Evolving Agent Societies with VUScape

P.C. Buzing and A.E. Eiben and M.C. Schut

Department of Artificial Intelligence
Vrije Universiteit Amsterdam
1081 HV Amsterdam, The Netherlands
{pcbuzing, gusz, schut}@cs.vu.nl

Abstract. The main contribution of this paper is twofold. Firstly, it presents a new system for empirical investigations of evolving agent societies in SugarScape-like environments, which improves existing Sugarscape testbeds. Secondly, we introduce a framework for modelling communication and cooperation in an animal society. In this framework the environmental pressure to communicate and cooperate is controllable by a single parameter. We perform several experiments with different values for this parameter and observe some surprising outcomes.

1 Introduction

There are quite a few systems enabling experimental investigations with artificial societies based on the SugarScape framework [5], for example, [8, 2]. The added value of our new system VUSCAPE comes at two levels: on the conceptual level, it incorporates significant extensions of the SugarScape framework itself, and on the technical level it allows quick and easy specification of system variants together with an extensive monitoring of system behavior[1].

Our implementation offers practical advantages over existing social simulation software like Ascape [8], Repast [2] and Swarm [3]. Firstly, VUSCAPE gives the user more flexibility because all settings can be specified either in configuration files or by command line arguments. This enables the user to automate experiments, which substantially speeds up the time needed for, for example, investigating effects of varying experimental parameters (often combinatorial number of sessions). Data is automatically saved at specified locations, enabling detailed experimental logging. Secondly, VUSCAPE is programmed and documented such that it is very easy to add, replace or delete code when changes or extensions to the model need to be implemented. Finally, there is no direct necessary connection between the actual program and the graphical user interface. Initial exploration can be easily done by using the graphical user interface; for automated experimentation, the system can be run solely by configuration files or command line. In this paper, we use VUSCAPE to perform experiments on the effects of communication and cooperation (inspired by [7, 1]).

This paper is organised as follows. In Section 2 we present the artificial world VUSCAPE, that is used in the empirical study described in this paper. The built-up of individual agents is discussed in Section 3; our agents exhibit specific behaviours in this

[1] The source code can be found at http://www.cs.vu.nl/ci/VUScape/.

study, which we briefly summarise in this Section. Section 4 presents the empirical results as obtained in the VUSCAPE investigations. Finally, in Section 5 we summarise our findings and present issues for further research.

2 VUSCAPE

The artificial world used in this paper is VUSCAPE, inherently based on the well known SUGARSCAPE world, as introduced by Epstein and Axtel [5] as a generic testbed for social simulation. For the purpose of the study described in this paper, we extended the SUGARSCAPE world in a number of ways, thereby introducing the possibility to research the specific emergent phenomena of our interest. Additionally, these adaptations can be considered to extend the SUGARSCAPE domain in an interesting generic way, opening up possibilities to investigate SUGARSCAPE worlds in wider perspectives.

Like SUGARSCAPE, the VUSCAPE world is a two dimensional grid, wrapped around the edges. Each position corresponds with an area which can contain multiple agents and an amount of sugar.

We made the following adaptations to the SUGARSCAPE world:

- **Cooperation** – Cooperative behaviour of agents is stimulated by means of setting a maximum amount of sugar that an agent can eat on its own. This maximum amount of sugar can be set by the *cooperation threshold* parameter described in Section 3.
- **Communication** – Agents receive information (listen) from other agents about amounts of sugar at their locations and broadcast (talk) the amount of sugar at their own locations.
- **Explorative Behaviour** – Agents in VUSCAPE have explorative behaviour as such that an agent randomly moves around in case it does not know of any sugar to move to or to eat. Traditionally, in SUGARSCAPE, agents do not exhibit such explorative behaviour but stay at their location in such situations.
- **Increased grid-point inhabitance** – We allow for multiple agents to be at a single grid-point in VUSCAPE, whereas this is not allowed in SUGARSCAPE.
- **Randomised sugar distribution** – The conventional sugar distribution in SUGARSCAPE is based on two sugar-hills that are located in the world at appropriate distances from each other. In this world sugar is concentrated and grows back at the given location making passive (i.e., not too mobile) agents viable. As we are interested in explorative behavior, we initially distribute so called sugar seeds randomly over the grid and move a seed to another random location if its sugar has been eaten.
- **Randomised age initialisation** – As in SUGARSCAPE models, the age parameter of agents is initialised uniformly (at 0), this brings with it some surprising phenomena that are not necessarily realistic. Additionally, the effects of such a setup echoes through the whole simulation until finished. We initialise the ages of agents randomly (between fixed minimum and maximum values), thereby preventing the mentioned methodological errors resulting from initialisation at 0.

