,d)$:= 0 for all $a \in A(s)$;
  $Q(a,s,d)$:= 0 for all $a \in A(s)$;
  Obtain sampled accumulated discounted reward $r(\pi, s, d)$ by simulating base policy π for d steps starting at state s;
  return $r(\pi, s, d)$;
end
if *node(s,d) in G* **then**
  $Bonus(a) = C\sqrt{2\log(N(s,d)/N(a,s,d))}$ if N(a,s,d)>0,else ∞ for each $a \in A(s)$;
  Select action a = $\mathrm{argmax}_{a\in A(s)}[Q(a,s,d)+Bonus(a)]$;
  Sample state s' with probability $P_a(s' \mid s)$;
  $nv = r(s, a) + \gamma \mathrm{UCT}(s', d-1)$;
  $N(s, d) := N(s, d) + 1$;
  $N(a, s, d) := N(a, s, d) + 1$;
  $Q(a, s, d) := Q(a, s, d) + [nv - Q(a, s, d)] / N(a, s, d)$;
  return nv;
end

Algorithm 4: UCT algorithm([53])

UCT explores the board in a hypothetico-deductive manner desirable for our purposes. We could see that a new node to explore is a open hypothesis. And we can see the deductive part when UCT executes the hypothetical action against its private virtual copy of the world. Like in game Go, UCT will be executed to guess next action to launch; UCT is not used to produce a policy to be used the rest of the simulation. Each time step UCT will have to recalculate the most promising possibilities for each action on each patch where it will be applied.

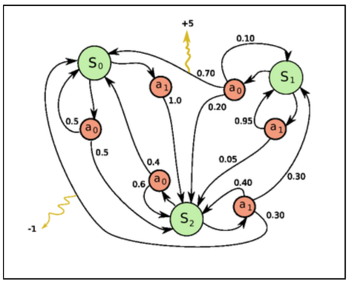

Figure 3.9: Markov Decision Problem automata.

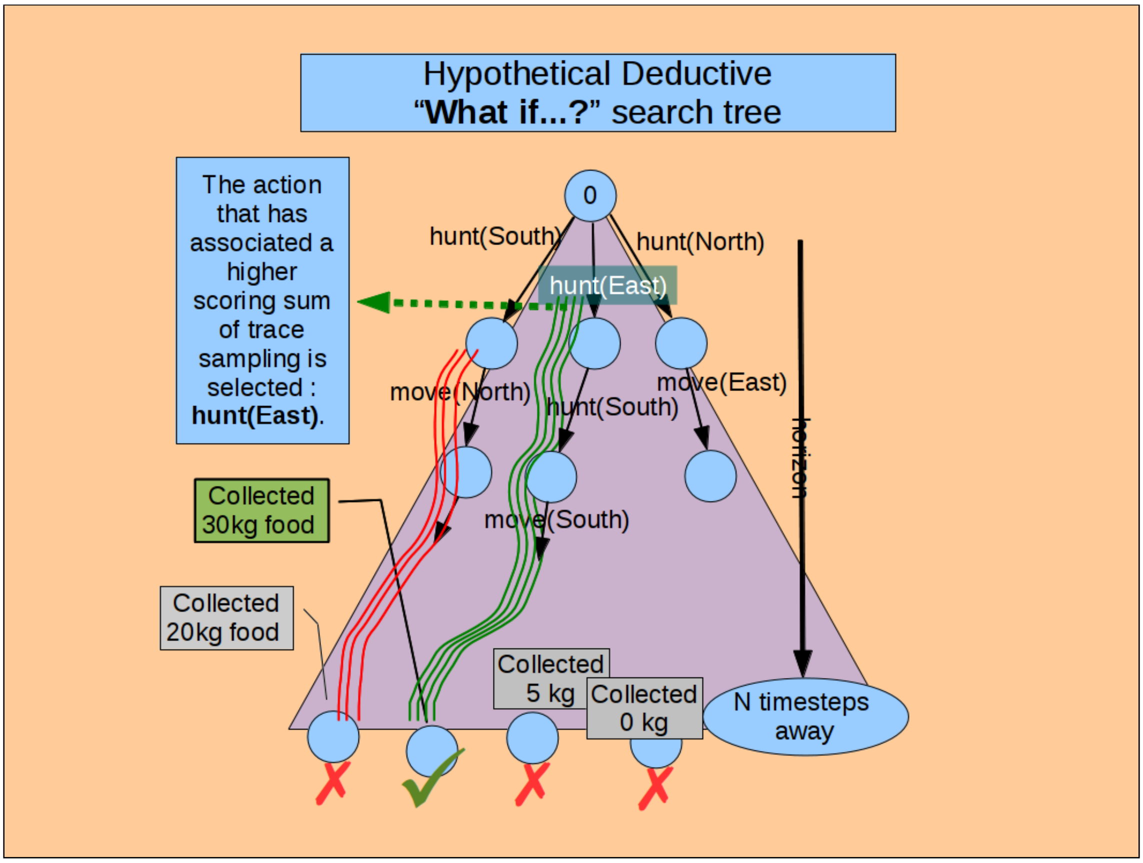

Figure 3.10: Figure. Uct search tree.

Reducing States of Search Tree

Recall that UCT is a planning algorithm through statistical sampling. Recall that each node of the search tree corresponds to a system state and that nodes generate their node offspring to fill the next level as the result of applying a stochastic process. This would correspond to the stochastic process of change which has been produced by the effect of applying an action of the agent. States are stored in a dictionary and can be revisited if another node induces a new state that is identical to one that already has been generated in the past (Fig. 3.11). Revisiting a state changes its weight and it guides probabilistic search and selection of actions to continue generating the range of nodes. The selection of actions affects the generation of traces that lead to final states. After executing all the amount of shots set by the width parameter, the algorithm has performed a statistical sampling of the different states and utilities in tree leaves that end up guiding the decision.

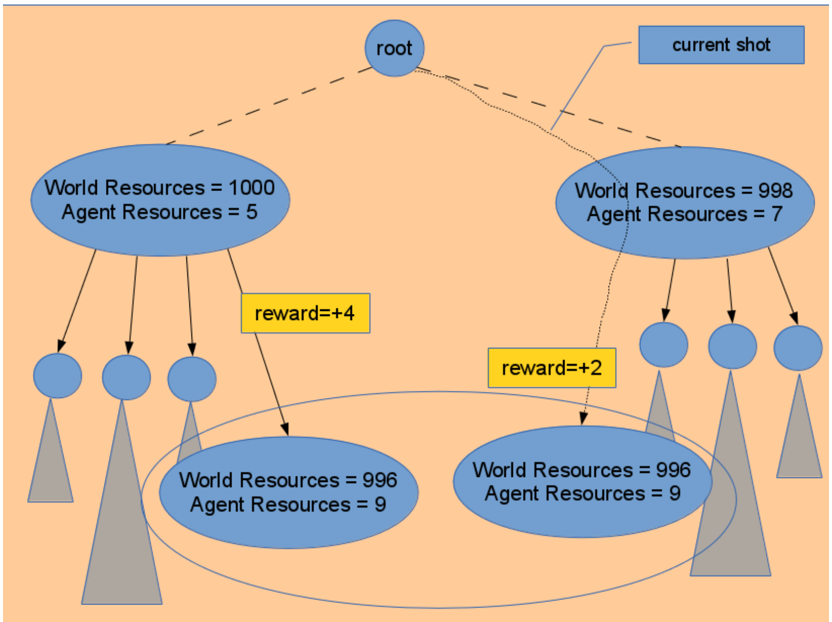

Figure 3.11: Past states can be revisited due to stochastic effect of actions in the UCT search tree.

It is important to make relevant sampling and to crystallize significant probabilistic weights along the tree nodes, from the statistical point of view. The sampling should allow to recover an approximation of the distribution of states and utilities to filter the states which are preventable from which are desirable decisions. To visit and sample correctly traces is a need for the update factors of the underlying reinforcement learning that is done based on the sieve of the found positive and negative rewards.

In the case of the model for Hunter-gatherers, states are characterized by attributes with a wide spectrum of values. Just to mention it the amount-of-resources is the core measure of this distinctive feature. The environment of an agent is divided into sectors where the amount of resources for each one can sum up to hundreds of miles of units of biomass. If the procedure in charge of declaring two states are alike or different is based on a strict comparison of this amount, most likely, any two nodes are going to be classified as different. Taking two states where everything is the same except for the northern sector where there are

899,999 registered units in one state and the other one has an amount of 900,000, the algorithm will see both states are different and the difference is only one unit of measure of a magnitude many times over. From the standpoint of this agent they should be considered in short-term identical because they were for survival purposes equivalent. From the point of view of the UCT algorithm, if the probability of equality between states is much lower (we trust that stochasticity after applying the action makes values match exactly by chance), part of the heuristic factors and strategies from the algorithm are not applied. The third conditional branch in code(appendix. 3) will run just for a few nodes and it will hardly generate the count N and Q to follow the preference for rewarding promising nodes that lead to a successful trace. Then, the node will count for UCT, most of the times, as a new non visited node and the update of heuristics will be empty. The multiplicity of states emerging due to the low probability of synonymy between states will make sampled traces to be disjoint between themselves. The distribution generated is very flat because you can not make the connection between state, action and effect of the factors of Q-learning. All UCT execution will be absorbed by the branch of code that executes the basic policy, usually a random sampling.

In order to stimulate synonymy between nodes, some changes were needed in the representation of data from the node states.

Value Range Reduction and Categorization The idea is to encompass the same information for the decision process but in a way we reduce the range/domain of the variables. Attributes related to agent resources and environment resources are simplified. The amount is converted to average amount of individuals sustained for a unique day by the resources. If the original value is 400.000 units and an individual needs a mean of 2000 units to survive one day, the new value will belong to a new reduced range and its value will be now 200 individuals per day, 200 daily rations. The value is reduced one step further. If an agent is composed of 4,6,10 individuals, eventually some reduced amounts also become indistinguishable for survival. The second reduction is a categorization we have seen works for our experiments (section 4.1). The final assigned value to the resource attributes of the nodes is one of the categories in the table (tab. 3.4).

CATEGORY			
0	0	$\leqslant$ rations $<$	2
1	2	$\leqslant$ rations $<$	15
2	15	$\leqslant$ rations $<$	40
3	40	$\leqslant$ rations $<$	100
4	100	$\leqslant$ rations $<$	∞

Table 3.4: Second reduction, categories assigned to first reduction.

Manage Multiplicity due to Action Stochasticity Each node has actions produced depending on the information of the world state in the trace it is being explored. If actions are produced stochastically, same attribute sets characterizing the mensurable part of the state will see associated different actions sets and hence it will yield two different states which will end up not being equal(synonyms). We force that for a state a fixed amount of actions is generated. If we allow to a state be associated to four move actions and there are eight locations to choose for movement, the association must be done throwing the dice and choosing four sectors from the eight possibilities to give four actions to the state. This increases stochasticity and multiplicity of states due to the combinations of taking four items from a set of eight. The solution is that all states open as many move actions as sectors there are, and the same for forage actions. And the only features involved in node matching are the relative time-step inside the trace, the resources of the agent, the resources of the environment, the location of the agent and the amount of accumulated days of starvation.

The results of applying state reduction can be found in the section of experiments (sect. 4.1)

Divergent Trajectories

3.12 Efficiency

3.12.1 Low Resolution map

In our experiments agents have been working with two different engines for decision making. The planning engine produces hundreds of millions of access to the cell storages in the search of profitable traces of actions. Each state in the search tree requires updates and many calculi managing the cells in the sector visited by the action. This represents around 40,000 cells per sector to manage for a home range with a radius of 300 cells. Holding this amount of information and accesses in a simulation with 100 agents implied traces that lasted more than eight hours of computation per year simulated. This is not affordable for the scales we need to simulate. Some real world cultural transitions last about the order of hundreds of years.

The direction taken to reduce the load of the planning synthesis was to reduce the resolution of the representation of the world in the mind of the agent. Indeed real HGs when think about profitable patches in his domains do not have a high resolution representation of the resources. They have fuzzy mind maps of more or less large rich areas. They do not think with a granularity of few meters when planning the day. They think about patches that would cover several high resolution cells of our model.

Each agent has private lists of references to low resolution cells, the sectors. The low resolution cells belong to a shared raster. Each low resolution cell covers an area of 60x60 high resolution cells. Each time step, once the high resolution raster world receives all the updates, the utility scoring of the low resolution cells and minds(sectors) of the agents is updated 3.12[1].

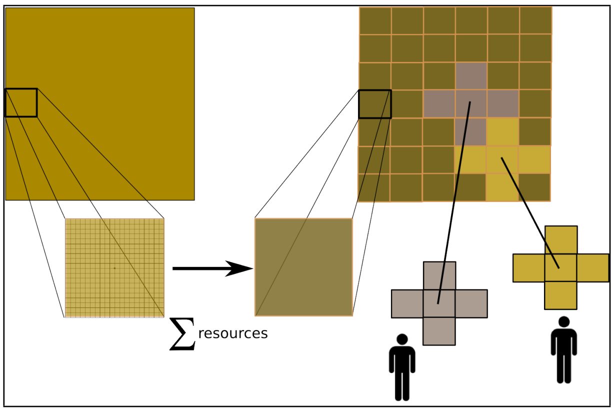

Figure 3.12: HR vs LR correspondence.

The utility of a low resolution cell is the sum of resources of the covered cells. Now the sectors in the mind of the agent contain less items. These low resolution structures are the only information resource used by the planning engine. No more high resolution access is done. When the planning engine simulates the effect of actions in the virtual representation of the world it processes a low resolution cell iterating over a formula to emulate the steps the agent would do visiting the high resolution cells. So, at each state of the search tree, the update of utilities happen in internal low resolution rasters and low resolution cell containers. The produced plan is a action to be performed in a chosen sector. The agent has also a representation of sectors with high resolution cells referenced. The sector and the action proposed by the planner are launched normally exploring the high resolution space as explained before in section **??**.

3.12.2 UCT efficiency improvement

UCT replicates the world dynamics to generate the results of exploring some action in his internal search tree. UCT has a private copy of the raster and a private copy of the knowledge of the agent that runs UCT to guess the next action to launch. Each time the search moves forward to a new state in the search tree the costly procedures of “update knowledge” are computed over the private copy of the agents knowledge. Some agent actions does not need a really exhaustive update. The solution was to reuse knowledge between search nodes through referencing shared knowledge structures stored in a dictionary. Avoiding all the unneeded calls

[1]The resolutions shown in the figure does not correspond to the one in the model. It is a even lower one to make clearer the graphics.

to knowledge update and memory operations of creation and destruction of objects implied almost and order of magnitude of improvement.

3.12.3 Throughput measures

Number of UCT agents	seconds per time step
100 agents	120 s/step

Table 3.5: First throughput measures.

Number of UCT agents	seconds per time step
500 agents	2.7 s/step

Table 3.6: Throughput measures after efficiency upgrade.

