



Nigel Gilbert, Matthijs den Besten, Akos Bontovics, Bart G.W. Craenen, Federico Divina, A.E. Eiben, Robert Griffioen, Gyorgy Hévízi, Andras Lőrincz, Ben Paechter, Stephan Schuster, Martijn C. Schut, Christian Tzolov, Paul Vogt and Lu Yang (2006)

## Emerging Artificial Societies Through Learning

*Journal of Artificial Societies and Social Simulation* vol. 9, no. 2

<<http://jasss.soc.surrey.ac.uk/9/2/9.html>>

For information about citing this article, click [here](#)

Received: 02-Sep-2005 Accepted: 10-Dec-2005 Published: 31-Mar-2006



### Abstract

The NewTies project is implementing a simulation in which societies of agents are expected to de-velop autonomously as a result of individual, population and social learning. These societies are expected to be able to solve environmental challenges by acting collectively. The challenges are in-tended to be analogous to those faced by early, simple, small-scale human societies. This report on work in progress outlines the major features of the system as it is currently conceived within the project, including the design of the agents, the environment, the mechanism for the evolution of language and the peer-to-peer infrastructure on which the simulation runs.

### Keywords:

Artificial Societies, Evolution of Language, Decision Trees, Peer-To-Peer Networks, Social Learning

### Introduction

#### 1.1

One goal of social simulation is to develop models that shed some light on the functioning of human societies. The advantages of a simulation approach to understanding human societies include the requirement to express theories in complete and unambiguous terms; the opportunity to derive the implications of proposed social mechanisms; and the possibility of performing experiments on the simulated society ([Gilbert 2005](#)). As a result of these advantages, there has been a rapid growth in the popularity of social simulation over the last decade ([Gilbert & Troitzsch 2005](#)).

#### 1.2

The NewTies project [\[1\]](#) aims to discover whether an artificial society can 'construct itself' with only the bare minimum of experimenter-provided rules or theory. It takes inspiration partly from work on the evolution of language, which has shown that, given a capacity to learn, artificial agents are capable of developing a simple 'language' with which to communicate (see Cangelosi & Parisi ([2002](#)) for an overview). Initially agents utter only random noise with no information content, but through repeated interactions, some of which are rewarded, the agents

gradually develop a shared lexicon, 'a consensus on a set of distinctions' ([Hutchins & Hazlehurst 1995](#)).

### 1.3

If agents can develop a lexicon from 'nothing', could they also develop a shared culture? Taken strictly, the answer must be 'yes', since language and thus a lexicon is an important part of human culture. But we wish to see whether agents can develop culture in a wider sense, as a set of shared behaviours and a common understanding of the society and environment in which they live. This culture, like the shared lexicon, must be developed collaboratively by the agents from 'nothing'. This means that we give the agents the ability to learn, but do not direct them about what to learn, and initialise them with a bare minimum of knowledge about their worlds. However, the agents are given a rich and extensive simulated environment in which to learn how to survive.

### 1.4

The agents are constructed to be able to learn in three ways:

1. Individual learning through trial and error.  
Agents act according to their genetic pre-dispositions, overlaid with random variations. Some actions are more effective than others. Those actions that succeeded in the past are remembered and the agent is then more likely to repeat them.
2. Population learning through reproduction and selection  
Agents with predispositions to carry out effective actions are more capable and are therefore more likely to reproduce, transferring a version of their genetic material to their offspring. Thus the population of agents as a whole will tend to become more successful over the course of many generations.
3. Social learning  
Neither individual nor population learning require any communication between agents. However, these types of learning could be the means by which the agents begin to develop the capacity to communicate. If they do so, they can start to use a more direct and effective mode of learning: that of one agent teaching another.

### 1.5

To achieve complex behaviours, the NewTies project will study very large populations. The resulting system will allow for sufficient complexity at the macro level to allow interesting behaviours to emerge in separate parts of the system. Experiments of the proposed size necessitate the use of a large distributed computing infrastructure. This is being implemented as a shared platform communicating in a peer-to-peer fashion. This paper also addresses some of the problems that arise in designing such an infrastructure.

### 1.6

The next section of the paper enlarges on the types of learning available to agents. The following two sections introduce the environment and the challenges that the agents face. The fifth section describes the interface between the environment and an agent. Agents perceive their surroundings and act in their world through this interface. In section [6](#), the ways that individual and evolutionary learning are being implemented are described, and section [7](#) introduces the mechanisms that support language evolution. Section [8](#) describes the design used to distribute the simulation across many processors. The paper concludes by emphasising the importance of the design of the environment for social simulation and suggesting that this aspect has too often been neglected.



## Environmental challenges

### 2.1

If agents are to learn, they must have some motivation to do so. In the NewTies project, that motivation [\[2\]](#) is ultimately that of their survival. Agents are placed in an environment which they find individually and collectively challenging. Unless they master survival in this environment they will 'die'. This notion is operationalised by constructing environments in which there is a limited amount of 'food' to provide agents with energy and then requiring the agents to maintain at least a minimum energy level. At first, agents have to act on their own, since they

have not yet learnt to act collectively. Those that manage to gather sufficient food from the environment may survive long enough to breed, while those that are less successful are more likely to 'starve'. The environment thus imposes a strong selection pressure on the agents. Eventually, the agents may discover how to communicate and then be able to engage in collective action. This is likely to be more effective than individual acts in obtaining food from the environment.

## 2.2

The fine detail of the environmental challenge is an extremely important factor in the agents' development. First, if obtaining food is too easy, the agents will not need to learn much, and will probably not do so. If the environment is too unfriendly, all the agents will die of starvation before they have had a chance to learn anything. If the environment does not pose problems that are solved as a group more effectively than individually, there will be no reason for agents to learn how to act together. Secondly, if the environment requires agents to engage in activities which they are not able to carry out, the agents will surely fail, since they are only able to increase their knowledge, but not their repertoire of basic actions. For example, humans have learned to fly, not by growing wings, but by learning how to build aircraft. Thirdly, the long-term objective of the research is to understand human societies better. The environment must set challenges that are analogous to those faced by humans if there is to be even the possibility of reading across from the simulation to human development.

## 2.3

We have designed four environmental challenges, each based on a well studied aspect of human society. In the descriptions below, the human system is first summarized, the challenge stated in terms of the simulated environment, and the expected outcome is specified.

### **The Kula Ring**

## 2.4

A complex system of visits and exchanges among the Trobriand Islanders of the western Pacific was first described by Bronislaw Malinowski ([1922](#) [[1978](#)]). Necklaces were exchanged in one direction among the residents of a chain of islands and armbands exchanged in the opposite direction (hence the notion of a ring). These exchanges did not primarily serve an economic function but created a network of social obligations which could be depended upon at various times in an individual's life. In particular, the social network seems to have been the basis for economic relationships such as trading food for pottery.