We can catagorise these changes over two dimensions. Firstly, we can divide them according to whether a change is *methodological* or *experimental*. For example, the moving sugar seeds and randomised age initialisation are methodological, i.e., we do not explore their effects, but only investigate worlds that work according to these modifications. On the other hand, the cooperation threshold and communication are experimental extensions, as such that we describe empirical findings on their effect in Section 4. Secondly, some changes relate to the world, while other changes relate to the agents.

2.1 Execution Cycles

The VUSCAPE world evolves with discrete time steps, called cycles. In one such an execution cycle, the world (including agents) is updated. For the agents, this means that they can generate and execute actions; the world then processes these actions and generates outcomes, feeding these back to the agents. More precisely, the following stages take place in chronological order within a single execution cycle. During a single cycle, all stages are executed for each agent in parallel[2].

1. An agent *gathers information* about the presence of sugar in the world. This is done by means of listening (from other agents along the axes) and looking (by looking at the directly surrounding locations and the current location). Upon completion of this stage, the agent has at its disposal an array of locations and amounts of sugar on these locations.
2. Based on this array, the agent picks out the location with most sugar and *moves* to this location. In case there are multiple locations with the most amount of sugar, the agent chooses a random one from these locations and moves there[3].
3. Having arrived at the sugar, this sugar is *harvested* in case the amount is under the cooperation threshold. If the amount is above the cooperation threshold, the agent cooperates immediately if there are more agents at the location. Otherwise, it *broadcasts* (with some probability) to the other agents among the same axes that it needs help.
4. If possible, the agent *reproduces* and generates offspring. For this, it is (at least) necessary that there is another agent of the opposite sex at the location. (Other conditions to reproduction are applicable, and are discussed in detail below.)

In Figure 1, the agent control as described above is shown. Additionally, it shows the way communication in VUSCAPE is currently implemented.

2.2 World Features

A world in VUSCAPE has specific features that can be set as parameters in empirical investigations performed in the testbed. We describe the features below.

Traditionally, the initial *population size* can be set. Typically, in SUGARSCAPE this is set to 400. An initial population is usually half female, half male. The *width* and

[2] Conceptually, all agents execute these stages in parallel. However, technically, the stages are partially executed in sequence. Therefore, the order of agents, performing their control loops, is randomised over the execution cycles to prevent order effects.

[3] Choosing the closest by (and breaking ties randomly) is the method traditionally used in SUGARSCAPE. We have the option in VUSCAPE to either choose a random one or the closest by. Since a move action does not have a cost associated with it, we do not consider choosing the closest by having a reasonable rationale.

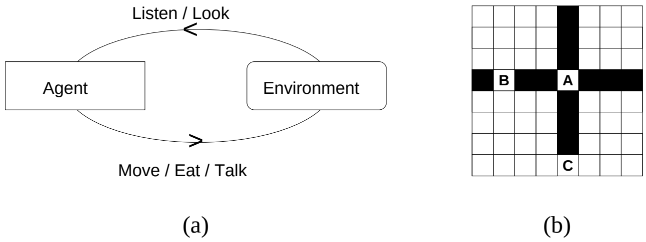


Fig. 1. (a) The agent control loop in VUSCAPE. (b) Communication in VUSCAPE over the axes. In this example, agent A broadcasts information, which is received by agents B and C.

height of the world can be set, typically to 50×50. Finally, the *sugar grow-back rate* determines at which speed sugar grows back in the world.