We can simulate 1000 years of 500 sophisticated UCT agents in 11 days of CPU.

Chapter 4

Experiments

The chapter of experimental results follows part of the trajectory of the experience of comparing the performance of the AI agents versus the classical rule agents. The first results are related to the first experiments of rule based agents in the model for the Gujarat[1]. Taking the same setting, mdp agents were put to test showing us that they could not improve the rule based agent resilience capability. Multiplicity reduction of the planning search states(p.65), compacting the range of values and limiting stochasticity proved to be a good solution to solve it. Results are shown where it is obvious that starvation is lower for mdp agents. Afterwards, some odd trends in the behaviour of parameters of the decision making process motivated another analysis of the model for the knowledge layer of the agent (p.68). Managing the change of biomass due to biological growth and decline of resources is introduced in the mdp layer of the agent. Following, results showed that tackling with resource grow/decline prediction contributes with only a slightly upgrade of starvation rates. Next experiment tries to set a lower bound to starvation rate based from biomass guessing and optimality of resource retrieval(p.77). The breach, that marks the distance to the ideal foraging, seems only to be explained by uncertainty an stochasticity effects, and not a big advance can we get from refining further the biomass prediction procedures. Many of these experiments are run under an scenario where the agent is alone in the world. One of the motivations is to discover faults in the foraging and migration patterns and verify the modelling decisions. That is the reason that introducing divergence management due to neighbour presence (multi-agency) is left for the next step of this model in the project. But anyway, although there is no multi-agency awareness we have tested the agent in a scenario of indirect competition for resources with other agents.

Three more packs of experiments follow to complete the chapter now focused on the differences between AI agents and classical simple agents. First there is a straightforward comparison of annual starvation rates between the random agent, the rule based one and several configurations of mdp agent. The next experiment is the extended version to a ten year trace. And finally there is an exploration of the relevance of adding more iterations to the deliberative engine of the agent. We observe a consistent advantage of the mdp agent over the rule based agent. We will expose the evidence in the numbers and also in the patterns of mobility produced by the logic of the rules. Following the trace of actions and change rate in starvation we could glimpse that the mistakes of the rule based agent were directly related to a problem of inability to adapt to two time lapses in the year were opposite strategies were needed.

4.1 Tuning the planner (State Reduction And Statistical Significance)

This section exposes the first results and issues of the tests of the mdp agent under the same conditions that the rule based agent was run[1]. An Agent Based Model(ABM) of rule based agents was designed to study resilience and persistence of Hunter-Gatherers(HG) in the north zone of Gujarat, a north-east province in India. The topic question was rooted in the premise that the main factors conditioning the presence of HG were related to climatic changing conditions and its effect on resource availability. Simulations were run taking the climatic conditions of three spatial bounds, the zone of Kutch-Saurashtra (north Gujarat), the region of Gujarat and the average rainfall of the whole continent. The combination of the regions, the extrapolation of climatic parameters to four selected time lapses in Holocene plus the intervals of exploration of the rainfall unfold the simulation scenarios. The analysis shows that historical decrease of average rainfall along millennia is not enough explanation to answer the question about extinction of HG. On the other hand, variation associated to yearly precipitation played a strong role on population dynamics as the major cause for group collapse if a threshold is surpassed. Although, the conclusion is open to consider feasible that other factors coocurred with rainfall variation in the decline of the HGs.

Against our intuitions the mdp agent could not break the starvation rate of the rule based agent. Statistical sampling of the starvation rates showed us a performance equivalent to the rule based agent. Furthermore, the exploration of different parameters in the planning engine did not produce improvement and results were unstable. We saw agents with deeper future prospecting and greater sampling in the search tree to perform sometimes worse than other configurations.

Fig. 4.1 is a sample simulation of one year run illustrating the performance of mdp agents versus the rule based agent in the scenarios from the above paragraph. Agents able to prospect three days in advance were not able to surpass the one day time frame of the rule based agent, and agents with a horizon of six days got only a small advantage compared to our expectations.

The problem lied in the internal representation of the world for the search nodes of the mdp layer. The representation scattered the states due to the wide domain of the features. It produced non matching states and led to search traces with low statistical significance needed to distinguish preferable traces from the harmful ones for the agent. The representation of the world states was reduced categorizing

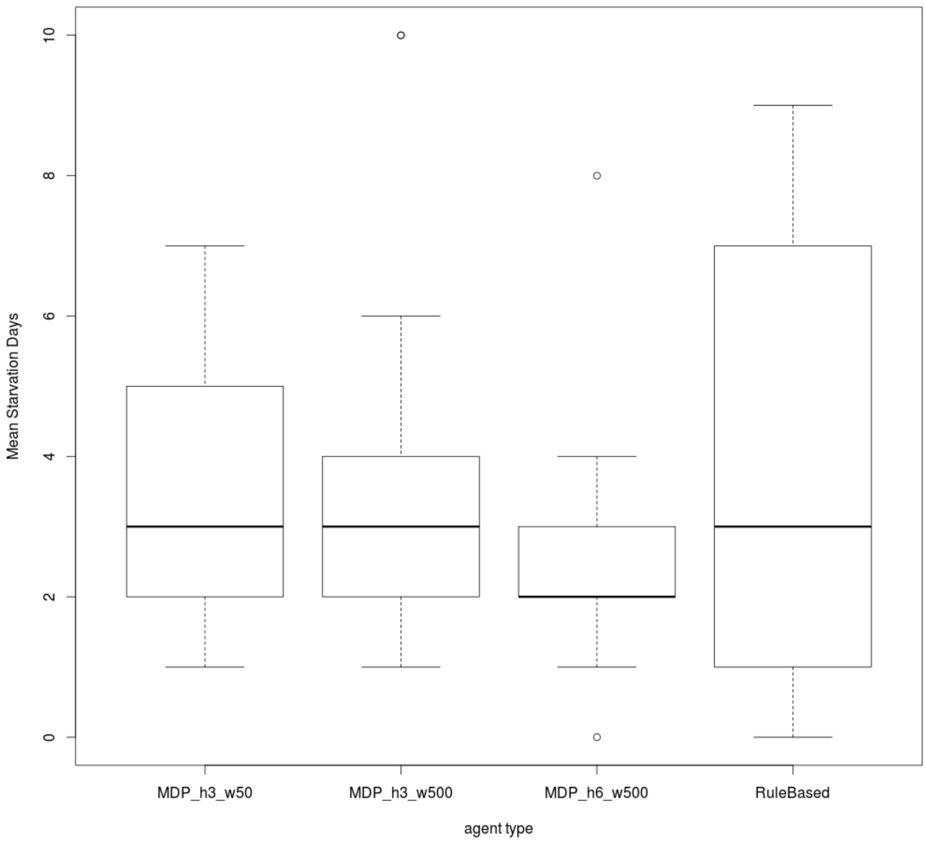

Figure 4.1: Comparing Rule Based Agent with MDP agents under North Gujarat rain condition, 4000 rainfall units/year.

numerical features and reducing stochasticity at the children node expansion as the corresponding section illustrates (p. 65). After state reduction the results took the direction we were expecting. The figure 4.2 corresponds to an exploration of the rainfall in the interval from a quarter of the mean to the mean of rainfall in KS-Gujarat scenario. For each value of rainfall in the axis there are ten runs with an associated starvation rate distribution as output measure. For each set of runs we register the mean of starvation rate and represent it in the plot with coordinates the rainfall and starvation rate involved. Each of the dots is connected with lines to its adjacent neighbours to reveal visually the functional relationship between the variables. The shape exposes an exponential growth of starvation as agents are deprived of resources due to scarcity of rainfall. For the mdp agent the slope is not so high and the proportion is quite a few times lower than the rule based agent. Also the shape of the starvation along the axis is far less bumpy, meaning that the response of the agent is more uniform and stable. Less variation in getting the resources means robustness to change and hence we can see it as a clue of greater adaptability.

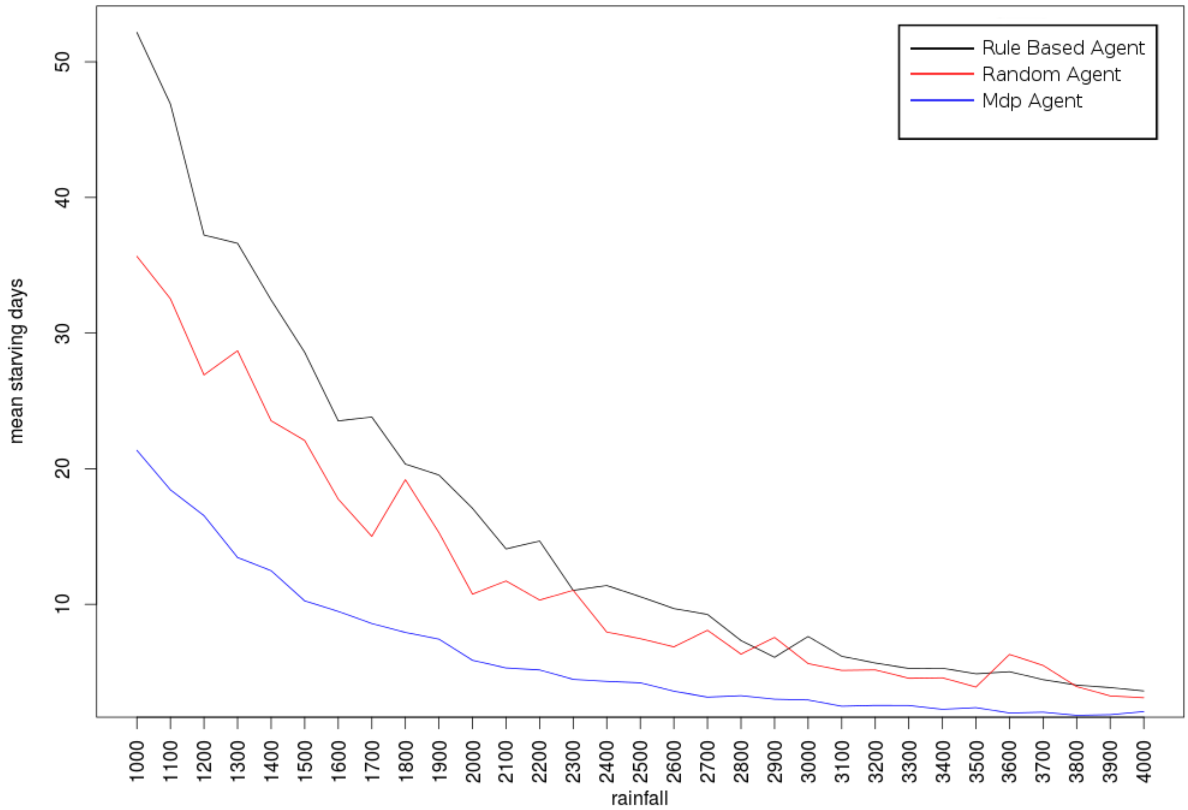

Figure 4.2:

The plot shows an extra agent labelled as Random. This agent chooses an action with uniform and equal probability from the combination of the actions and sectors where the action can be applied. This agent was introduced to have its performance as reference in the comparisons.

Below, you can find a detailed explanation of the results discovered for the rule based agent and why they are worse(p.81). The enumerated reasons are related to the cause why we have the unexpected result of the random agent's performance being better than the rule based one. The conditions of movement of the rule based agent polarize in two results tied to the two critical moments of the year, the beginning and the end. For one case, the agent launches more movement actions than what is advisable; for the other case, the agent delays too much the next move action to launch. The random agent with its uniform distribution of actions does not replicate the same movement patterns that harm the rule based agent avoiding the penalties in starvation.

4.2 Divergence And Biomass Prediction

Having managed to offer a better response than before (Fig. 4.2), the next step was to explore the best parameters for the simulation with mdp and find a lower bound to the achievable starvation or a feasible balance between depth of reasoning and CPU time.

Our simulations explored the parameter horizon and width of the UCT algorithm[53]. The expected result was to find how starvation decreased as we increased horizon and width in parallel. We did not paired low values of width to the higher values of horizon of the exploration set. UCT follows a search tree of states where depth is the horizon parameter and width the amount of traces that are launched against the tree to retrieve a sample of its leaves to produce a eligible set of future desirable states. Horizon values in exploration must grow paired with width. Low values of width retrieve poor samples from a huge pool of leaves if horizon is big enough.

One part of the simulations helped to discover that for higher values of width and horizon we got no improvement in resilience in comparison to other runs with lower horizon. Also, for a same horizon, greater width had a negative impact on starvation rates(Fig. 4.3).

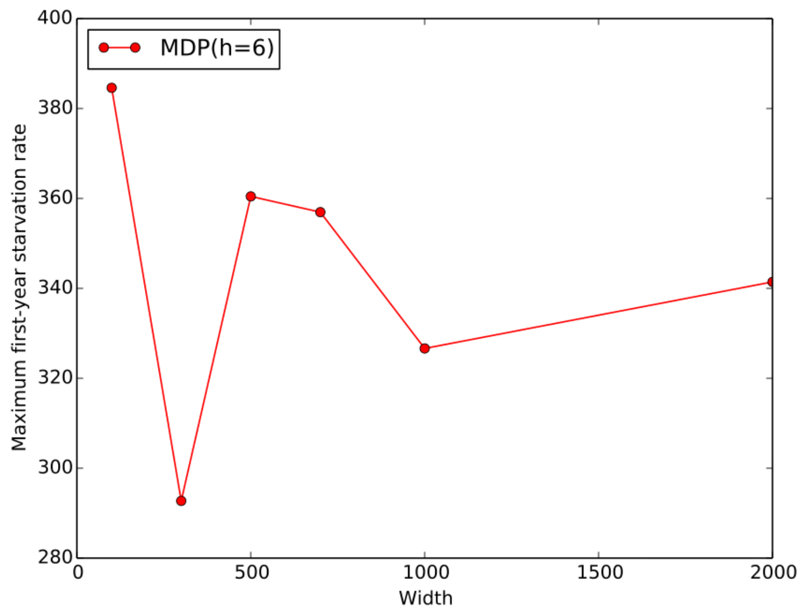

Figure 4.3: Exploration of width parameter showing non monotonicity.

The same happens in weather forecast. As we put farther the day we want to know its weather, the worse is the prediction. Weather simulation works over a representation of the world. There are not perfect measures, all the variables are not taken into account to design the model and at the end, they sum up in the accumulated error, step after step of the simulation. This leads to a state which is different to the one that will happen. We call this divergence of traces(p.68); divergence between the real trajectory and the simulated trajectory. Divergence happens because we introduce some error or because the model is incomplete; indeed, it is always incomplete.

UCT uses an internal model emulating the real world model in order to explore the choices that must offer to the agent. The model that UCT executes in this decision making process does not take into account two important points of the mechanics of the environment. First, each time-step, resources automatically grow or decrease depending on the season of the year; they do not remain constant. Second, the agent is not aware of any neighbour present in its home-range.