#### **The challenge parameters:**

Food is distributed in spatial patches and the amount of food in a patch varies over time. The overall quantity is more than enough to feed the population, but there may be short-term local shortages. These can be alleviated by trading or by theft. Trade is less costly in energy, but requires the prior development of mutual trust by the traders.

#### **Expected outcome:**

The establishment of a 'gift-exchange' system in which not only food but also tokens are exchanged.

### **Herders in a semi-arid area**

Nomadic herding is another human solution for dealing with variable and uncertain shortages. Herders and their cattle move to where food is available, leaving exhausted areas until the grass has re-grown. This requires herders to find ways of managing common pool resources (the grass) so that no individual herder overgrazes the grass. The human solution involves well developed status hierarchies and no private property.

#### **The challenge parameters:**

Plants yielding food are randomly distributed with the mean level of food just sufficient to support the population. The rate of plant growth varies randomly over time. Food is perishable. Some plants must be left unharvested on each patch since their subsequent growth and yield of food is proportional to the number of plants remaining.

**Expected outcome:**

Agents leave unharvested plants when they move away, even if they leave hungry.

### **Central place theory**

Walter Christaller developed Central Place theory in 1933 ([King 1985](#)) to explain the size and spacing of cities that specialize in selling goods and services. The theory consists of two basic concepts:

- threshold -- the minimum market needed to bring a firm or city selling goods and services into existence and to keep it in business
- range -- the average maximum distance people will travel to purchase goods and services

The theory predicts that settlement size will follow the rank size rule. It works well for human settlements.

**The challenge parameters:**

The distribution of types of food is such that agents need to trade food with other agents. The food types vary in their transportability. Agents can move to find the best location to maximise their income from trade.

**Expected outcome:**

Agents settle into spatial clusters separated by relatively empty areas. The size of the clusters is power law distributed.

### **Branding**

When producers produce and consumers consume complex goods (i.e. ones with a large number of distinct attributes), and there are a large number of producers and consumers, search problems occur. Producers find it hard to locate consumers that desire goods having the precise set of attributes that a producer is selling, and consumers find it hard to identify producers with the desired goods. One solution to the problem that each side faces is for producers to brand their range of goods (targeting them at a subset of consumers) and for consumers to use the brand as the major preference criterion. Similar processes may help to account for prejudice and discrimination among human populations.

**The challenge parameters:**

Agents have characteristic sensible attributes ('tags'). Agents seek to locate other agents with a similar or identical set of tags (through movement and communication), but this search is expensive. Agents are able to create additional tags (the brand) by collecting tokens and carrying them around.

**Expected outcome:**

Agents either generate one additional tag or specially distinguish an existing tag and this becomes a linguistic category that labels agents and leads to differences in behaviour towards those agents that are labelled and those that are not.



### 3.1

An environment that offers these challenges to agents must be sufficiently rich in features to allow each challenge to be constructed, but also no more complicated than necessary. Any features beyond the minimum required would slow down the simulation and, crucially, make the agents' task of learning how to manage in the environment more difficult, because they would need to learn to disregard irrelevant features.

### 3.2

The NewTies environment consists of a very large simulated flat surface over which the agents are able to move. The surface is divided into cells or 'locations'; an agent or other object is of a size that it occupies exactly one location. A virtual clock counts 'time steps', used primarily to synchronise the agents' actions. To remain in accord with the real world, agents do not have direct access to their location on the surface, nor to the time. They are, however, able to detect geographical features ('places') and the relative position of the 'sun', an object which slowly traverses the surface, crossing it once per simulated day (there is no night — the sun is always visible). Places are bounded areas of the landscape which differ from the rest of the surface in having a varied, but lesser degree of roughness, making it easier for agents to move within places than in the wilderness outside places.

### 3.3

On the landscape are a number of objects as well as the agents: tokens, plants, and paving stones. Tokens are distinguishable, moveable objects, some of which can be used as tools to speed up the production of food, but most of which have no intrinsic function, and can be employed by agents as location markers, symbols of value (e.g. 'money'), or for ritual purposes.

### 3.4

Plants are the source of food. They are annuals, living for one year. At the beginning of the year, eating them gives agents little energy, but as the year progresses, they ripen and become better food. In the 'autumn', their energy value decreases again, and is entirely lost at the end of the year when they die. However, before they die, they produce two seeds, one at the parent plant's location and one in an adjacent location. If a seed is the only one in the location, it grows, but if there are more than one, only one will survive. If a plant is picked by an agent, it starts decomposing and will lose all its goodness if not consumed or replanted within a few days.

### 3.5

Agents lose energy (the rate depending on the roughness of the location) when they move over the landscape. The effort required to move can be reduced by building roads. Roads are constructed from paving stones laid besides each other. Stones can also be placed on end to build walls that obstruct agents attempting to cross them.

### 3.6

With these simple ingredients, we can construct scenarios corresponding to each of the challenges. For example, the Trobriand Islands can be represented as places, with the rest of the surface (having a very high value of roughness) representing the sea. The varied availability of food among the Islands (and the seasonal availability of crops) can be represented by arranging the plants in the places. The agents can learn to use tokens as symbolic gifts. Economic trading between islands could involve exchanges of food and of token tools. The other challenges could be modelled by constructing scenarios in similar ways. For example, the 'branding' challenge would involve agents trading many similar but not identical tokens between themselves, with search being costly (i.e. the roads are rough).



## The Agent-Environment Interface

### 4.1

To survive in this environment, agents need to be able to perceive the landscape and the objects in it and able to act on objects and other agents. Moreover, it is expected that experiments will be carried out using a variety of agent designs, possibly including agents constructed outside the NewTies project, and so a simple and precisely specified interface between the agents and the environment is desirable.

## 4.2

At each time step, every agent is given a slice of computational resource. During this step, it must complete two phases in sequence: a *perceive* phase and an *act* phase. During the perceive phase, an agent is given the following information about the environment:

- a. a list of the attributes (type, characteristics, colour, heading, and weight) of each object located within a segment defined by the direction in which the agent is facing, plus or minus 45°. The information returned about each object also includes its distance and direction from the agent and, if the object is an agent, its age and sex. These data do *not* include any direct indicators of the objects' identities; the agents have to infer these from the objects' attributes..
- b. A list of the places in which the agent is located (places can overlap, so there may be more than one).
- c. The agent's own current energy level.
- d. A list of the attributes of all the objects that the agent is currently carrying.
- e. The roughness at the current location.
- f. The result of the action performed in the Act phase of the previous time step, if any.
- g. A list of messages that other agents have sent during the preceding Act phase, including any mating proposal.