As mentioned above, sugar is randomly distributed over the world during initialisation in VUSCAPE. The amount of sugar to be distributed can be set by the *sugar richness* parameter: this is the average amount of sugar that is on each location. Sugar richness of 1.0 means that on a 50×50 world, 2,500 units of sugar are distributed. Possibily, there are more seeds per location. The actual sugar that can be consumed by agents grows from sugar seeds. Each sugar seed has a maximum amount of sugar it can grow back to after consumption. For example, when a sugar seed with maximum 4 is consumed, it grows back to 4 until consumed again. In VUSCAPE, this can be set by a parameter; usually, one allows for sugar seeds with maximum grow-backs 1, 2, 3 and 4. The initial sugar distribution determines what seeds are on which location. Each seed is initialised with its maximum sugar, achieving the initial sugar richness as desired. Although VUSCAPE allows for other possibilities here we initialize in such a way that the number of sugar seeds is equal for all maximum grow-back values. For 2,500 sugar units, this means that there are 250 seeds with maximum 1, 250 with maximum 2, etc. thus amounting to 2,500 in total.

In order not to depend too much on the initial sugar distribution, we allow for the sugar to be redistributed during the existence of the world. Therefore, in VUSCAPE, seeds can be moved to another location after consumption. A parameter allows for this option to be on or off. By doing this, we stimulate agents to travel more to other locations. For example, it is not possible anymore for an agent to stay put at a sugar seed is that sufficiently feeds the agent; instead, it encourages the agent to travel to find even better locations.

Monitors In VUSCAPE it is possible to monitor a variety of variables as the basis of obtained empirical findings. Although a number of monitors have been predefined, it is possible to put a monitor on every numeric variable in the source code. If a variable is monitored, this means that every execution cycle, the value[4] of the variable recorded.

[4] Each variable can monitored as average, minimum, maximum, sum, variance, standard deviation, frequency, or any combination of these.

Type	Name	Denotes	Domain
Agent	age	age of the agent	[0:100]
Agent	listenPref	whether agent listens	[0:1]
Agent	talkPref	whether agent talks	[0:1]
Agent	sugarAmount	sugar contained by an agent	[0:∞]
Agent	inNeedOfHelp	percentage of agents on sugar > coopThresh	[0:1]
Agent	cooperating	percentage of agents that cooperates	[0:1]
Agent	exploreCell	percentage of agents that moved to a new cell	[0:1]
Agent	hasEaten	amount of food that agent has eaten	[0:4]
World	numberOfAgents	number of agents	[0:∞]
World	numberOfBirths	number of just born agents	[0:∞]
World	numberOfDeaths	number of just died agents	[0:∞]

Table 1. An overview of VUSCAPE monitors.

In Table 1 we briefly enumerate the predefined monitors here. Generally, monitors refer either to variables within the *agent* or within the *world*[5].

3 Agents

Agents have a specific genetic make-up in VUSCAPE. In this Section we describe the particular *features* of an agent and the specific *behaviours* that the combination of features brings forth.

3.1 Agent Features

An agent consists of and possesses some particular *features* that determine the make-up of a particular agent. In VUSCAPE, a number of these features have directly been taken from the original SUGARSCAPE model. Additionally, we extended these features by including a cooperation threshold, reproduction threshold and initial amount of sugar.

From the traditional SUGARSCAPE model [5], we took the basic agent features which make up an agent. These features include metabolism, gender, child bearing, death, vision, allow sex and replacement.

Each agent has at its disposal a maximum amount of sugar that it can harvest on its own. As mentioned previously, this amount is called the *cooperation threshold* (or:*maximum sugar harvest* MSH). If an agent is at a location at which the amount of sugar is over this threshold, it needs other agents to harvest the sugar. If there are more agents at such a location, these agents harvest the sugar together and the sugar is evenly distributed over these agents. In the empirical investigations described below, the cooperation threshold is the same for all agents.

As agents realistically need energy (here: sugar) to reproduce, VUSCAPE offers the possibility to set the amount of sugar needed for mating by setting the *reproduction threshold*. If the amount of sugar contained in an agent is over this threshold, then

[5] The world is called a *scape* in VUSCAPE.