UCT launches a path from the root node to one leave node generating a search tree on-the-fly. There will be as many shots as the amount assigned to the width parameter. Each step, from one parent node to a child node, is executed as a simulation over the structures that are the representation of the world inside the UCT process. If the dynamics do not apply growth or shrinkage of resource due to climate or resource depletion exerted by other agents, there will be incoherence. The state will register a false amount of resources. The procedure will continue along the path towards the final leave node assigning deviated utilities to states that should have a different quantity of resource. An incorrect assignment of utilities in states leads to misinformation in the search process, an a bad classification of the traces considered as a good prediction of what will happen if the sequence of actions associated to the path is executed. Misinformation leads to divergence which will lead to loosing predictive power in the UCT. If you increase width you increase the weight of diverging states because you are repeating the same faulty procedure time after time loading of statistical weight to bad scored states. This way, width does not imply to be more informed. Increasing misinformation implies more uncertainty, and more bad choices for the agent.

Experiments in this section test the benefit from adding biomass growth prediction to avoid divergence due to bad biomass estimation in the planning process. Equivalent scenarios were applied to search for a way to solve the phenomena announced above. The experiment tests one single agent for a year against three different rainfall conditions. The first condition uses 500 rainfall units to set an environment that will ensure a noticeable amount of starvation rate. The next scenario has a rainfall of 1500 units to have an intermediate stage between the first and the third which reproduces a mean rainfall similar to Gujarat, 4000 units. The experiments register the starvation rate as an indicator of better performance of the agent that applies biomass prediction. We have labelled the agents as Guessing and NoGuessing in the plot, the former corresponding to biomass prediction to avoid divergence, and the later corresponding to not applying biomass growth prediction. For rainfall with 500 units, there are few resources and it is hard to mark a difference because there is no space to exhibit a big exploitation of the terrain (Fig. 4.4). With rainfall 1500 the improvement is less than 1% (Fig. 4.4); and with 4000 units of rainfall the starvation falls only a 5% of the value (Fig. 4.4), from 1.55 to 1.475 as an average. Is it possible to devise a way of taking more profit from biomass prediction? There is any fault in the way we applied prediction in the states of the planning procedure? Next section discusses and reasons about the causes of the little effect of applying biomass prediction in solving the biomass divergence.

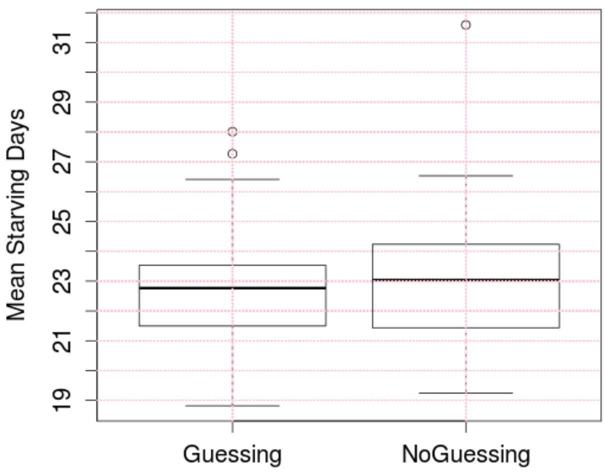

Figure 4.4: Mean Starvation days for an agent that predicts biomass grow and decrease versus an agent that does not apply a grow/decrease factor during the decision making. Experiment run with 500 units of rainfall.

4.2.1 Finding the limits to Improvement by Biomass Prediction

The results of the experiment in the previous section, contrary to the expected, do not show a substantial improvement when the agent predicts biomass when compared to the agent that does not use prediction. Clearly, the depths of exploration we use, three, six, ten days forward, do not allow to accumulate an excessive divergence because there is not enough length in the trace. In particular, when the agent is at the end of the year, when the agent exhibits a greater movement rate, fewer days are spent camping in the same settlement. The resources in the location are perceived along a very short segment of time. And reasoning about resources in a small window of time should not involve critical divergent traces. But on the other hand, decline of biomass moves forward and it is logical to expect that without biomass-prediction the planning process will suggest to move when it is too late and you will fall into the error of waiting too much in a poor resource area. One of the two possibilities must be discarded. We could think that the procedure for

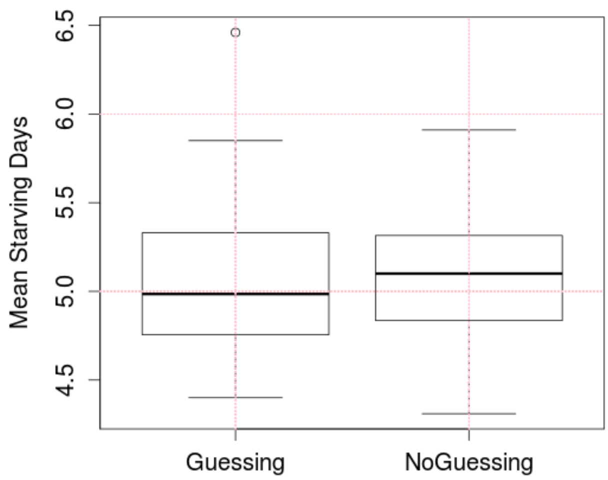

Figure 4.5: Mean Starvation days for an agent that predicts biomass grow and decrease versus an agent that does not apply a grow/decrease factor during the decision making. Experiment run with 1500 units of rainfall.

predicting biomass is not sufficiently accurate or incorrectly focused.

The experiment of this section was designed to assure ourselves about this matter. The target is to attempt to establish a lower bound to the starvation rate. We will see that there is a gap between the profile of an agent with biomass-guessing and the profile of the lower bound. The lower bound is given by a modified mdp agent whose foraging actions although return a reward, it does not deplete the resource patches. The experiment executes a normal mdp agent, a no-depletion agent and a modified simplified setting, let's call it the "gap agent". The idea is that the simplifications allow to jump the gap of the difference between the starvation rates of the normal agent and the no-depletion agent. The intuition is that this modifications will be related to why the mdp agent cannot get nearer the lower bound. The modifications are detailed next. The normal agent receives normally distributed reward from the forage actions in the real world, and in the planning process. When a move action is launched the reward of the day is halved compared to a foraging day. The gap agent does not have stochasticity in rewards and receives full reward from foraging actions all days. This modifications are related to uncertainty in reward and uncertainty in appropriate election for a move action. Foraging actions without stochasticity produce better prediction of rewards in the planning part of the agent. Removing uncertainty and seeing that the starvation rate for the gap-agent matches the starvation rate of the no-depletion agent will tell us that the gap can be explained by uncertainty non related to biomass guessing and the divergence

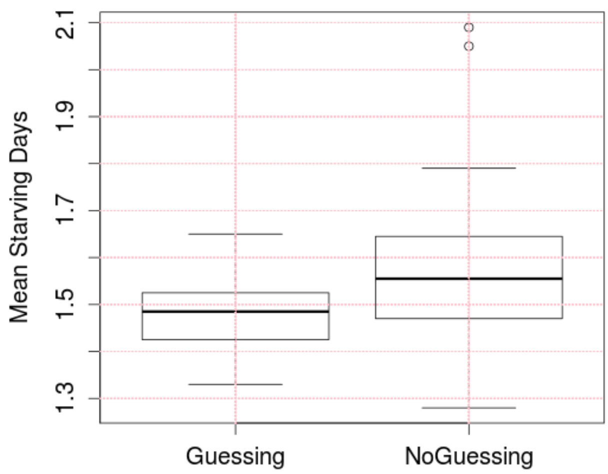

Figure 4.6: Mean Starvation days for an agent that predicts biomass grow and decrease versus an agent that does not apply a grow/decrease factor during the decision making. Experiment run with 4000 units of rainfall.

problem.

The way of producing the lower bound tracing the no-depletion agent has the following reasons. The track of the interaction of an agent with the world is a sequence of actions. Along the simulation, the agent produces a list of move and forage actions. Usually we will see sequences of forage actions between move actions denoting the pieces of time that a settlement occurs in a location till the next movement. The challenge for the planning layer is to propose movement actions in a way that the intermediate forage actions can always retrieve a maximal amount of resources. The ideal situation would be one were the foraging actions will endow the agent with enough resources to achieve the metabolic needs. And when no more foraging actions can be added to the sequence due to resource depletion, a move action must be launched to a new area plenty of resources.

The day when a movement action is launched, the agent receives resources from a secondary forage action that is bonded to the movement action. The reward is half of a normal forage action that would be launched in a day where only forage happens. It is critical to choose the step to launch the movement and minimize the effect of receiving only half of the reward. It is the most demanding feature for the planner apart of moving the agent towards richer areas as a long term effect of the drift of the movement. The artificial ideal planner would be imagined free from bad allocated move actions. The artificial ideal planner is imagined as proposing always movement actions in the time step when they would not increase the starvation of the agent. So that is what happens when you do not apply the depletion of resources to the environment after the execution of a forage action. Because the resources stay unaltered as if the patches were new and the agent just would have arrived in the last time-step. And then it seems that the sequence of actions does not disturb the flow of resources that the agent retrieves. It seems as if the planner would have been selecting each time the perfect action to be launched; either a normal foraging action or a move action in the time step before the system reaches a state low in resources, when the half forage does not cover the survival threshold. Why do we not fill the environment with infinite resources, and then have forage actions returning always plenty of resources? The infinite resources setting would not exhibit phenomena like the ideal planner in the original world model. There are seasons in the year, at the beginning and at the end, where biomass dynamics produce a low load of resources in the environment. Applying infinite resources masks this phenomena and also would produce a too low lower bound. Doing no-depletion consists in reading the amount of resources, apply the foraging formula and give the result to the agent without updating the world. But this does not mask the effect of biomass dynamics. Because in the poor days of the year the formula of foraging returns low rewards. At the beginning of the year there will be low resource because biomass has not fully grow and the end of the year there will be less resources because of the effect of biomass decline procedure. The most perfect planner would find this situation, and could not give to the agent a recommendation that would allow recover more resources than the low amounts present in the poor days of the year. The ideal planner conceptualization must not avoid the increase of starvation in these cases. It can manage to reduce it to a minimum but can not keep it to level zero for ever and for any condition. This discards the option of using infinite resources as a medium to produce a lower bound to starvation rates, and strengthen the use of no-depletion.

Simulations show indeed that the yearly profile of the starvation rate for the gap agent follows quite accurately the profile of the no-depletion agent. Most of the simulations show the pattern of the figure when overlapping the Starvation measure for mdp, no-depletion and gap agents corresponding to the same simulation seed (Fig. 4.7).

For the different rainfall conditions we kept finding the same result. Box-plots show that the modifications used in the gap agent approach the performance to the lower bound (Fig. 4.8, Fig. 4.9,Fig. 4.10).

In conclusion, there is little space to consider the possibility that more effort applied to biomass prediction would reduce the starvation of the agent. The area enclosed between both starvation profiles, the normal agent's and the no-depletion agent's is something we must assume due to the design and properties we chose to design the model. It is up to future steps to push in the problem of divergence through introducing multi-agency and neighbour awareness.

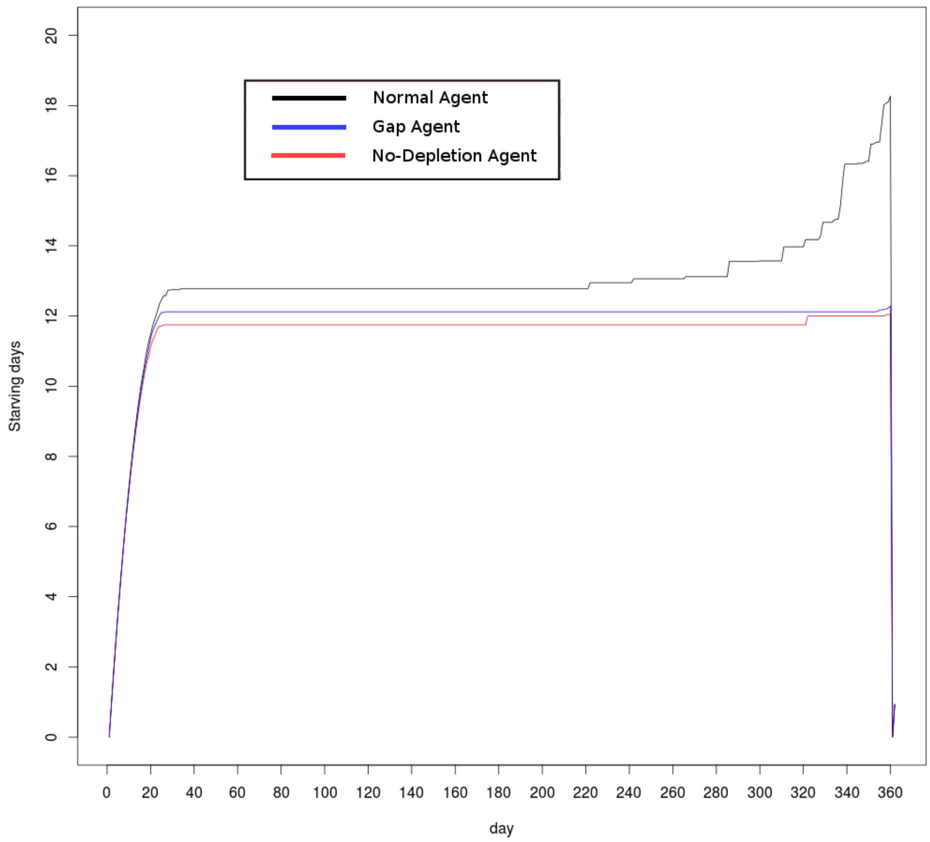

Figure 4.7: Overlapped starvation profiles of the agents in the no-depletion experiment with a rainfall of 500 rain units .

4.3 Comparing the Rule Based Agent and the MDP Agent

4.3.1 Experiment 1: Mean Starvation Day Comparison

The first experiment compares the three types, random, rule-based and MDP. The objective is to compare the performance of different decision mechanisms in movement and foraging activities without interaction of other agents, without the variability in rainfall, only the end product of survival at the end of a short period of time. The performance that we measure is directly linked to the critical indicator of survival mechanism, the measure of mean Starvation days at the end of the year. This measure is affected by the quality of decision making that the agent makes during the year to get resources to survive.