## 4.3

The agent is able to process this information as it wishes and can then carry out one action, chosen from the following:

- **Move:** The agent moves from its present location to an adjacent location in its forward direction.
- **Turn left / turn right:** the agent rotates in the indicated direction by 45 degrees.
- **Pick up object:** The agent acquires the object. The object remains with the agent until the agent puts it down or eats it (if the object is food).
- **Put down object:** The agent puts the object down at the current location.
- **Give object:** The agent transfers an object in its possession to another agent. The receiving agent must be in an adjacent location.
- **Take object:** The agent takes an object from another agent. The donating Agent must be in an adjacent location.
- **Build:** By modifying stones or putting new ones into the landscape, agents can build two types of structures: roads, which decrease the roughness of the environment and make travelling easier, and fortifications, which increase the roughness and make it more difficult to move over.
- **Talk to agent:** The recipient agent must be 'visible' to the speaker (An agent cannot talk to another agent while facing away from that Agent, but the hearer does not have to be facing the speaker). A character string emitted by the speaker is conveyed to the listener. The effect is that both the listener and the speaker are given the character string during the next Perceive phase.
- **Shout:** A character string emitted by the shouter is conveyed to all agents within a short distance (including the shouter itself) during the next Perceive phase.
- **Hit:** The agent chooses, first, the amount of energy to expend on the blow, which must be less than the current energy level of the Agent, and, second, which agent will be the victim (the victim must be in an adjacent location). Both the aggressor agent and the victim lose energy proportional to the ratio of the weights of the aggressor and the victim. If the victim's weight decreases to zero or less as a result of the violence, the victim dies.
- **Eat food:** The agent must already be carrying the food (see Pick up object). The energy of the food is added to the agent's energy and the food 'disappears'.
- **Propose mating:** A 'father' agent proposes to a 'mother' agent to have a child and sends at the same time its genetic information. Proposals are delivered to agents at the next time step.
- **Accept mating:** If the mother agent decides to accept mating with another agent, a share of the 'father' agent's energy is transferred to the unborn child. The environment tracks the 'pregnancy' of the mother agent and at each time step transfers a certain amount of energy from the mother agent to the child. The child is then 'born'.

The information given to agents about their environment is intended to reflect the information that would be available to a human. Particular care is taken not to give agents information which would not be accessible to people. For example, the identity of other agents is not provided, only some descriptive characteristics through which agents may be recognised. However, there is no guarantee that all agents will necessarily have a unique set of these characteristics. Utterances are labelled by the system, not with the identity of the speaker, but with the speaker's characteristics, for the same reason. Speakers hear their own utterances reflected back to them, again because this is the experience of humans, who are able to monitor their own speech.

#### 4.4

Initially, agents will have no common lexicon and therefore no understanding of what other agents say to them; we expect, in the light of studies on the evolution of language, that in time the agents will develop a shared vocabulary and ultimately a shared idea of grammar (see section [7](#)). However, because of the design of the agents and the environment, it is not necessary or even likely that this vocabulary will be entirely composed of utterances (i.e. 'words'). Because talking is just one of the actions available to agents, it would be expected that some actions other than talking will come to take on meaning for the agents — in the same way as human gestures, for example, can substitute for or even be preferred to speech for conveying some meanings. This is in contrast to current studies of the evolution of language, which have generally taken a more purely linguistic approach to interaction.

#### 4.5

Although the list of possible actions may seem long, it is intended to be the minimum set that would enable the challenges to be met by the agents while yielding social behaviour comparable to that of human societies. For instance, the actions 'give object' and 'take object' are required in order to make trade a possibility. Without these actions, the only way to transfer an object from one agent to another would be for one agent to put the object down and another subsequently to pick it up. However, there would be no way for the first agent to guarantee that the second agent is the recipient, and thus directed personal transfers (required for trade) would be difficult or very risky. The justification for the 'hit' action (aside from the fact that violence is an endemic feature of human societies) is that without violence, private property cannot be preserved. An agent wanting an object in the possession of another could simply remove it and the owner would have no recourse if there were no possibility of violence. To match the human situation, an aggressor will only be effective if it is stronger (i.e.. heavier) than the victim, so we can expect weak (light) individuals to be subject to theft which they cannot resist, at least until a protective social system evolves.

#### 4.6

In this environment, agents have only one over-riding need: to obtain sufficient food to survive. Human requirements are of course more complex, involving not just a reasonably balanced diet, but also warmth and water, but we are assuming that 'food' is an adequate abstraction for these more complex needs.

#### 4.7

It is intrinsic to the implementation of population learning that agents are born, reproduce and so pass on their genotype, and die. New agents result from the coupling of a male and a female agent (hence agents need to have a gender) and are born in an adjacent location to their parents. Parents have no predisposition to attend to their offspring, but because they are nearby, are likely to interact with them more than with other agents. Parental care of offspring is likely to be selected for since neglected children will find survival even more difficult than their parents (since they have had no opportunity for individual learning). To enable adults to identify children, one of the characteristic features of agents, perceptible by other agents, is their age.

## The agents

#### 5.1

The New Ties agent's cognitive ability [\[3\]](#) is based on an extension of a decision tree, made to fit evolutionary algorithms (EA) as well as reinforcement learning (RL), and which is general enough

to incorporate tunable function approximators ([Haykin 1999](#)).

## 5.2

An agent has the following four features ([Wooldridge & Jennings 1995](#)):

1. *autonomy*: agents have some kind of control over their actions and internal state. New Ties agents possess a controller that determines their actions.
2. *social ability*: agents interact with other agents. A NewTies agent can communicate with other agents by means of talking (one-to-one communication) and shouting (one-to-many communication). This communication can change the behaviour of an agent.
3. *reactivity*: agents perceive their environment and respond in a timely fashion to changes that occur in it. A NewTies agent can see objects located within a segment defined by the agent's heading. The agents cannot identify objects directly. They can only see bundles of object features that they have to learn to classify.
4. *pro-activeness*: agents do not simply act in response to their environment, they also take the initiative. A NewTies agent looks for food, a mate, and may interact with other agents.

## 5.3

An agent's lifetime consists of three periods: a childhood period, an adult period, and a period of old age. Each period has its own properties. In childhood an agent tends to follow one of its parents. As an adult an agent can reproduce. When an agent reaches old age this ability is lost.

## 5.4

The genetic features of an agent may be passed on through reproduction. A child's gene is created from the parents' genes by recombination and mutation. The properties of an agent encoded in the genome are summarized in Table 1 below. Note that all values are scaled except `OnsetTimeAdulthood` and `OnsetTimeElderdom`. These have values matching our intuition for human years.