(in prevailing circumstances) this is agent is able to reproduce. The offspring of two parents receives half of the amount of sugar from each parent at birth. Whereas in SUGARSCAPE, the reproduction threshold (called *endowment* in SUGARSCAPE) is implemented as being the same value as the initial amount of sugar an agent received, the VUSCAPE implementation enables one to set this parameter independently of the initial amount of sugar.

3.2 Agent Behaviours

For agents, we distinguish four types of behaviour: mating, consumptional, communicative and cooperative behaviour, similarly to [4]. With the exception of communicative behaviour, these behaviours are hardwired into the agents and, as such, do not evolve during a run.

Mating Behaviour Agents reproduce like in the original SUGARSCAPE [5]. Mutation can be applied to the talk and listen genes: after each value has been inherited, a random value, drawn from a Gaussian distribution with zero mean, is added to the inherited value (similar to [6]). In cross-over, the preferences for talking and listening are both inherited as the values for these characteristic of the wealthiest parent (the one with most sugar).

Consumptional Behaviour The consumptional behaviour prescribes an agent to always go to the location with the most sugar out of all perceived locations. If there are multiple such locations, the agent picks a random one from these locations.If an agent has arrived at a location with sugar, it eats the complete amount of sugar if it can, i.e., if the amount is under the cooperation threshold. If the amount is over this threshold, the agent cannot consume, but might communicate to other agents that it needs help.

Communicative Behaviour Two features of an agent determine its communicative behaviour, namely the features to *talk* and to *listen*. The listen feature is used in the observation process of the agent. By listening, the agent receives information from other agents about amounts of sugar at the locations of those agents. The talk feature determines whether the agent performs a communicative action itself, namely broadcasting to other agents: 1) the amount of sugar that is on its location, and 2) the coordinates of its location. After initialisation, the average talk preference over all agents is 0.5. With a preference p, an agent broadcasts the amount of sugar at its location with probability p in case it needs help to harvest the sugar at its location.

Cooperative Behaviour The maximum amount of sugar at one location that an agent can eat alone is called the cooperation threshold. This parameter allows for tuning the necessity to cooperate. If it is zero, all pieces of food must be eaten by at least two cooperating agents. Gradually raising this value the necessity to cooperate diminishes.

Explorative Behaviour – Agents in VUSCAPE have explorative behaviour as such that an agent randomly moves around in case it does not know of any sugar to move to or to eat.

4 Experiments

In this section we describe our experiments with different values for the cooperation threshold, or maximum sugar harvest, MSH. Its value is gradually varied from 0 (every

Parameter	Value	Parameter	Value
Height of the world	50	Minimum death age	60
Width of the world	50	Maximum death age	100
Initial number of agents	400	Minimum begin child bearing age	12
Sugar richness	0.5	Maximum begin child bearing age	15
Sugar growth rate	3.0	Minimum end child bearing age male	50
Minimum metabolism	1.0	Maximum end child bearing age male	60
Maximum metabolism	1.0	Minimum end child bearing age female	40
Minimum vision	1.0	Maximum end child bearing age female	50
Maximum vision	1.0	Reproduction threshold	5
Minimum initial sugar	50.0	Mutation sigma	0.01
Maximum initial sugar	50.0	Sex recovery	0

Table 2. Experimental settings.

piece of food must be eaten by at least two agents) to 4 (cooperation is never necessary). The other parameters of the system are summarized in Table 2.

Additionally to varying the MSH we also test two options for the communication facility. Runs with communication use the talk and listen features as described in Section 3, runs without communication have them disabled. In our present setup agents in a communicating society always listen, only their ”talkativeness” is determined by an individual gene.