The experiment consists of exploring three amounts of yearly rainfall for a monsoon season at the beginning of the year. The first scenario tries not to make

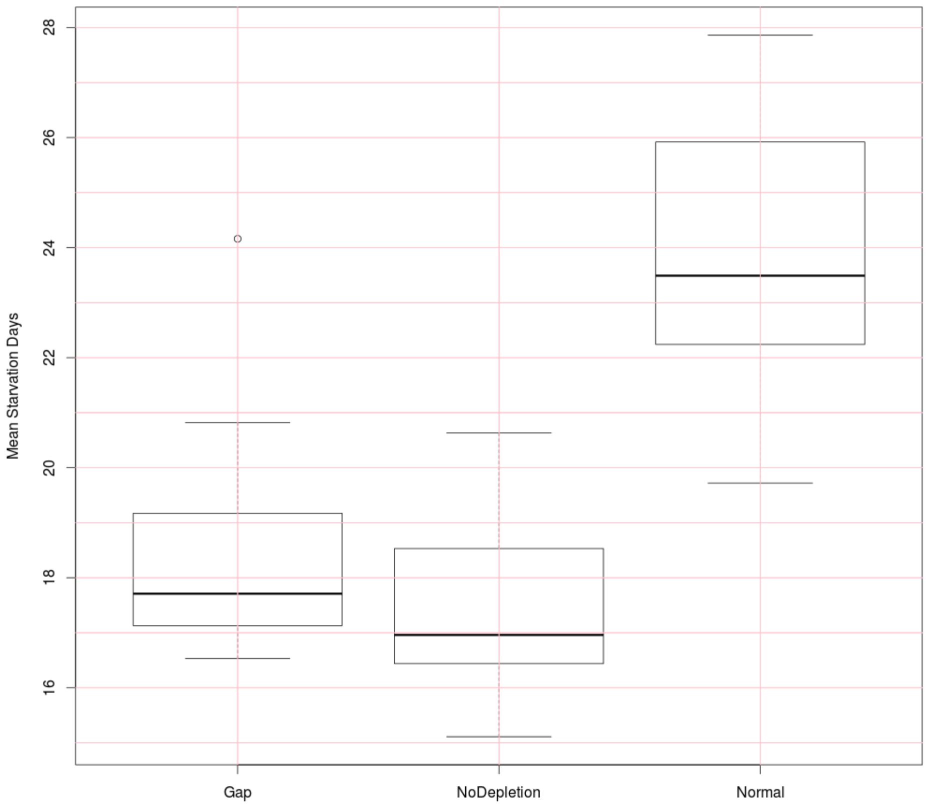

Figure 4.8: Mean Starvation days at end of the year box-plots for "No-Depletion" experiment with 500 rain units at the monsoon season.

things easy, 500 units of rain. The second value is mild and uses a value of 1500 units of rainfall. The third scenario replicates the environment that inspired the model, in north Gujarat, with a mean of 4000 units of rainfall, and standard deviation of 2000 units. The standard deviation is not used in this two year simulation but it is applied in the ten years simulation of experiment two. As it is mentioned, the simulation lasts two years. We hope to giving the agent time to explore and discern a patch during the first year with a minimum quality to see how they develop during the second year of the three engines that decisions are being compared. All runs start with the agent located in the upper left of the map, with two adults and four children. MDP agents are configured with the following combinations of parameters, horizon 10 plus width 1000, horizon 3 plus width 500, horizon 6 plus width 1000 and horizon 6 plus width 500. We simulate runs with one single agent because do not want the noise introduced by other neighbours in the analysis of the agent getting on in the foraging and migration activities.

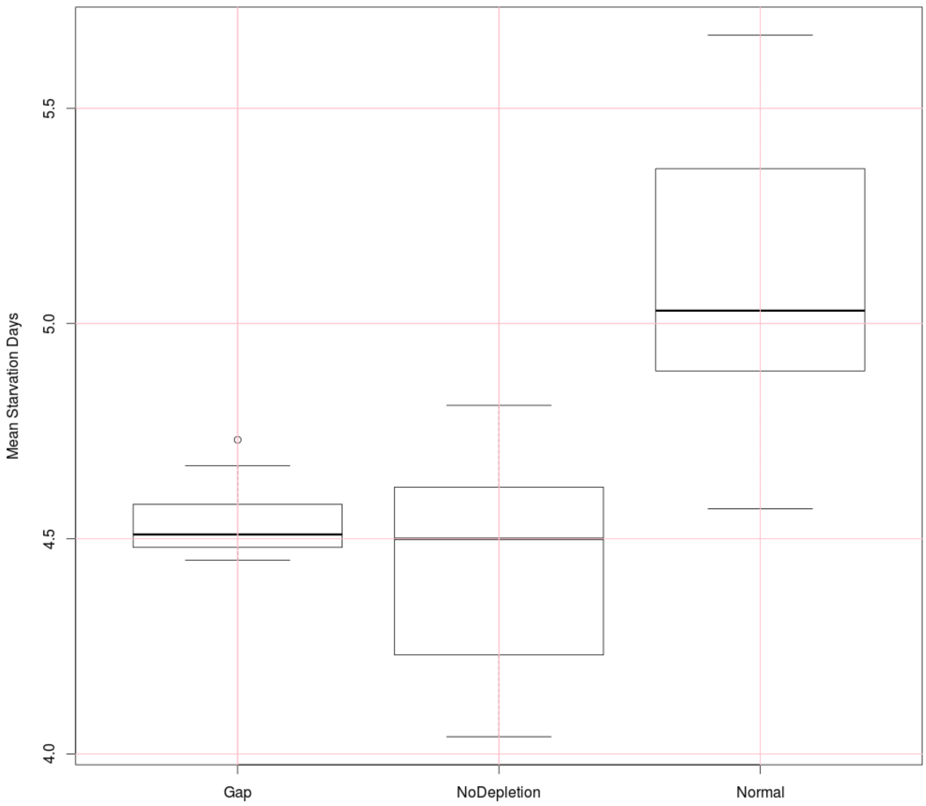

Figure 4.9: Mean Starvation days at end of the year box-plots for "No-Depletion" experiment with 1500 rain units at the monsoon season.

The data from the experiment is distributed producing a figure for each different rainfall. Each climatic condition contains a set of juxtaposed box-plots. Each box-plot is the statistical summary of starvation rate for one configuration of agent engine. The adjacent placement allows us to compare the deviation of the distributions and their averages.

We can clearly see that the rule based agent obtains the worse score in starvation rate (Fig. 4.11,Fig. 4.12,Fig. 4.13). All the mdp configurations get lower values respect the rule based and random agent.

MDPs exhibit a better performance than other agents. In the long term they will die from two to four times less than the classical approach of reactive rules. Also, we have been surprised by the better performance of the Random agent versus the rule based. We can offer an explanation to this based on movement rate and penalty due to movement. Agents produce chains of movement actions and foraging actions. One foraging action occupies the working time of a day. One

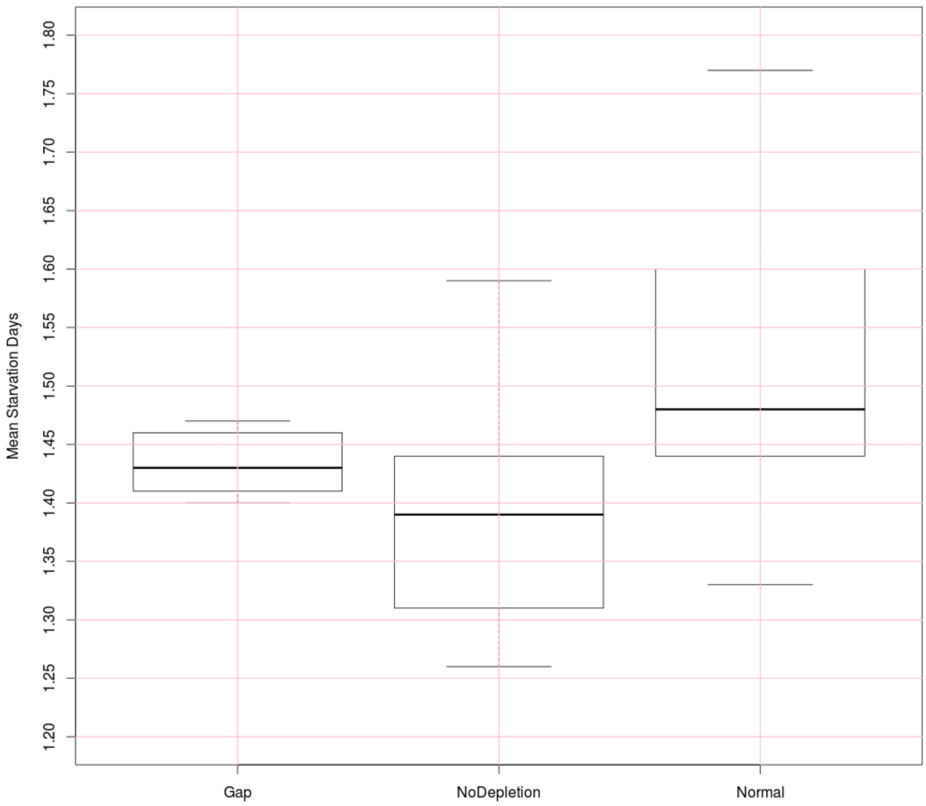

Figure 4.10: Mean Starvation days at end of the year box-plots for “No-Depletion” experiment with 4000 rain units at the monsoon season.

movement action produces movement and some resource retrieval that could be understood as foraging along the trip path. We have represented it as a half reward from a secondary foraging action bonded to the movement action that take place in the same day.

It is critical to know the correct time-lapse when to move the settlement: The day you move you will only retrieve half of the resources you could take in a normal day of only foraging. As days pass you deplete your home range till the time step that you cannot make a living from the resources around home. For the rule based agent that is the condition to move. If you move when resources are under your survival threshold the “half reward” associated to a movement day will be much more below your survival threshold; and it is a severe penalty. If you tune the decision rule to avoid an extreme depletion of the surrounding environment you will have an agent that moves too much which leads to half- reward penalty accumulation.

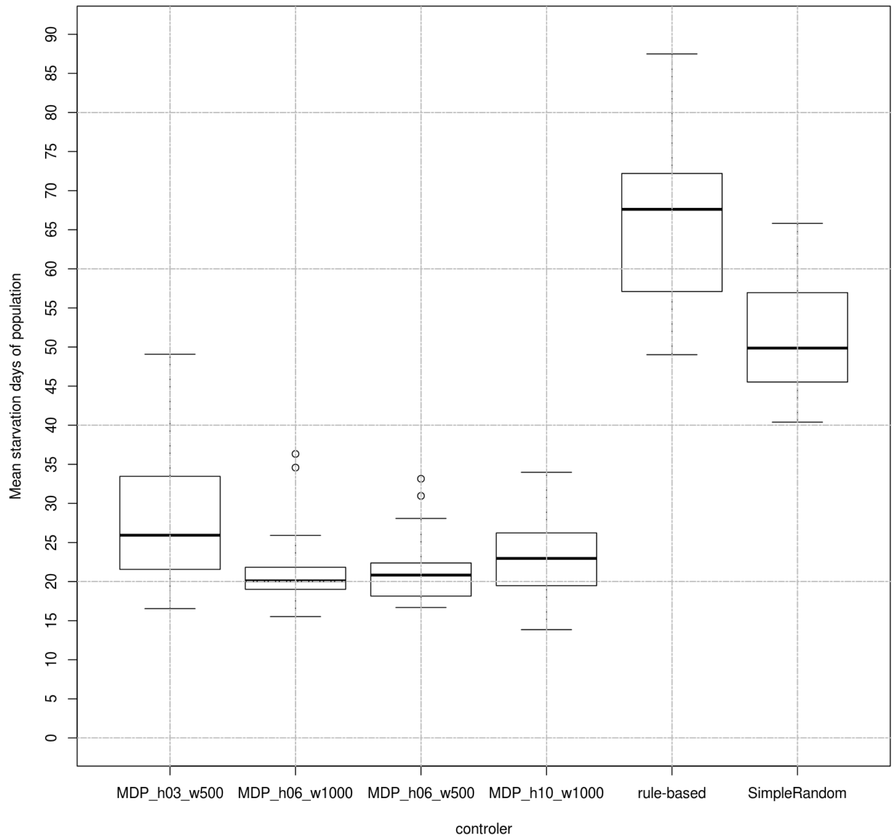

Figure 4.11: Mean Starvation days for a rainfall of 500 units

The figure (Fig. 4.14) shows a typical profile of mean starvation along a year for three kind of agents, a rule based one, an mdp agent, an the one called hypothesis. Starvation increases in the firsts days of the year from zero to some value that will be hold till the critical part of the year. The resource dynamics depends on the first season of monsoon. At the beginning it rains but there are no resources. It explains why starvations increases in the first days. And the last season of the year is a dry one where resource are scarce or practically zero. Taking into account only the rule agent and the mdp we can see there is no way to get a zero starvation score. There are no resources whether you move one place or another, there is no smart path selection that will give more resources when there is practically nothing. Any path of movement leads to low resource patches. That is why for both agents there is an increase on their starvation ratio in the first days of the year. But, why is it higher for the rule based? Because the rule based agent moves when it finds resources are under the threshold condition on the movement rule. The agent will move and will receive the penalty of the half-forage. By the other hand, the mdp agent

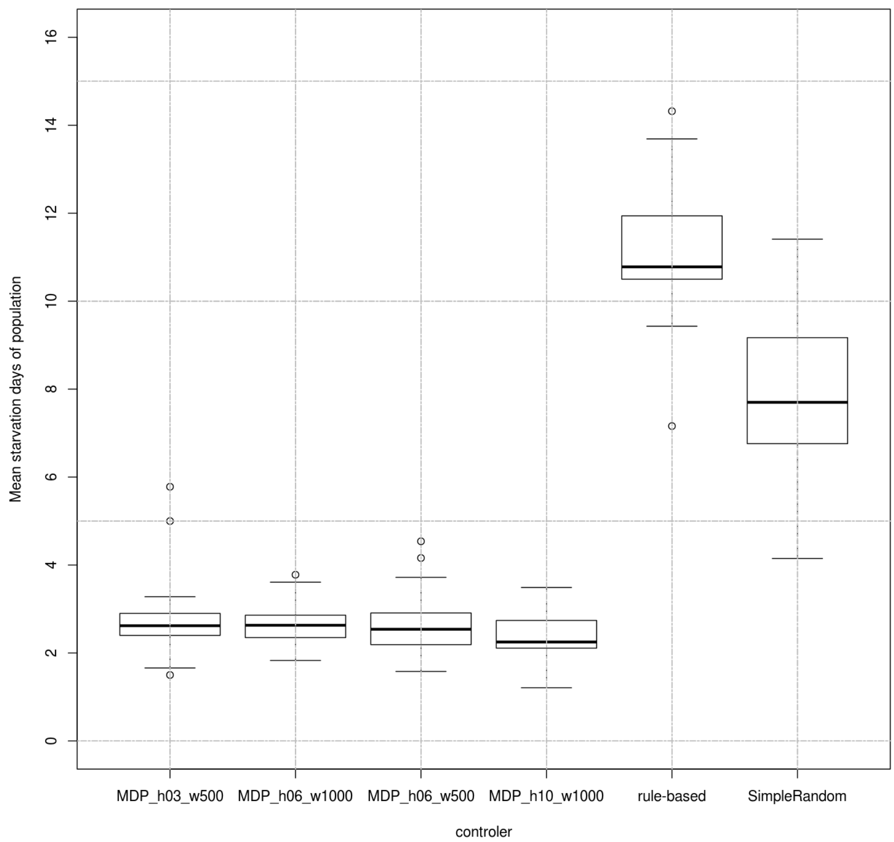

Figure 4.12: Mean Starvation days for a rainfall of 1500 units

can explore plans and detect the ones that will make arise the penalty associated to movement actions it is not advisable to move. The mdp agent waits. For the central part of the year there are enough resources and both type of agents can manage well. But again, when there are low resources around the rule based agent will not behave optimally. In a first approach one could think that you cannot scape from starvation, there are low resources like at the beginning of the year. On the contrary, the presence of plateaus tagged in the profile of the starvation indicate that for some steps the agent does not see its starvation rate increased, so there are enough resources to survive for some days. The plateaus are correlated with the movement actions. And each time we have seen a change from one plateau to another one there have been movement. These increments come from the penalty associated to half rewards the days you move. The rule based agent waits till it is too late and there so low resources that moving implies to receive a half-reward under the survival threshold of resources. The mdp agent can manage to choose the time-step to move minimizing the penalty and moving before it is too late while