---

**Table 1:** Properties encoded in the genome of the agent

---

Name	Type	Value	Additional comments
Metabolism	real	[0,1]	Determines how much food is converted into energy
OnsetTimeAdulthood	int	[15,25]	Reaching the adult age the agent becomes fertile
OnsetTimeElderdom	int	[55,65]	Reaching elderdom the agent loses fertility.
Socialness	real	[0,1]	The degree to which an agent wants to interact with other agents.
FollowBehaviour	real	[0,1]	The degree to which an agent wants to follow its parents
MaxVisionDistance	real	[0,1]	How far an agent can see.
InitialSpeed	real	[0,1]	Initial distance an agent can walk per time step
InitialStrength	real	[0,1]	Initial weight an agent can lift.
MaxShoutDistance	real	[0,1]	The maximal reach of an agent's shout

---



## Individual and evolutionary learning

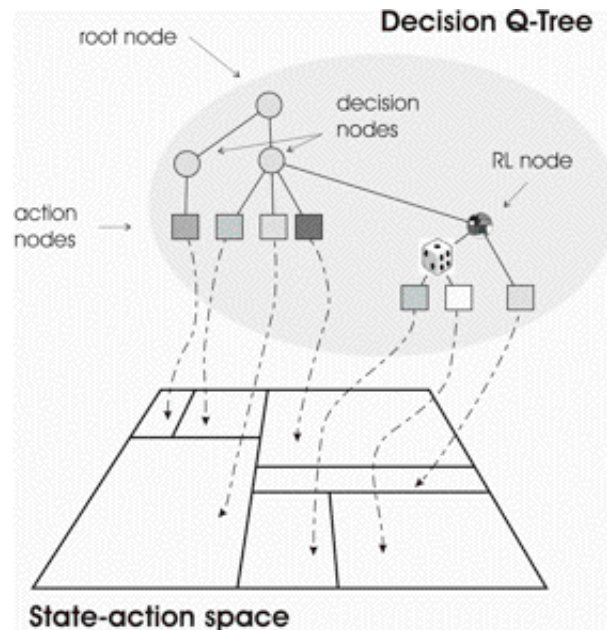
### 6.1

Individual learning is the knowledge acquired in every situation in which an agent (re-)acts and processes data — including data about its own (re-)actions — in order to improve performance in similar situations in the future. Individual learning changes the behaviour of the agent, which means that after learning the agent may produce a different action or sequence of actions in a similar situation or in response to a similar stimulus.



## 6.2

There are several ways of implementing individual learning, but for our purposes, a technique that could be integrated with population and social learning was needed. To achieve this, decision trees have been extended to what we call Decision Q-trees (DQT) (see Figure 1).



**Figure 1.** DQTs are decision trees with the following features: (1) nodes can denote rules, indices of states, and actions, (2) decisions can be stochastic (denoted by the dice), (3) nodes are assumed to have values (denoted by the gray levels of the nodes). Special cases of DQTs among others are: decision trees, the action-state value tables of reinforcement learning, and Bayesian decision networks. DQTs have large compression power in state-action mapping and suit evolutionary algorithms. DQT partitions of the state-action space can be overlapping.

## 6.3

A DQT is a value table with value entries belonging to the nodes of a decision tree. Graphically, DTQ is a tree made of nodes and the directed edges connecting them. Each node, other than those at the leaves of the tree, has a set of decision making rules. These rules can be deterministic, e.g., if-then rules or tables, or stochastic rules or network structures such as Bayesian networks. There are as many outgoing edges from a node as there are distinct cases in the rule set or entries in the table. However, the nodes at the leaves of the tree contain only a single action (which may be the null action).

## 6.4

All nodes have a value, representing the expectation value of the long-term discounted and cumulated rewards received after passing that node. Subject to certain conditions, this value can be estimated iteratively during the lifetime of the agent (see, e.g., [Sutton & Barto 1998](#)).

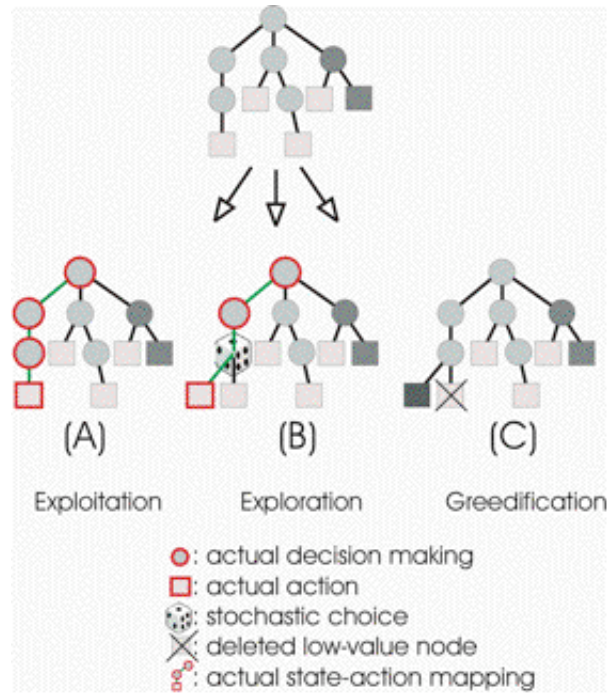
## 6.5

Reinforcement learning and Holland's Learning Classifier systems ([Holland et al. 2000](#)) have been contrasted in the literature ([Kovacs 2002](#)). The main difference is in the way these methods use and modify the policy, the state-action mapping. The DQT architecture unifies these algorithms. The smoothness of this unification underlines the similarities instead of the differences of these two paradigms. We expect to make use of the strengths of both algorithms, such as the tuning property of reinforcement learning and the compressed representations searched for and offered by genetic algorithms.

## 6.6

Individual learning occurs through exploration, i.e., the selection of previously untried nodes, the evaluation of these nodes and the "greedification" of the policy (here the tree), i.e., the deletion of actions that have low values. These operations are depicted in Figure 2. Fine tuning the values of the nodes is part of the individual learning process, and thus can be separated from evolutionary learning. During the course of evolutionary learning, the value of the root node can be treated as the 'fitness' of the tree as a whole, while the structure of tree can be

evolved using well-established methods for mutating and recombining tree-based genotypes (see [Banzhaf et al 1998](#)). The way we define the fitness of the individual resembles to genetic algorithms augmented by local searches, where the fitness of the individual is influenced by the GA defined position where the local search starts from and is determined by the value of the local maximum that attracts the local search. The difference is that for our agents, local searches are replaced by reinforcement learning and thus optimising short term gains is replaced by optimising for long-term cumulated gains.



**Figure 2.** (A) exploitation using DQT, (B) exploration and (C) "greedification" of a DQT. The value of each node is shown by the depth of the grey shading.