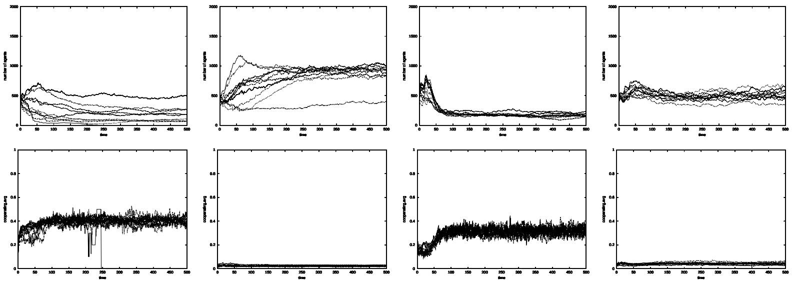
The system's behavior is illustrated here by two measures, the population size and the number of cooperation acts (when two or more agents effectively eat a pile of sugar together). Figure 2 shows the outcomes of 10 independent runs overlaid[6].

5 Discussion

Somewhat to our surprise the most visible effect of communication is not that it implies larger surviving populations, but that it makes the behaviour of the system more stable with much less variation than without communication. Using the communication feature all runs have terminated, while without it some ended in extinction by starvation. Yet, our experimental data does not supply hard evidence for the superiority of a communicating species. At the moment we are investigating other communication schemes (e.g. spherical instead of axis-wise) to check if this is an implementation artefact or has a conceptual reason.

As for the number of cooperations, we can observe that for a hard world (with low MSH) it is greater: more agents cooperate in such worlds. This is in line with our expectations. We observe that in these worlds, the difference between the cooperation curves with and without cooperation is relatively small. It can be hypothesised that the random

[6] To save space, the graphs are plotted small, only clearly revealing the trends as mentioned in the text. The reader can find these and the omitted Figures in readable format on http://www.cs.vu.nl/ci/VUScape/graphs/.



(a) (b) (c) (d)

Fig. 2. Population size (above, from 0 to 2000 on y-axis) and percentage of cooperations (below, from 0 to 1 on y-axis) over 500 execution cycles (on x-axis). Results are shown (from left to right) for MSH = 0 (a) and (b) MSH = 1 *without* communication and for MSH = 0 (c) and MSH = 1 (d) *with* communication.

exploratory behaviour of the agents results in sufficient random encounters to support large populations. Additionally, we have checked the evolution of the distribution of the "talkativeness gene". Preliminary results indicate a tendency to become more talkative over time, which suggests an evolutionary advantage of talking and thereby initiating cooperation. However, this issue needs further investigation.

References

1. E. Avdis and K. Dautenhahn. Self-organisation of communicating agents - linguistic diversity in populations of autonomous agents. In *Proceedings of the Eighth Symposium on Intelligent Robotic Systems*, 2000.
2. N. Collier. Repast: An extensible framework for agent simulation. Technical report, Social Science Research Computing, University of Chicago, Chicago, Illinois, 2000.
3. M. Daniels. Integrating simulation technologies with swarm. In *Proceedings of the Workshop on Agent Simulation: Applications, Models, and Tools*, University of Chicago, Illinois, 1999.
4. A. E. Eiben, D. Elia, and J. I. van Hemert. Population dynamics and emerging mental features in AEGIS. In W. Banzhaf, J. Daida, A. Eiben, M. Garzon, V. Honavar, M. Jakiela, and R. Smith, editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 2, pages 1257–1264, Orlando, Florida, USA, 13-17 July 1999. Morgan Kaufmann.
5. J. Epstein and R. Axtell. *Growing Artificial Societies: Social Science From The Bottom Up.* Brookings Institute Press, 1996.
6. D. Hales. Evolving specialisation, altruism and group-level optimisation using tags. In J. Sichman, F. Bousquet, and P. Davidsson, editors, *Multi-Agent-Based Simulation II. Lecture Notes in Artificial Intelligence 2581*, pages 26–35. Berlin: Springer-Verlag, 2002.
7. J. Noble. Cooperation, conflct and the evolution of communication. *Journal of Adaptive Behaviour*, 7(3/4):349–370, 1999.
8. M. Parker. Ascape: Abstracting complexity. Technical report, Center on Social and Economic Dynamics, The Brookings Institution, Washington D.C., Washington, USA, 2000.