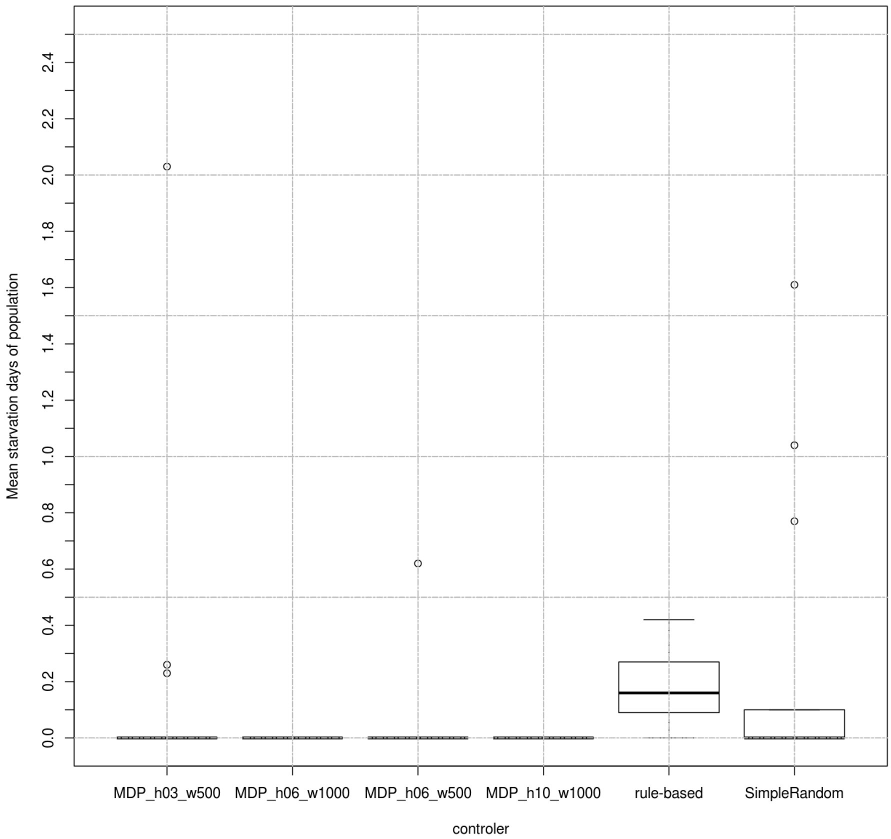

Figure 4.13: Mean Starvation days for a rainfall of 4000 units

there are enough resources and a half-forage will not be as harmful as if you have waited too much. We checked this hypothesis extending the rule based agent with two new rules. The rules are inspired in the mistakes shown by the rule based agent. The first extra rule says "At the beginning of the year do not move till time-step 7". The second rule says "If last move action was launched more then 5 steps ago, move now". The extended rule agent corresponds to the dotted profile in the figure(Fig. 4.14) with the label "hypothesis". It is not our aim to enhance the rule based agent adding more and more rules, because many times any set of rules will have a fail point that can be triggered by the changing conditions of the environment. But the important thing is that through this comparison, apart from checking one agent type versus the other, we obtained a deeper insight of the system, of the mechanics and conditions for correct resource depletion. And this is a by-product of putting side by side different decision making engines.

Besides, we also observed that configurations with horizon 6 in mdp agents performed like horizon 10 configurations. Our hypothesis is that after doing some

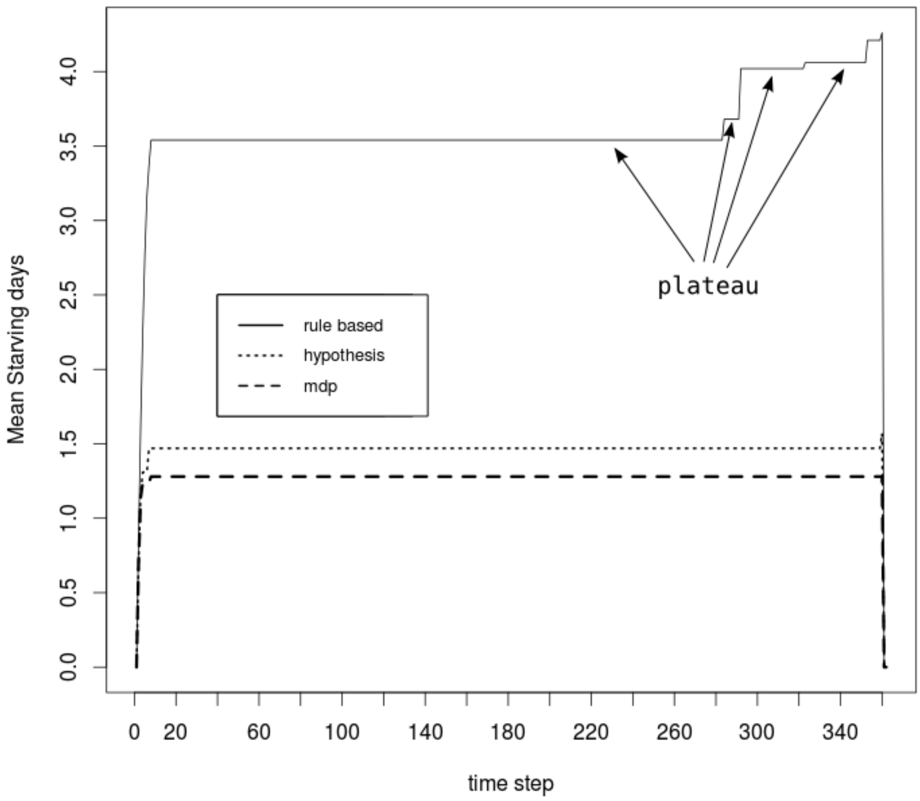

Figure 4.14: Starvation rate profile along a year for three types of agent. The higher the value, the worse.

statistics the size of time lapse between consecutive move actions is approximately five or six steps in mean, or even less for the last days of the year, when resources are more scarce. It seems that the optimal time to exploit resources in a patch is no more than four or five days, then, if you exceed this bound you must leave launching a move action (as previously mentioned). The decision process must foresee the best time step to move paying the minimum penalty. And if the horizon covers such time lapse between move actions it will cover the feasible days inside the set to choose the day with optimal balance of minimum penalty and maximal normal forage reward.

To assess this explanation we produced some density plots (Fig. 4.15 for a rainfall of 4000 units), (Fig. 4.16 for a rainfall of 1500 units), (Fig. 4.17 for a rainfall of 500 units). For each run from the experiment, the amount of time steps(time lapse length) between move actions is registered . But we only consider time steps from the critical part of the year, the last forty steps. Each run will have a mean of lengths associated. For each configuration of the mdp agent we will have a set of means of lengths. This set is a sample of the distribution that will characterize a configuration. Each run is a two year simulation from the experiment 1 we are describing in this section. We took each set of means and produced a density plot for the distribution of mean lengths for each different horizon, three, six and ten.The more two density plots match the more we could guess that the distribution of sequences of move actions match and that the two compared configurations produce plans that are not very different one from another.

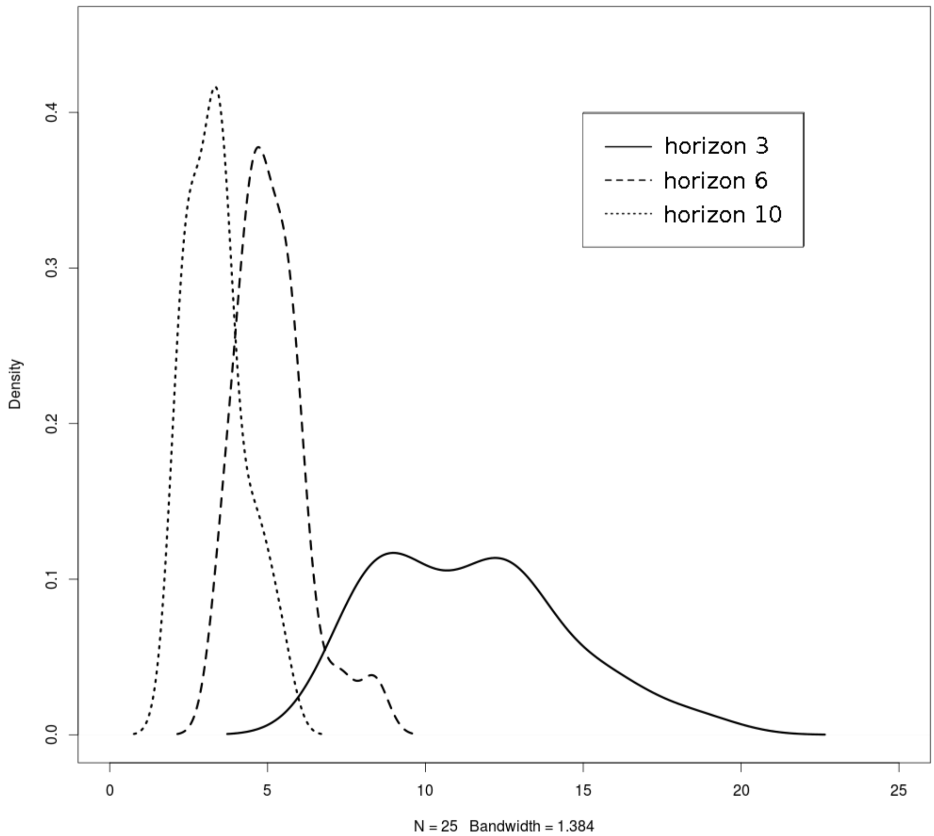

Figure 4.15: Density distribution for time lapse length between move actions for a rainfall of 4000 units.

The plots strengthen the evidence that configurations horizon 6 and horizon 10 besides having similar starvation and hence, survival performance, they have plausible matching distributions of lengths.

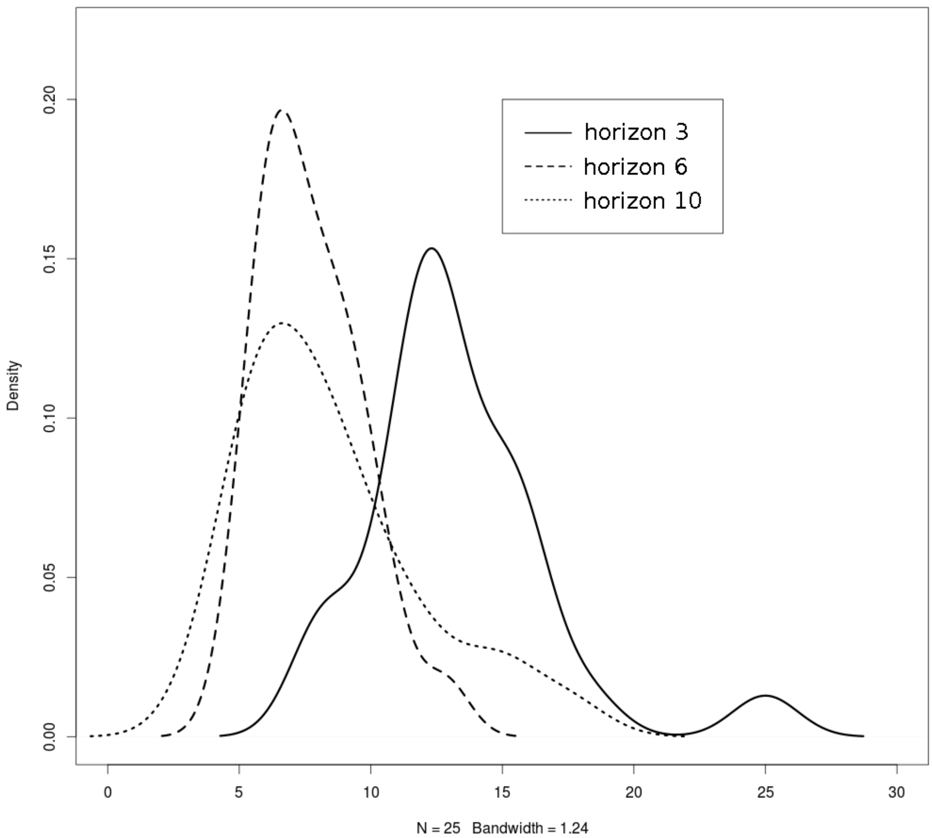

Figure 4.16: Density distribution for time lapse length between move actions for a rainfall of 1500 units.

4.3.2 Experiment 2: Ten year simulation of Rule Based Agent vs MDP Agent

The second experiment extends the simulation from two year runs to ten year runs. A longer run will make explicit the response to accumulated changes through the years, mainly, the dynamics of individuals that conform the agent, as new children are born or some die due to starvation or the demographic mortality basic rules. We replicate the settings and configurations used in the first experiment, except that the random agent is not taken into account; we focus on comparing the drift of the classical rule agent with the mdp agent.