## Language evolution

### 7.1

Although agents start without any knowledge about the world, so that they have no representations of meaning, the aim of the NewTies project is to have the population evolve a common language with which they can communicate. Language, as we view it, can be viewed as a complex adaptive system ([Steels 1997](#)). In this view, language evolves through the interplay between cultural interactions, individual learning and self-organisation. From this viewpoint, we start by assuming that agents are equipped with basic communication skills and learning mechanisms and see how in the NewTies project a shared language can emerge and how this language evolves over time. Based on earlier work, we expect that this will present three major problems:

1. At each moment in time we aim to deal with a population of around 1,000 agents or more. No known experiment in language evolution has had such a large population size. It is expected that having all agents interact with all other agents will lead to an unrealistic scenario and require a huge number of interactions to arrive at a shared language. On the other hand, agents will need to communicate frequently enough for language to arise. So, how do we control communication?
2. There are a relatively large number of different objects (food items, tokens, agents, roads, places etc.), which are perceived by agents through a fixed, but relatively large number of feature channels. In addition, there are many actions that can be performed. How do we allow agents to categorise the different objects/actions such that they become sufficiently similar to allow language learning (cf [Smith 2003](#)), and such that these categories are not predefined?
3. The contexts in which communication takes place are acquired by the agents autonomously. As a result, these contexts may differ from one individual to another. In addition, the internal languages of two individuals may differ, for instance because one of the individuals is still a 'child'. How, then, can one agent infer the meanings of other

agents' utterances, such that a shared communication system can emerge? This problem is related to the *indeterminacy of meaning* problem ([Quine 1960](#)), which states that when learning the meaning of a novel word, this word may have — logically speaking — an infinite number of possible meanings. We therefore need to develop heuristics that allow agents to reduce the number of possible meanings.

## Large populations

### 7.2

Large populations could be regarded as an asset from which we can learn about how different dialects and languages may evolve. Nevertheless, we do not want each agent in the population to communicate with all other agents, as this will give huge convergence problems. In addition, we do not want each agent to communicate unrestrictedly with another agent, as this may lead to unlimited chat sessions among agents who happen to be near to each other. On the other hand, agents will need to talk frequently enough to allow language development.

### 7.3

Because of the nature of the environment and the low average density of the population, the first problem is unlikely to occur. In addition, agents are physically restricted to communicate only locally since the range of their verbal utterances is limited. Furthermore, agents will not always decide to talk: the decision to talk is taken by the agent controller (DQT) as one of many possible actions.

### 7.4

The tendency to talk will be based on the Socialness gene (see Table 2), and possibly on the social bond between the individual and another agent. This social bond is a variable based on the social network an agent constructs, and could be a function proportional to the number of interactions between two agents (assuming that agents have learnt to recognise each other) and the effectiveness of such interactions (cf [Gong et al. 2004](#)). Note that, this way, agents can also develop a notion of friendship: the higher the social bond, the stronger the friendship between two agents. As children will have an innate tendency to follow one of their parents (cf the `FollowBehaviour` gene), it is expected that the social bond between this parent and its offspring will become relatively high, which may give rise to a notion of kinship, but also to an increase in communicative interactions. The children's tendency to follow a parent has the further advantage that it can learn from a parent using rather consistent input. Although the type of learning used for language needs variation and selective pressures to evolve successfully, allowing children to learn from only a few adults seems to improve learning ([Vogt & Coumans 2003](#)).

## Categorising objects

### 7.5

When perceiving objects or actions, each agent individually tries to categorise the raw input for each object or action. To this aim, an agent constructs during its lifetime a set of categories from scratch. These categories are sensitive to one particular feature channel (e.g., type or colour) and have a prototypical focal point. When categorising an object, the agent will map each perceived feature with that category having the nearest focal point. These categories are then combined to form a conceptual representation of the object. Once each object is categorised this way, the agent will see for each object if its concept distinguishes this object from all other objects in the present context. (The context is the set of objects perceived at a particular time step.) If an agent fails to discriminate an object, new categories are constructed for which the object's features serve as exemplars and the object's concept is adjusted accordingly. This approach of making distinctions is based on the *discrimination game* introduced by Steels ([1996](#)). Note that if the context contains two or more objects with exactly the same features, these objects will be grouped.

### 7.6

Actions are not perceived directly, but can be identified by comparing two time steps. By calculating the difference between the two feature vectors of an object (typically another agent) as perceived in two consecutive time steps, the agents can form a feature vector of the

perceived action. Such a feature vector is then categorised in a similar way as described above. When more than one action is perceived, the agent can use the discrimination game to form distinctions and update the set of categories.

### 7.7

A similar approach will be taken for categorising an agent's own energy level, road roughness, time — as indicated by the position of the sun, and places. The agent's own possible actions are assumed to be innate categories.

### 7.8

It is important to notice that this way, categories need not be perfect exemplars of perceived objects or actions. Furthermore, each individual will construct its own private set of categories, so these may differ from one agent to another. This will particular be true for old agents who have a mature set of categories in comparison to children whose categories can be very general and overextended. However, as each agent will use the same heuristics, has the same feature channels and acts in the same environment, we expect that categories will arise that are sufficiently similar across agents to allow language learning.

### 7.9

As mentioned, categories are defined in relation to one feature channel and when combined they can form concepts representing an object or action. Our aim is to allow agents to construct general concepts that represent, e.g., food or tokens, but also concepts that are more specific, such as colour and location. Objects can thus be described with multiple concepts, which in turn may give rise to the emergence of grammar. The concepts formed are used by the agents' language modules, but also by the agents' controllers. In the latter case, it is possible that an agent will incorporate concepts as test nodes.

## Word learning and language games

### 7.10

We refer to linguistic interactions between agents as *language games* ([Steels 1996](#)). A language game — illustrated in Table 2 — is played by two or more agents: one speaker and one or more hearers. The speaker decides to communicate about some object or action (called the *target*). This target selection is primarily based the set of test nodes that the DQT traverses when deciding upon what action to take. From this set the agent constructs a (set of) concept(s) and produces an expression to convey this. If the speaker is unable to produce an utterance based on its lexicon (this happens, e.g., when the lexicon is still empty), the speaker invents a new word and adds its association with the target concept to its lexicon.

---

**Table 2:** An example scheme for playing a language game between a speaker and hearer. The game may take up to 5 time steps  $t$ . See the text for details.

---

	Speaker	Hearer
$t$		
$n$	Perceive context Categorisation/DG Target selection Produce utterance Update memory1 Send message	
$n + 1$		Receive message Perceive context Categorisation/DG Interpret utterance Update memory1 Respond

$n + 2$	Evaluate effect	
$n + 3$		Evaluate effect
$n + 4$	Update memory2	Update memory2

---

p>

### 7.11

On receiving an expression, the hearer(s) will try to interpret the expression based on its learnt vocabulary (or *lexicon*). Each time an agent produces or interprets an utterance, the agent will update its lexicon. The lexicon is maintained in two memories: one stores *a posterior probabilities* (memory1) and the other stores *association scores* (memory2). The *a posterior probabilities*  $P_{ij}$  are calculated from the co-occurrence frequencies  $u_{ij}$  with which words  $w_i$  and concepts (or meanings)  $m_j$  co-occur in a certain situation.

$$P_{ij} = P(m_j | w_i) = \frac{u_{ij}}{\sum_j u_{ij}} \quad (1)$$

When an association is used, its co-occurrence frequency  $u_{ij}$  is incremented. This update is referred to in Table 3 as 'update memory1'. The association scores  $u_{ij}$  are calculated based on the agents' evaluation of the success of the game. If the game is a success, the association is reinforced according to Eq. (2), otherwise it is inhibited using Eq. (3).

$$u_{ij} = u_{ij} + \eta \quad (2)$$

$$u_{ij} = u_{ij} - \eta \quad (3)$$

where  $\eta$  is a learning parameter (typically  $\eta=0.9$ ).

### 7.12

Whereas the former update is based on a model of cross-situational statistical learning ([Vogt & Smith 2005](#)), the latter is based on the guessing game model ([Steels & Kaplan 1999](#); [Vogt 2003](#)). We opt for combining the two as the former does not require an evaluation of a game's success, which may not always be possible. However, this cross-situational statistical learning is very slow and may yield incoherent lexicons ([Vogt & Smith 2005](#)). The guessing game, on the other hand, has proven to be a much faster learning mechanism ([Vogt & Coumans 2003](#)), so when an evaluation of a game's effectiveness is possible, this mechanism should be favoured.

### 7.13

When searching its lexicon, an agent selects the association  $\alpha_{ij}$  that is consistent with the target concept (if the agent is the speaker) or received utterance (hearer) and of which the strength  $strL(\alpha_{ij})$  is highest, where

$$strL(\alpha_{ij}) = (1 - \alpha_{ij})P_{ij} \quad (4)$$

This equation has the neat property that if the association score is low (i.e. the agent is not confident about the association's effectiveness), the influence of the *a posterior* probability is large, and when the association score is high, the influence of the *a posterior* probability is low. (For a case study using this combined model, consult [Vogt & Divina 2005](#)).

### 7.14

If the hearer is unable to interpret the expression based on its lexicon, the hearer will have to learn the expression's meaning. As mentioned, learning a word-meaning mapping is considered as the biggest problem we have to deal with, because of Quine's (1960) indeterminacy of meaning. Researchers in child language acquisition have proposed many possible principles that aid children to reduce the number of possible meanings for a word (see, e.g., [Bloom 2000](#)).

for an overview. Although there is little consensus on what mechanisms are actually used by children, we intend to implement a number of principles (or heuristics) that appear consistent with the way children learn the meanings of words.

## 7.15

Below follows a short list of principles that we intend to implement in one way or another. The general idea is that using these principles, the hearer reduces the number of possible meanings of a word (ideally to one correct meaning), and adds associations of the uninterpreted word with these meanings to its lexicon.

- a. **Principle of Contrast** ([Clark 1993](#)): According to the principle of contrast, children tend to assume that any difference in form marks a difference in meaning. This is similar to the principle of *mutual exclusivity* ([Markman 1989](#)), which relates to the idea that children use the assumption that different words do not share the same reference. Although it has been shown that applying mutual exclusivity to cross-situational statistical learning works well ([Smith 2005](#)), it must be treated carefully, since an agent may have acquired a wrong association and preventing the acquisition of another mapping could harm the learning process. Therefore, the principle of contrast must be used flexibly as a bias, rather than as a fixed principle.
- b. **Joint attention** ([Tomasello 1999](#)): Joint attention refers to the situation where participants of an interaction jointly focus attention on an object or event, while monitoring each other's attention. As this principle requires a very sophisticated monitoring mechanism, we will implement a minimal mechanism of joint attention by using pointing. Agents can use this mechanism to point at an object or action, thus indicating its reference to the other agent. Naturally, this mechanism can only be used when the object or action is present in the current context. Simulations and robotic experiments have shown that this mechanism provides a quick learning mechanism, but may introduce a lot of ambiguity in the language if joint attention is treated as a perfect deterministic mechanism, i.e. when the agents indicate exactly one object or action as the target ([Vogt 2003](#); [2005](#)). We therefore intend to allow agents to use this principle only sparsely, and perhaps only when communication fails otherwise (see next principle).
- c. **Corrective feedback**: We intend to use corrective feedback by allowing the speaker of a language game to provide the hearer with the correct reference of an expression if the language game was unsuccessful, e.g., by means of pointing at the intended target. It is unclear to what extent children receive corrective feedback on erroneous use of word-meaning mappings. However, it is clear that like in adult-adult communication ([Clark 1996](#)), participants of adult-child communication carefully monitor the correctness of the language use ([Chouinard & Clark 2003](#)). Chouinard & Clark have argued that, although, children may not be corrected by providing them with the correct meaning of a word, they are provided with the correct word for a certain meaning. Furthermore, we believe that children are well capable of evaluating the effectiveness of a language game, and alter their hypotheses about a word's meaning when they misinterpret a word. For instance, a child may notice, e.g., through self monitoring, that it misunderstood an expression such as "can you give me the X" if it handed the speaker a Y instead of X. To avoid complicated dialogues, we assume that in such cases, the speaker can point at the intended target such that the hearer can infer the expression's meaning using this extra-linguistic cue. Computer simulations have revealed that this type of learning leads to the emergence of highly efficient lexicons ([Vogt 2005](#); [Vogt & Coumans 2003](#)).
- d. **Theory of Mind** ([Premack & Woodruff 1978](#)): It is hypothesised that when humans learn the meaning of a novel word, they try to simulate the speaker's thought process to form their own theory of the speaker's intention, thus trying to understand what the speaker meant ([Gopnik & Meltzoff 1997](#); [Premack & Woodruff 1978](#)). To do this, the hearer must know that the speaker has a mind similar to its own and must be able to estimate how the speaker came to its decision to talk about something. In order to implement some form of Theory of Mind, we assume the hearer would use its own decision mechanism (for instance the DQT) to simulate the speaker's decision process. However, instead of using its own perceptual input and internal states, it will estimate the speaker's perceptual input and internal states if possible. (If this is impossible — for instance, an agent cannot know the exact energy level of another agent — then the hearer will use its own state.) Feeding this estimated input into its own decision mechanism will result in one or more possible

hypotheses for the expression's meaning.