The ten year trace (Fig. 4.18 for a rainfall of 500 units, Fig. 4.19 for a rainfall of 1500 units, Fig. 4.20 for a rainfall of 4000 units), contains a set of cells with a number indicating the year on top of them. For each year there will be a set of box-plots of the mean of starvation rates. The Y axis is the measure for starvation and

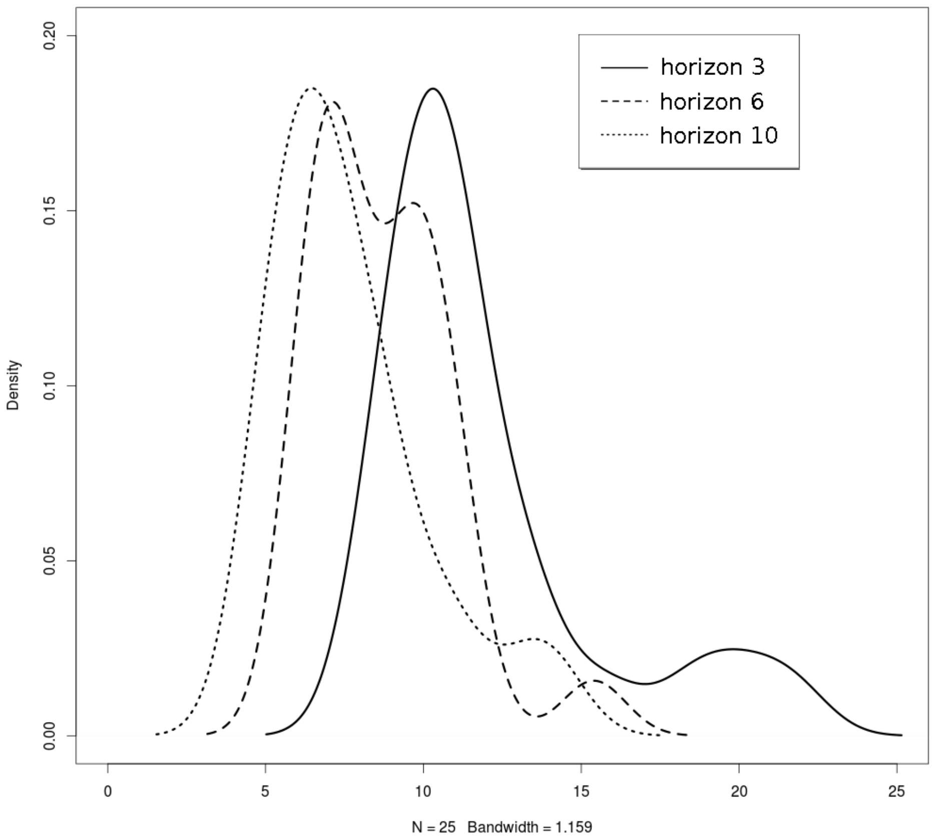

Figure 4.17: Density distribution for time lapse length between move actions for a rainfall of 500 units.

the X axis contains the spread of configurations evaluated. The box-plots follow the order in the legend. The top most item of the legend, configuration "h3_w100" (horizon 3, width 100) corresponds to the most left box-plot in a year cell. As we go down in the legend we find the configurations that appear in a sorted manner from left to right.

Box-plots for configurations from mdp agents appear clustered in two sets. The horizon 3 set and the horizon 6 and 10 set. As we expected the rule based agent falls apart and exhibits a very different trend from the mdp agents. For the scenario with rainfall set to 500 the mdp agents thrive towards to an increase of population that maybe will settle in the carrying capacity. But, the rule agent follows a decrease that by the last year gives us a first case that does not see its agents alive, the black dot under the rule based's box-plot at level zero. We could conclude that with enough time the rule based group will collapse although the decrease rate is asymptotic to forty. For the plot with rainfall 1500, things get easier and probably the rule based agent group would not collapse and remain stable around a population of one hundred fifteen. The third scenario is very friendly for agents, there are enough resources for groups of one hundred agents and all the types we are comparing evolve to greater sizes in their populations. The rule agents can not enjoy a same growing rate as the mdp ones. Let's see that as we increase rainfall the deviation in the box-plots of mdp agents decreases. There are more resources and it masks the depletion of the other agent for some agent that most calculate its plan. This reduces uncertainty and stochasticity in the rewards and starvation rates.

The differences between the classical approach and the AI approach arise more explicitly in experiment number two. Indeed, one of the scenarios could allow to predict some collapsing dynamic or low recuperation rate in one type of agents while the other agents exhibit some proliferation. Configurations for mdp agents, when compared, gives us quantitative differences. We discover trends in the same direction with more or less intensity. And maybe it is sure we can find many systems which are sensitive to the mdp configuration parameters where we would find runs producing different trends and conclusions for every configuration with a soft slope from the range of classical agents performance to sophisticated agents performance. But for the system we have been studying, the scale of discovered differences between rule based agents and mdp agents points out that maybe for some systems equivalent to ours, when we introduce the rule based agent or the mdp agent, the drift and the conclusions will be qualitatively different. And it marks a border between classical rule agents and the mdp agents.

4.3.3 Experiment 3: Width Exploration

The third experiment explores configurations for mdp and its performance for optimal foraging and migratory actions. MDP configuration parameters denote the depth of the reasoning in the decision process. Horizon sets the amount of steps you look in the future; and width sets the amount of hypothetical traces explored to get an statistical sample of eventual outcomes and distinguish between good actions from bad to be applied in the next time step. This exploration is needed to detect sensitivities and interactions between horizon, width and the starvation rates for the sake of the tests in the evaluation of the mdp approach versus the rule based model. To test horizons with values '1' and '2' is out of discussion because you cannot take all the advantage of the planning engine. And horizons above '6' we have seen are of great computational time expenditure. Even horizon '6' demands big amounts of hours making simulations non reasonable when applied to ranges in the scale of decades and hundreds of years.

The configurations are explored under the same conditions as the experiment one. The chosen combinations follow, h10 and w500, h10 and w1000, h10 and w5000, h10 and w10000, h6 and w200, h6 and w500, h6 and w1000, h3 and w50, h3 and w100, h3 and w500.

We represent in the plots (Fig. 4.21, Fig. 4.22, Fig. 4.23) the same measure and distribution of data in the way of experiment one.

A first approach based on the shallow evidence restates the observations from the first experiment. Horizon 6 and 10 go coupled and horizon 3 produces a lower performance of the survival skills. As rainfall increases the variability in the distribution of starvation rate decreases as decreases uncertainty and low resource states in the simulation. It seems roughly that there is no difference when the agent uses one width or another for some given horizon. But if we look back to experiment two, they appear; it is caused by the presence of other agents who introduce uncertainty. When the vicinity displays greater variability in resource availability that is due to the other's activities it results in having a greater range of possible states to evaluate. To explore a bigger set and filter worthy trajectories from bad ones it requires to increase the width to ensure a greater statistical significance when selecting and discarding trajectories. And the plots from experiment two show that agents with greater width benefit from it.

The second point, from the technical point of view, is that if we achieve same statistical results for the output variables in the system (e.g. starvation rate) with a same width with different horizons, we could extract the same conclusions. Then the model would be statistically equivalent under the simulated scenario and hence we could choose the configurations that imply less computational power to run cheaper and longer simulations. Summarizing, for environments with no indirect competition, the widths we have applied for each horizon do not differ very much for the survival capability.

4.3.4 Conclusions

Experiments comparing the different agent types confirm our hypothesis. Introducing AI in an agent can produce meaningful changes in the results of a simulation. When applied to a resilience test scenario choosing one or another can produce different conclusions. Whe have seen qualitative differences in the results. MDP agents can achieve better levels of resilience. The key factor is adaptability. The greedy schema of the rule based agent makes it blind to the critical points of the year where two opposite strategies must be chosen and applied. The beginning of the year demands a low movement rate while the end of the year demands a higher movement rate with optimal decision for the time step when to move. One of the experiments confirms this pattern through an extra set of rules introduced in the rule based agent. And we could keep on extending the agent with more and more rules. The objective is not to design an expert system in survival in Gujarat. We wanted to find the differences of classical social agent's static behaviour and adaptability versus dynamic behaviour.

Also, results from experiment 3 in comparison to experiment 2 make explicit how important it is the effect of indirect competition. Changing UCT parameters horizon and width affect the performance. A future stage, just like we do for biomass prediction, should consider to introduce neighbours' effect. An evaluation will be needed in order to find the best technique to integrate this component to the knowledge layer of the agent and the UCT as well as find a lower bound for the new starvation. This way we will be able to asses the gain of introducing it and the computational cost in exchange.

As the concluding reflection, just to mention AI was born to emulate human performance in intelligent tasks. But it does not mean AI will give all your work done. Internal procedures in simulation are not like pure problem solving nor pure support to decision making. Many activities where AI is involved allow to AI techniques to express the power without restrictions, "perform as best as you can" : achieve the best accuracy on Part-Of-Speech tagging, cluster data with minimum number of false positives, produce an optimal policy for an autonomous robot, for instance. AI in social science simulation must be handled with good judgement. AI algorithms cannot be used at their fully without taking into account the cost that will represent after thousands of simulation steps. But it is more important the modelization task. AI algorithms are subjugated to the bounds(sometimes clear, sometimes fuzzy) or the features of the behaviour chosen for the agents.

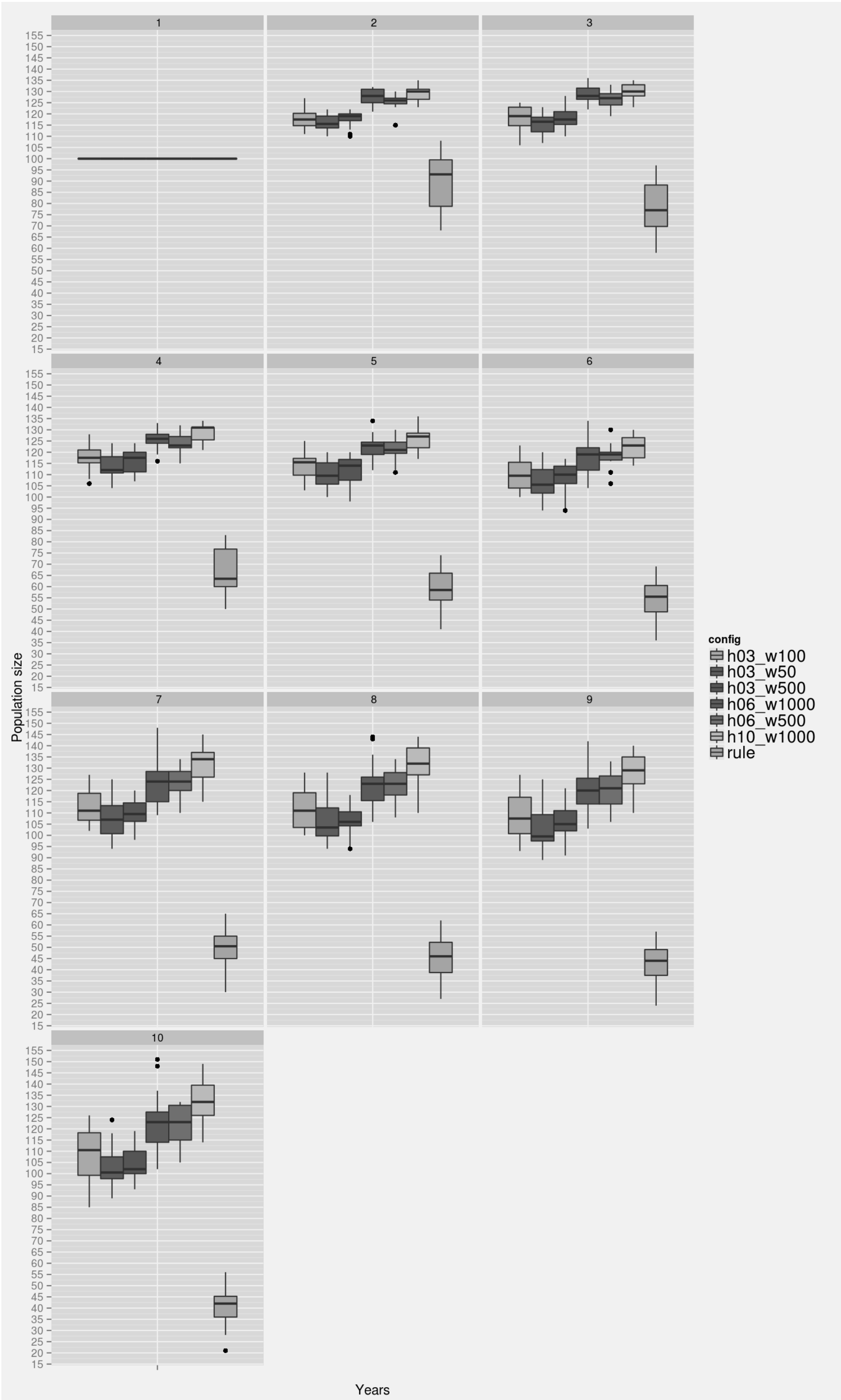

Figure 4.18: Trace of starvation rate along a ten year simulation with one hundred agents. The resources are produced by a rainfall of 500 units.

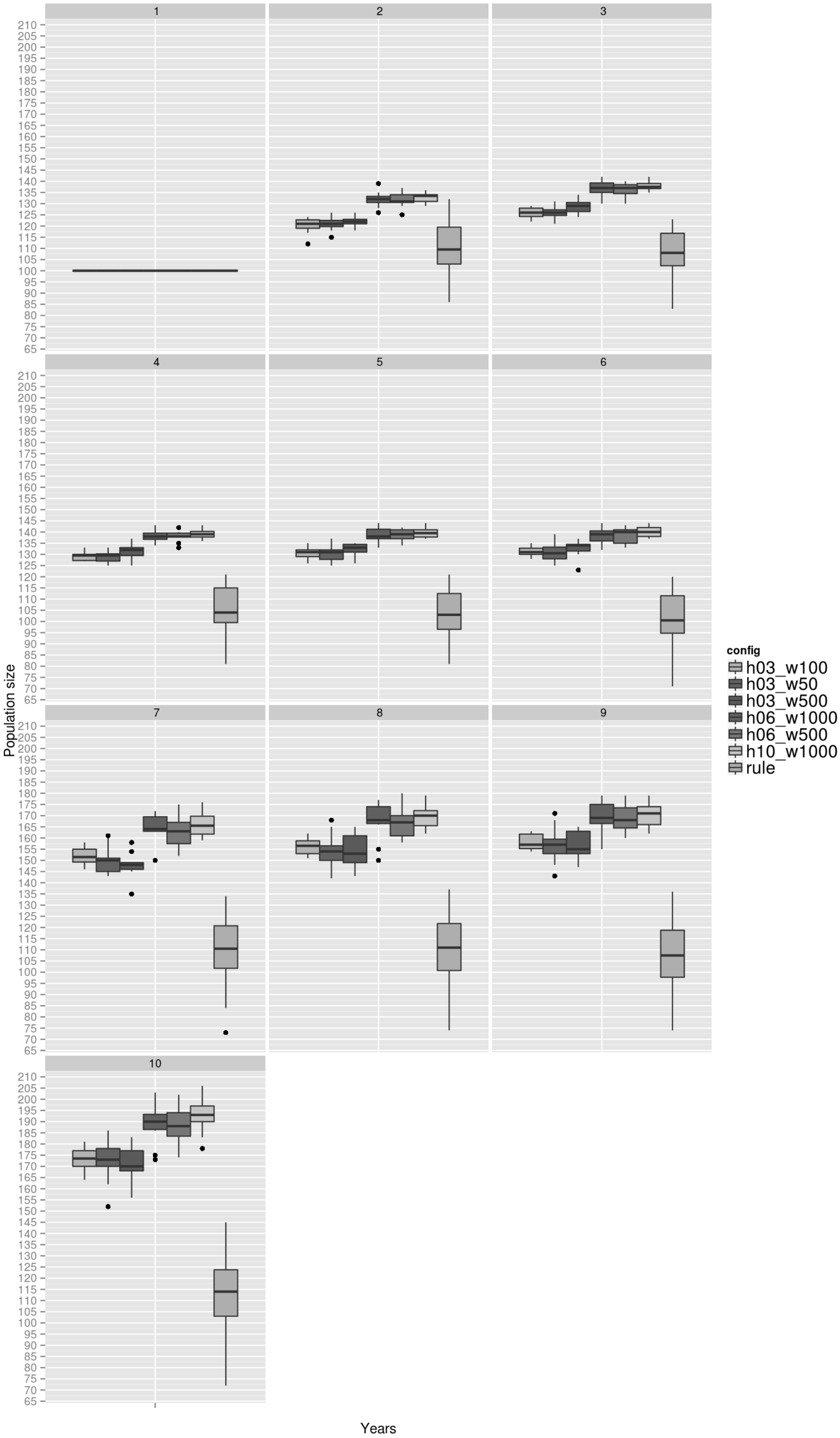

Figure 4.19: Trace of starvation rate along a ten year simulation with one hundred agents. The resources are produced by a rainfall of 1500 units.