### 7.16

When the agents have identified one or more possible meanings to go with a novel word using one or more of the above principles, the new association is added to the lexicon. Although it is still likely that the agent will not be able to determine the meaning precisely, we believe that if the number of possible meanings is sufficiently reduced and the correct meaning is among the meanings associated, the agent will be able to learn the proper meaning in future language games using the *a posteriori* probabilities and association scores.

### 7.17

In this section, we have only focused on the emergence of a simple vocabulary-like language. However, we intend to extend this model to include the emergence of a simple grammar based on the model introduced in Vogt (2005). In this model, agents are capable of discovering regularities in both the conceptual space and the expression space, which are then used to form simple grammatical rules that allow them to combine different concepts to form complex sentences.

### 7.18

Initially we intend to hardwire all mechanisms controlling the language game — except of course the language itself. Once we have implemented a model that can facilitate language emergence, we will allow the population to evolve some of these properties genetically. For instance, we might remove the four above mentioned principles and include genes that control the likelihood with which an agent will incorporate one or more of them into its language learning. We can then investigate empirically if population learning will favour the evolution of Theory of Mind or perhaps some other principle.



## The peer-to-peer infrastructure

### 8.1

The size of the simulated world that can be handled efficiently by a single computer is restricted by the computational resources available on that computer. By combining several computers in a network, the possible size of the world can be increased. In the NewTies project, we use a peer-to-peer (P2P) network as the underlying architecture for the simulation.

### 8.2

A grid landscape is both straightforward and efficient to implement on a single computer. In a distributed simulation this is not necessarily the case. Since a (pure) peer-to-peer network cannot have a central server, it has to partition the landscape so that it can be handled collectively by the nodes in the network. Because the peer nodes may differ in local configuration, processing speed, network bandwidth, and storage capacity, the size of the partitions can vary greatly and can even change over time when new nodes become available and other nodes disappear. In this section, we discuss how using a peer-to-peer network necessitates partitioning of the landscape over the nodes of the network and how this relates to the scalability of the network, the consistency of the information stored in the network, and the efficiency of the simulation. Two basic agent actions are identified as central to these issues, the move action and the look action, and we consider how a 'move-buffer' and a 'look-index' can be used to solve the problems generated by these actions.

### Peer-to-peer networks

### 8.3

A peer-to-peer network is one that does not rely on dedicated servers for communication but instead uses direct communication between clients (peers). A pure peer-to-peer network does not have the notion of clients or servers, but only peer nodes, all with equivalent functionality. There are two relevant properties of a peer-to-peer network: the bandwidth of all peers can be fully used and the network is scalable. This means that when the number of nodes in the network increases, the total available bandwidth also increases. In a client-server network, all clients have to share the (limited) bandwidth of the server, so that having a larger number of clients entails slower data transfer.

## 8.4

Although peer-to-peer networks are well-suited to handle large scale distributed simulations, they also pose some problems. For one, all distributed simulations must assume at least some level of unreliability in the availability of the resources in the network. Although in modern computers and computer networks the failure-rate is small, if the number of computers in the network is large enough or the duration of the simulation long enough, some resource- or information-loss has to be expected. This is summarised as the *unreliability assumption*.

## 8.5

The efficiency of an agent simulation implemented on a peer-to-peer network depends to a large extent on the efficiency with which the underlying network can supply information. Environment information is requested frequently and queries for landscape information have to be handled efficiently. Query efficiency is measured by the response-time, that is, the time it takes between an issue of a query and the return of the requested information. In order to reduce query response-times, peer-to-peer networks use information indexing, a technique borrowed from database management systems ([Aberer et al. 2002](#); [Dabek et al. 2001](#)). Indexing implies the generation and maintenance of redundant (meta) information, in order to more quickly locate pieces of information stored in the system. A balance has to be struck between the cost of maintaining the index and the reduction in response time it allows.

## 8.6

In order to increase the scalability of the system, information about the landscape should be maintained as sparsely as possible. Landscape locations should be created only when needed and only locations that are referenced should be preserved. However, landscape information inconsistencies can then occur. This is because it is possible to create a location on one node that has already been created on another. The underlying peer-to-peer network has to be able to handle this kind of inconsistency. However, it has to first be aware that the inconsistency exists, hence the need to propagate information about location creation throughout the network. An *inconsistency assumption* can be formulated, that a certain level of inconsistency of information is inevitable in a distributed system. In some cases, the simulation itself is robust to some level of inconsistency, reducing the need to handle them quickly. In other cases, inconsistent information can have serious consequences, placing more stringent requirements on the network. Dealing with inconsistencies incurs some overhead and a balance between efficiency and ability to reduce inconsistency has to be found.

### Managing the move and look actions

## 8.7

The delicate balance that has to be struck between scalability, consistency and efficiency is most clearly apparent for two of the actions that an agent can take: move and look. When an agent moves to a new location — one that has not yet been created — the simulation has to create one for it. When two nodes in the peer-to-peer network need to create the same location at the same time, this can lead to an inconsistency. We solve this problem through the use of a buffer-zone of locations around agents and other movable objects.

## 8.8

The look-action provides the agent with the perceptual information it needs to decide on the actions it is going to undertake. The number of locations visible to an agent is determined by the `MaxVisionDistance` parameter and the heading of the agent. Together, they define a *look-arc* of locations visible to the agent. Since agents look often and sometimes can look far, the action can produce a large number of queries.

## 8.9

A query in a peer-to-peer network is handled by propagating it through the network. Each node with relevant information then participates in resolving the query. A query that cannot be resolved takes a long time to fail, since it has to propagate throughout the whole network. So that the queries resulting from a look-action can be handled efficiently by the system without adversely affecting its scalability, a technique involving predictive indexing of locations has been designed.



## The move-buffer

### 8.10

The move-buffer is a buffer-zone around the locations in the landscape that are or have been used by the agents. Instead of creating locations at the moment when an agent wants to move to them, we create a buffer-zone around all the locations that the agents or other movable objects have used before or are using now. If an agent wants to move into the buffer-zone, the locations are already created and the information about these locations is consistent throughout the network. For example, when two agents on two landscape islands, handled by two nodes in the network, start moving toward each other, the buffer-zone is repeatedly extended in front of each agent at each time step. At some point, the buffers of both nodes will include some of the same locations. At this point, an inconsistency can occur when two nodes try to create the same location at the same time. This can be resolved by choosing one node to handle the location and copying over information to the other node. However, while the inconsistency is being resolved, the agents may keep moving. The buffer-zone has therefore to cover sufficient locations around the agents that the network has time to handle any inconsistencies before the newly-created locations in the buffer-zone are needed by the simulation.