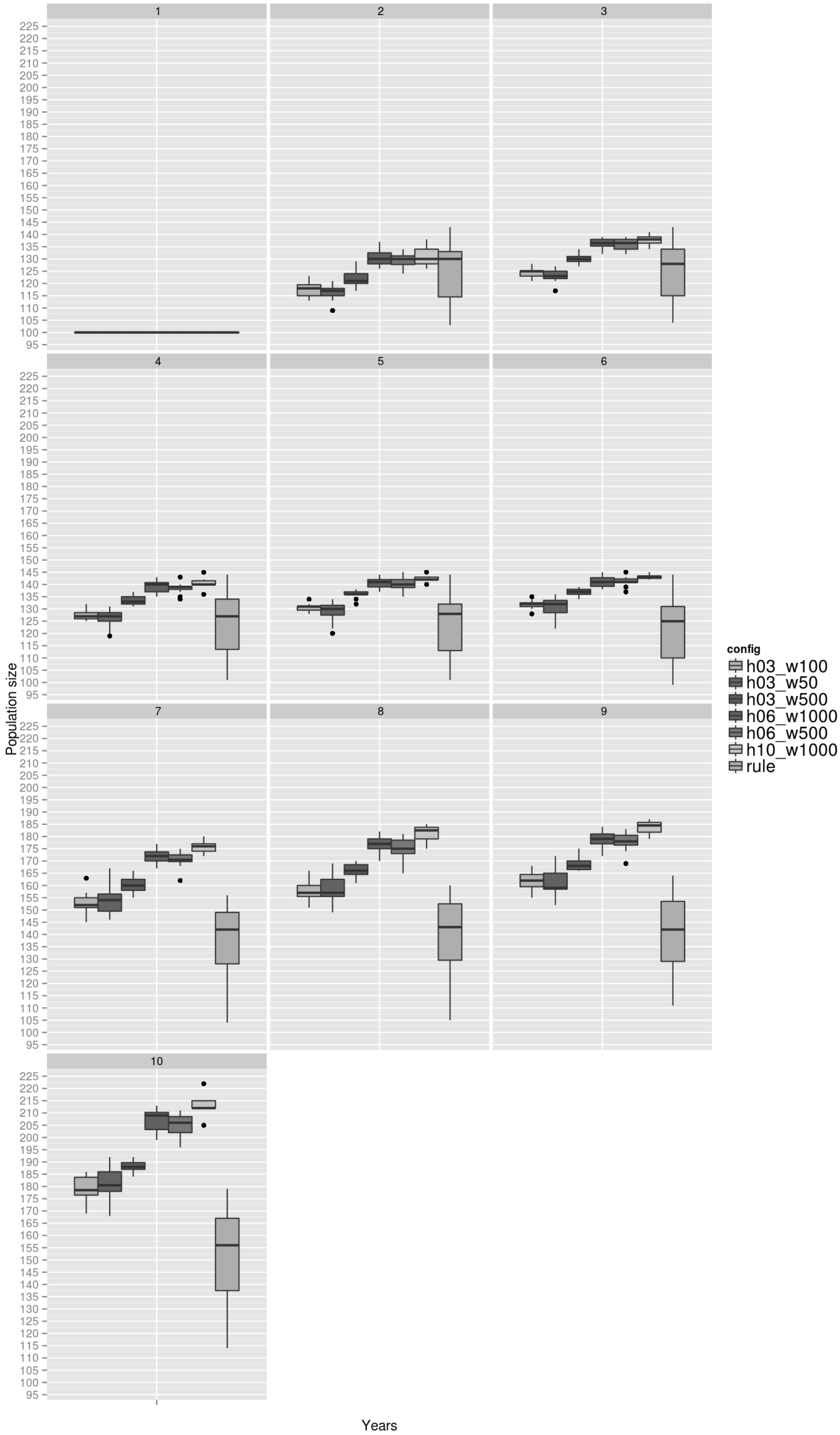

Figure 4.20: Trace of starvation rate along a ten year simulation with one hundred agents. The resources are produced by a rainfall of 4000 units.

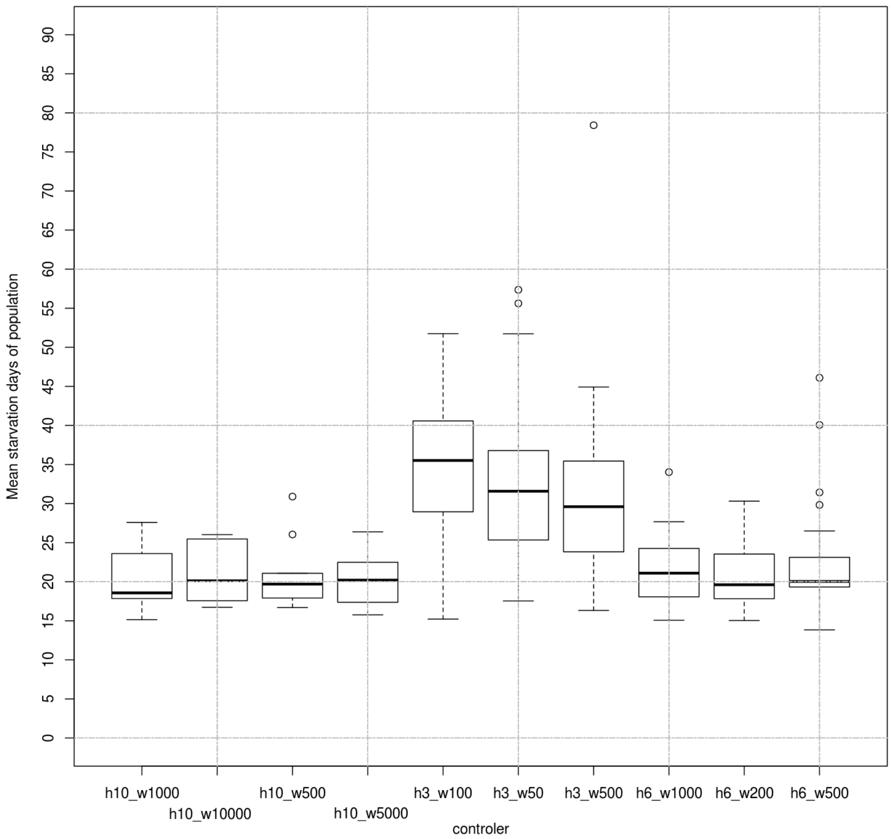

Figure 4.21: Exploration of widths with a rainfall of 500 units.

0

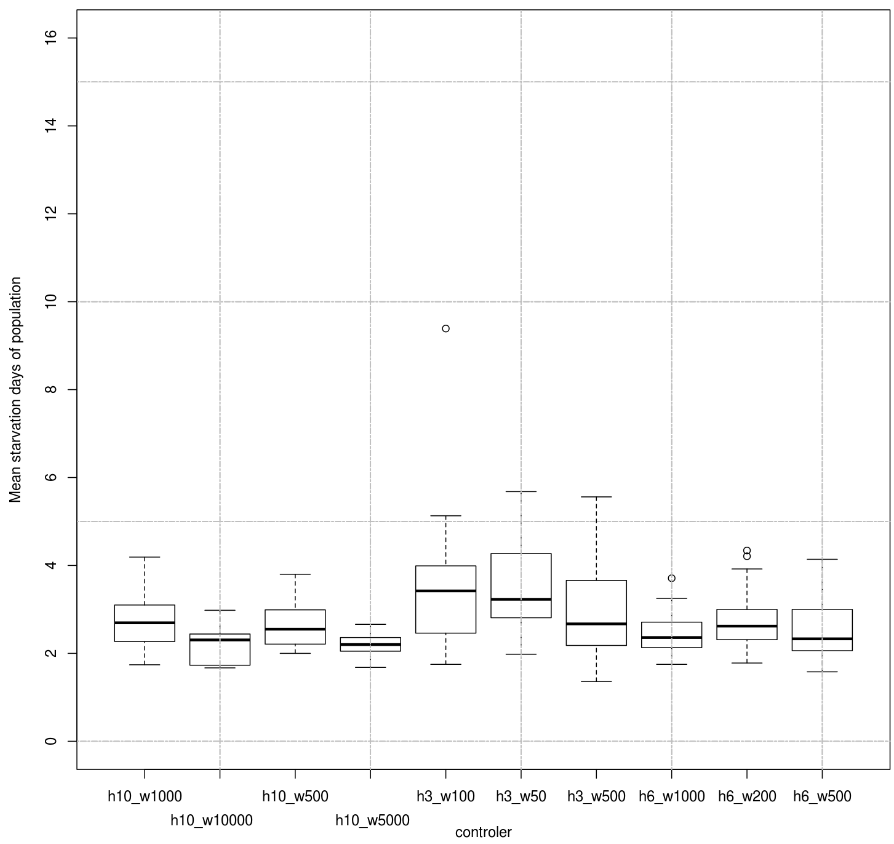

Figure 4.22: Exploration of widths with a rainfall of 1500 units.

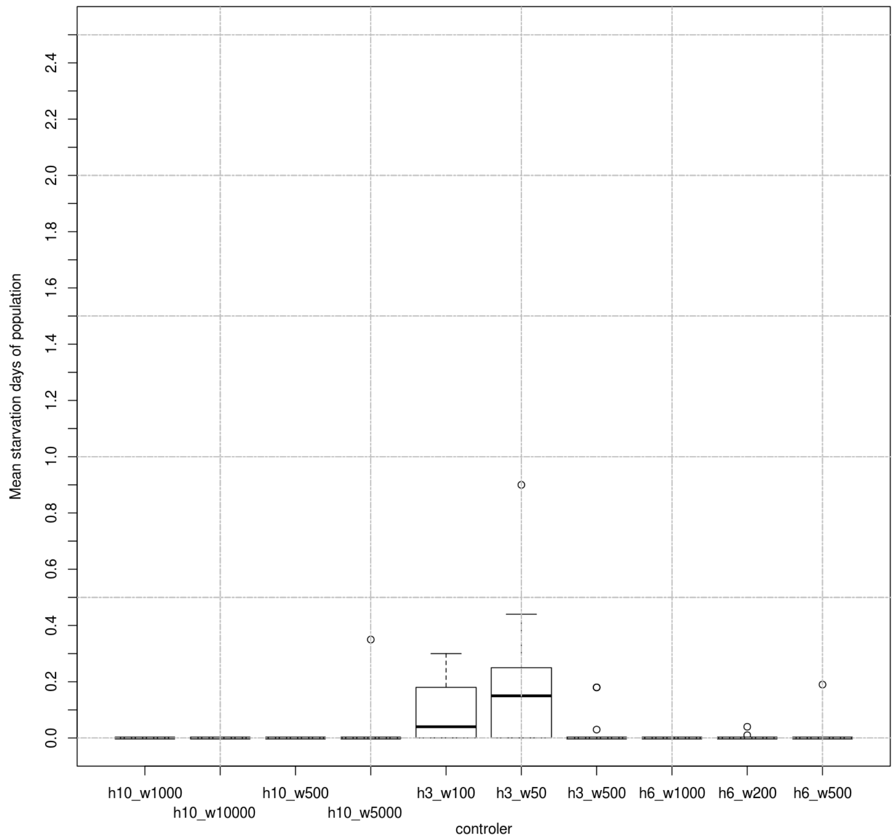

Figure 4.23: Exploration of widths with a rainfall of 4000 units.

Chapter 5

Conclusions, Future and Recommendations

This master thesis reflects part of the experience of my contribution to Gujarat Project in its first stages. This part of the project had as objectives to find an stable model to begin study resilience of HG and AP in semi-arid environments, and the possibilities offered by the introduction of sophisticated decision making engine in the agents populating the simulation.

The main hypothesis of HG resilience, in comparison to other cases where they disappeared at greater pace, is about climate variations and hard environmental conditions. The model has grown from a toy model based on Sugarscape[3]. The initial prerequisite was to tune the model towards a stable set of conditions. Experts selected the set of conditions for weather parameters and associated biomass from anthropological and climatic registers that would allow a basic degree of stability. This stability floor must be altered applying variations in rain and other future factors to be introduced. The variability and extreme values induce stress in the environment and will make more explicit the factors where we can asses the resilience of HG in comparison AP. We can say we have achieved the stability basic configuration of the environment. Agents survive, reach a carrying capacity and keep it along the generations. The system has plausible environmental conditions and behaviour.

A very important component in HG social organization is environmental information sharing. This feature is introduced to add another layer to the mechanisms of resilience of HGs. Anthropologists in their field work have observed it is an important and necessary activity in the HG daily activities. Information Sharing has been tested in three different scenarios and has shown positive results that confirm its a necessity in the model. Summarizing, bounded knowledge plus information sharing approach the agents to omniscient mind. This simple statement entails a question concerning the always present issue of simple models, throughput and efficiency. Although this is not place to ask it, and surely the project will end up providing evidences to stick to information sharing, one would dare to say that the model could be simplified removing information sharing and following Ockham's razor running a lighter model applying omniscient knowledge in agents.

The achievement in efficiency finally allows the project to seek results in a broader perspective. The simulation of a millennia in less than two weeks of CPU will allow to detect and follow hidden drifts of patterns of slow movement. If the simulation is not long enough you do not give time to the system to develop many effects that in real world span centuries and millennia.

The conclusions about introducing AI and more sophisticated techniques in social science simulation receive the confirmation for its use from two points. The empirical evidence show that sophisticated agents outperform classical agents in their evolution and response to challenges. The second is that modelling requirements of this project force us to use AI tools to represent agents behaviour. There are many details and relationships. The environment is very changing due to the mutual interaction of agents. And simple classical models cannot encompass a behaviour that will show successful, reasonable and or realistic decisions under this environment.

5.0.5 Future Work

1. Introduce multi-agency.
2. Extrude UCT for Learning and Pattern Discovery
3. Learn social, foraging and migration rules from the stochastic explorations of UCT.

The current state of the Gujarat project has the principal building blocks to follow its scheduling. The project will keep advancing through different parallel front lines in charge of subgroups of members of Gujarat case study.

1. AP Agents have their development in a very advanced phase. The decision making for APs will introduce planning at different scales. HGs launch actions that are resolved in one day. Launch, effect, reward happen in the same time step. But for APs the action of preparing a field for crop produces the reward two seasons later. As APs also have one-day-atomic actions the planner will manage short-term and long-term decisions.
2. The efficiency solution based on delayed cell update will be finished. This enhancement will lower the running times to a point which would mean that less then one week of CPU will be needed to simulate one thousand years.
3. Information sharing and Partial Information add-on will also see added a "scouting" action to the agents for resource discovery.
4. The model for dynamic social group based on skill profiles will be complemented with research in charge of two members of the Gujarat project with long experience in cultural transmission.

1. Introduce multi-agency.
2. Extrude UCT
3. Computational Efficiency.
4. Learn social, foraging and migration rules from the stochastic explorations of UCT.

Chapter 6

Acknowledgement

Chapter 7

Annex 1 : Platforms

Social Science Models can be implemented with a great number of optional tools and libraries. For small models or prototypes there are case-tools and graphical environments that integrate a designer and an internal layer containing and interpreter. The model is executed by the interpreter it in some virtual machine. When the model demands high computational requirements it is implemented with a procedural language like C++ or Java. To that end many libraries have been developed to offer a framework of objects and procedures common to most of the simulations.

7.1 Pandora

Social Science Simulation group at Barcelona Supercomputing Center(BSC) is an important collaborator in Simulpast project. The group has developed a library for agent based simulation models and a tool for trace visualization. Simulpast project simulations and the library Pandora are highly interrelated. The use of Pandora and Mare Nostrum allowed to aim to the purposes and questions stated in the case study of Gujarat.

The aim was to produce an internal set of classes from scratch to get free of design decisions of other known packages used in agent simulation. The library Pandora has its strong point in parallelism. Pandora has been designed to get maximum profit from OpenMP threads and MPI layer.

Pandora executes its simulations applying data parallelization**??**. The space representation of the world is split in parts that will be in charge of a cpu-core. Supercomputers arrange their CPUs in blades. Some blades contain two, some other four. CPUs in a blade share the same memory. Pandora makes the most of this architecture. The blade memory has the representation of the world. Each CPU processes the needed calculi for decision making occurring in its slice of world. When an agent traverses from one slice to another the CPU exchange information through MPI. This communication events happen inside of the blade, the simulation is no spread along the rack of blades of the supercomputer. Parallelizing the world this way produces a possible problem of corruption of data. Agents decide their next action on the information taken from their vicinity. The action can alter the state of some piece of the world in this vicinity. The issue arises when a agent is near enough of the bound that separates its world slice from the adjacent one. An agent cannot interfere with agents in the same world slice. The assigned CPU is dedicated to the current agent, it is not possible data corruption. But if an agent can see and alter a location in another slice will collide with the agents of the adjacent world slice. The solution is to apply a next level of slicing. Each slice is again sliced in four. Each CPU visits the sub-slices in the same order. So all CPUs are working on the same sub-slice σ_i. World sub-slices with same id are not touching each other so data corruption is avoided with the constraint that the radius of the vicinity of an agent is less the half the side of a sub-slice, otherwise an agent in sub-slice 0 would reach cells in sub-slice 1 also reachable to an agent in the sub-slice 0 of the neighbour slice.

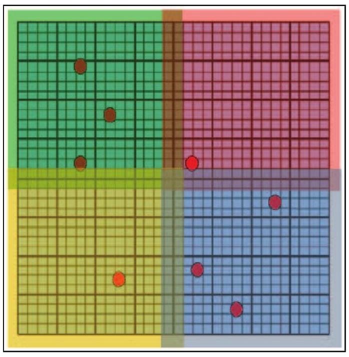

Figure 7.1: World paralellization.

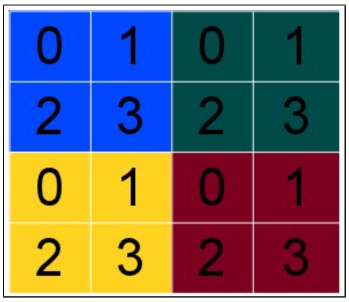

Figure 7.2: World sub-slicing.

7.2 Other platforms

RePast-HPC Repast-HPC is the most popular tool for tightly-coupled, large computing clusters and supercomputers [54]. It is based on the tool Repast for sequentially execute ABMs. Repast-HPC is intended for users with basic C++ expertise and access to high-performance computers. Repast HPC implements a dynamic discrete-event scheduler with conservative synchronization. It offers useful features such as data collection, specifying agent interactions by space and networks or automatic management of agent interactions across processes. Repast-HPC allocates a region of space to processors and manages the boundaries by copying them (and their agents) in the adjoin region.

NetLogo Netlogo[56] is an agent-based programming language and modelling environment for simulating complex phenomena.Netlogo uses a modified version of the Logo programming language. This is a different approach from other toolkits such as Swarm and Repast which make simulation facilities available to programs written in a general-purpose language such as Java. Despite being free, *Netlogo is not open source*. Netlogo is written in Java so can be run on all major platforms, requiring Java version 1.4 to run the current Netlogo version 2.0. Although the majority of users find Netlogo fast enough for most purposes, we do not find easy adapting it to an HPC environment and due to the heavy load calculi Gujarat project is involved we need platforms where it could be easy to tackle with low level details to apply performance tricks.

DMason DMASON is the Distributed Multi-agents simulation toolkit, based on Mason. As in Repast-HPC, users can create an ABM in MASON and then use the framework to easily distribute it over many machines. It provides a partitioning functionality that self-balances regions and requires and all-to-all communication. As opposite to Pandora and Repast-HPC, partitioning of the field is decided by the user in advance and, in our opinion, may cause an additional unbalance. Agents can migrate from one region to other and, therefore, it does not guarantee load balancing. However, it implements a simple balancing mechanism: when a region is overloaded it decides to split itself in smaller regions, dividing consequently the amount of agents in each of these regions. To deal with boundaries problems, neighbouring regions communicate before each simulation step. DMASON is developed in Java and uses Java Message Service (JMS) for communication between workers.

GridABM Using grid technology, the open source project GridABM [55] provides a set of templates to enable researchers to run ABMs in computing clusters and computational grids. It is based on Repast and depending on the topology of communication of agents it allow user to choose the appropriate schema and run their simulations in different parallel and distributed platforms. The programming is very similar to developing Repast sequential applications.

Bibliography

[1] A.L. Balbo, X. Rubio-Campillo, B. Rondelli, M. Ramírez, C. Lancelotti, A. Torrano, M. Salpeteur, N. Lipovetzky, V. Reyes-García, C. Montañola, M. Madella. *Agent-based simulation of Holocene Monsoon precipitation patterns and hunter-gatherer population dynamics in semi-arid environments.* Journal of Archaeological Method and Theory;vol21,issue 2.pag 426-446. Springer US 2014. 2014

[2] J.M. Epstein. *Agent-based computational models and generative social science. Generative Social Science: Studies in Agent-Based Computational Modelling*, pages 4-46 1999

[3] J.M. Epstein and R. Axtell. *Growing Artificial Societies*, 1996. 1996

[4] Joshua Epstein *Why Model?*. 2008.

[5] R. Axelrod. *Advancing the Art of Simulation in the Social Sciences. Japanese Journal for Management Information System, Special Issue on Agent-Based Modelling*, Vol. 12, No. 3 Dec. 2003.

[6] R. Axelrod. *Simulation in Social Sciences. Handbook of research on nature-inspired computing for economics and management*, 1:90, 2007

[7] Doran, J., *Prospects for Agent-Based modelling in Archaeology*. Archeologia e Calcolatori, 10, 33-44. 1999

[8] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach* 3rd edition. 2009

[9] Nigel Gilbert. *Agent-Based Models*. SAGE Publications, California. 2008

[10] Nigel Gilbert, Rosaria Conte. *Artificial Societies. The computer simulation of social life*. 1995.

[11] Charles S. Taber, Richard J. Timpone. *Computational modelling*. 1996.

[12] Gilbert, N., Troitzsch, K.G., *Simulation for the Social Scientist*. Open University Press, USA. 2008.

[13] John Scott *Rational Choice Theory*. From Understanding Contemporary Society: Theories of The Present, edited by G. Browning, A. Halcli, and F. Webster. (Sage Publications, 2000). 2000.

[14] Michael Wooldridge. *An Introduction to Multiagent Systems* 2002

[15] Michael Wooldridge, Nick Jennings. *Intelligent Agents: Theory and Practice* Knowledge Engineering Review Volume 10 No 2. (c) Cambridge University Press 1995

[16] Michael Wooldridge. *Agent-based software engineering*, IEE Proc. Software Engineering 144 (1) 1997.

[17] Yoav Shoham. *Agent-Oriented Programming. Stanford University: Computer Science Department.(Technical Report STAN-CS-90-1335).* 1990

[18] Lake, M W, *2000. Computer Simulation of Mesolithic Foraging, in: Gumerman, G. J., Kohler, T.A. (Eds.), Dynamics in Human and Primate Societies: Agent-Based Modelling of Social and Spatial Processes*, Oxford University Press, New York, pp. 107-143.

[19] Sugarscape. *http://ccl.northwestern.edu/netlogo/models/community/Sugarscape*

[20] Juan Pavón, Millán Arroyo, Samer Hassan, Candelaria Sansores. *Agent-based modelling and simulation for the analysis of social patterns*. Pattern Recognition Letters 29(8): 1039-1048 2008

[21] Mike Pidd. *Tools for Thinking: Modelling in Management Science, 2nd Ed. Wiley.Chichester,UK*. 2003.

[22] Kathy Kotiadis, Stewart Robinson *CONCEPTUAL MODELLING: KNOWLEDGE ACQUISITION AND MODEL ABSTRACTION*. 2008.

[23] Grimm V, Berger U, DeAngelis D L, Polhill J G, Giske J and Railsback S F *The ODD protocol: A review and first update*. Ecological Modelling 221 (23), 2760-2768. 2010.

[24] Patrick McGivern, University of Wollongong. Alexander Rueger, University of Alberta. *Hierarchies and levels of reality*. 2010.

[25] *The Unity of Science*, Stanford Encyclopedia of Philosophy (http://plato.stanford.edu/entries/scientific-unity/). 2007.

[26] L.S.Premo Equifinality and Explanation:The Role of Agent-Based Modeling in Postpositivist Archeology, published at Simulating Change; archeology into the twenty-first century Ed. Andre Costopoulos, Mark Lake. 2012.

[27] Peirce, C. S. Abduction (cf. Hypothesis [as a form of reasoning], Retroduction, Presumption [as a form of reasoning], À posteriori Reasoning). Commens Dictionary of Peirce's Terms.

[28] Pascal Perez and David Batten *Complex Science for a Complex World; Exploring Human Ecosystems with Agents*. August 2006.

[29] George Box Section heading, page 2 of Box's paper, *Robustness in the Strategy of Scientific Model Building* in Robustness in Statistics: Proceedings of a Workshop (1979) edited by RL Launer and GN Wilkinson May 1979.

[30] Malcolm Williams *Science and Social Science: An Introduction.* Psychology Press 1999.

[31] Gordon Burt *Conflict, Complexity and Mathematical Social Science* 2010.

[32] Coleman,J.S. *An introduction to mathematical sociology*. New York: Free Press. 1964.

[33] Balci *Balci book*. 1988.

[34] Heath *Heath book*. 2009.

[35] Banks *Banks book*. 1998.

[36] Davies *Davies book*. 2003.

[37] Axelrod *Axelrod book*. 1997.

[38] Eric Bonabeau *Agent-based modelling: Methods and techniques for simulating human systems, PNAS, vol 99* 2002.

[39] Robert L. Axtell *WHY AGENTS? ON THE VARIED MOTIVATIONS FOR AGENT COMPUTING IN THE SOCIAL SCIENCES.The Brookings Institution* 2000.

[40] D. Helbing, I. Farkas, and T. Vicsek *Simulating dynamical features of escape panic. Nature 407, pg 487-490.* 2000.

[41] Melanie Mitchell *Complexity, a Guided Tour.* 2009.

[42] John Holland *Philosophica 59 (1997, 1) pp. 11-40. Emergence.* 1997.

[43] John H. Miller, Scott E. Page *Complex Adaptive Systems: An Introduction to Computational Models of Social Life (Princeton Studies in Complexity)* 2007.

[44] Tamas Vicsek *Complexity. The bigger picture. Nature 418, pg 131.* 2002.

[45] Sontakke and Singh 1996 Sontakke et al. 2008 Attri and Tyagi 2010 Singh and Ranade 2010 *www.tropmet.res.in* 1871 - 2008.

[46] Takahashi, Shingo; Sallach, David; Rouchier, Juliette *Advancing Social Simulation: The First World Congress, Springer, p. 354, ISBN 978-4-431-73150-4*. 2007.

[47] Claudius Gross *Complex and Adaptive Dynamical Systems.* 2008.

[48] Holling CS *Resilience and Stability of Ecological Systems Annual Review of Ecology and Systematics* Vol. 4: 1-23. 1973.

[49] Richard Dawkins *The Selfish Gene.Oxford University Press, 2nd edition* 1990.

[50] Tim Lewens *"Cultural Evolution", The Stanford Encyclopedia of Philosophy*. (Spring 2013 Edition), Edward N. Zalta (ed.), URL = ¡http://plato.stanford.edu/archives/spr2013/entries/evolution-cultural/¿. 2013.

[51] Mielke, A. and Roubicek, T.. *"A Rate-Independent Model for Inelastic Behavior of Shape-Memory Alloys"*. Multiscale Model. Simul. 1 (4). 2003.

[52] Balla, R., and Fern, A. *UCT for tactical assault planning in real-time strategy games*. In Proc. IJCAI-09, 40–45. 2009.

[53] Hector Geffner, Blai Bonet. *Action Selection for MDPs: Anytime AO* Versus UCT.* Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence. 2012.

[54] *Repast for High Performance Computing* : http://repast.sourceforge.net/repast_hpc.html 2012.

[55] *GridABM* http://gridabm.sourceforge.net/ 2010.

[56] *NetLogo* http://ccl.northwestern.edu/netlogo/index.shtml

[57] *http://senseis.xmp.net/?UCT*