## The look-index

### 8.11

With the look action, the efficiency of the system is the more important issue. As the landscape is stored as sparsely as possible, there will be (possibly large) portions of the landscape that have not yet been created. The agents in the simulation, however, might still want to look at these locations, and since this landscape information is unavailable in the network, we must assume that a (possibly large) number of queries will fail. As a result, the efficiency of the simulation will be adversely affected.

### 8.12

To avoid this, information about the landscape is indexed in a *look-index*. If a large enough portion of the landscape is indexed, all the information needed for a query should be available in the index. The index is generated by adding either an empty entry to the index for locations that have not yet been created or used, or the address of the node where the location is handled. Instead of propagating information about the location, we propagate only the newly created index entries through the network. The node in the network that handles the location then sends back its address to the node that sent the index entry. When an agent does a look action, the index is first inspected. If the location corresponds to an empty entry in the index, the location has not yet been created and no query is issued. If the location corresponds to a non-empty index entry, a direct network connection can be set up to retrieve the information about the location efficiently. The look-index is maintained so that the information that a query can request is complete, in order to make sure that no query can fail. The size of the look-index is a trade-off between the efficiency of the simulation and its scalability. In our implementation, the look-index extends a few locations further than the distance that the agents can look. The extra extent beyond the look-distance allows some time to propagate the index entries through the network and so that when two entries about the same location are created in separate parts of the network, it can ensure that the two index entries do not contain conflicting information.

### 8.13

To summarise, when a social simulation is distributed over a peer-to-peer distributed computing infrastructure, two problems have to be addressed: that distributed infrastructures are inherently unreliable and that, because concurrent behaviour is necessary to maintain efficiency, inconsistency of information has to be assumed. Dealing with unreliability and inconsistency has to keep both efficiency and scalability in mind. We deal with these problems by implementing a move-buffer and a look-index. The move-buffer reduces inconsistencies created by a moving agent, and the look-index increases efficiency of a perceiving agent. Both methods reduce unreliability by caching information.

## Conclusion

### 9.1

In the preceding sections, we have outlined a design for a simulation which can be tuned in ways that are expected to promote the emergence of agent social behaviour to solve environmental challenges analogous to those that human societies have been able to overcome.

### 9.2

If such behaviour does arise, the simulation could serve as an invaluable test bed for examining a wide range of social theories. Its great advantage is that while one cannot experiment on human societies, one can on artificial societies. It will be possible, for example, to determine the conditions under which particular social phenomena emerge and survive in a way undreamt of by social theorists who can observe only a small number of human societies as cases on which to test their ideas. Even these few societies have been subject to an unknown amount of cross-fertilisation (for example, it is believed that the practice of agriculture was only discovered in two or three places in the world's history; all other agriculture was learned by copying these early innovations ([Smith 1995](#))).

### 9.3

Nevertheless, there must be some caveats about making too close a link between the simulation and human societies. On the one hand, the simulated agents are lacking many of the qualities of humans, and we do not know to what extent the differences between humans and the agents are important for the generation of analogous social phenomena (for example, we noted above that the simulation does not treat 'warmth' as a distinct need for the agents, although in cold climates it is for humans).

### 9.4

On the other hand, what we observe in human societies is one outcome from an unknown number of other possibilities. For example, although most simple societies engage in some form of trade with other communities, the Kula Ring is unique. No other society has ever been discovered in which there is a two-way flow of symbolic goods. It follows that if the agent society does not generate an institution resembling the Kula Ring, this may simply be because an alternative institution has evolved, as it did in the great majority of human societies faced with similar challenges. This is of course a question that can be explored using the simulation: the experiment can be repeated many times to see whether a Kula phenomenon ever appears.

---

## Acknowledgements

The NewTies project is partially supported by the European Commission Framework 6 Future and Emerging Technologies under contract 003752. Opinions are the authors' own and do not represent those of the Commission. The ideas in this paper have greatly benefited from interaction with the partners in the project (Vereniging voor Christelijk Wetenschappelijk Onderwijs; University of Surrey, Eötvös Loránd University; Napier University; and Stichting Katholieke Universiteit Brabant).

---

## Notes

<sup>1</sup>New and Emergent World models Through Individual, Evolutionary, and Social learning (NEW TIES), <http://www.newties.org>

<sup>2</sup>This motivation does *not* have to be 'hard-coded' or built in; it too can be evolved, since agents that have no 'motivation' to survive are also likely to fail to reproduce. Hence there will be evolutionary pressure in favour of agents that behave in ways that make them appear to be motivated for survival.

<sup>3</sup>We call it the 'NewTies' agent to distinguish this agent design from others that may be introduced into the environment. As noted later, the intention is to make the environment open

so that other agent designs may be trialled.

---

## References

- ABERER, Karl, Hauswirth, Manfred, & Puceva, Magdalena. (2002). Improving data access in P2P systems. *IEEE Internet Computing*, 6(1), 58–67.
- BANZHAF, W., Nordin, P., Keller, R.E., & Francone, F.D. (1998). *Genetic Programming: An Introduction*. San Francisco: Morgan Kaufmann.
- BLOOM, P. (2000). *How children learn the meanings of words*. Cambridge, MA: MIT Press.
- CANGELOSI, A. and D. Parisi (2002). *Simulating the evolution of language*. London, Springer.
- CHOUINARD, M. M., & Clark, E. V. (2003). Adult reformulations of child errors as negative evidence. *Journal of Child Language*, 30(3), 637–669.
- CLARK, E. V. (1993). *The lexicon in acquisition*: Cambridge University Press.
- CLARK, H. H. (1996). *Using Language*: Cambridge University Press.
- DABEK, F, Brunskill, E, Kaashoek, F, Karger, D, Morris, R, Stoica, I, et al. (2001). *Building peer-to-peer systems with CHORD, a distributed lookup service*. Paper presented at the Proceedings of the 8th Workshop on Hot Topics in Operating Systems (HotOS VIII).
- GILBERT, Nigel. (2005). Agent-based social simulation: dealing with complexity. from <http://www.complexityscience.org/NoE/ABSS-dealing%20with%20complexity-1-1.pdf>
- GILBERT, Nigel, & Troitzsch, Klaus G. (2005). *Simulation for the social scientist* (Second ed.). Milton Keynes: Open University Press.
- GONG, T., Ke, J., Minett, J. W., & Wang, W. S. Y. (2004). *A computational framework to simulate the co-evolution of language and social structure*. Paper presented at the Artificial Life IX Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems.
- GOPNIK, A., & Meltzoff, A. N. (1997). *Words, Thoughts, and Theories (Learning, Development, and Conceptual Change)*. Cambridge, MA.: MIT Press.
- HAYKIN, S. (1999). *Neural networks: {A} comprehensive foundation* (2nd edition ed.). Upper Saddle River, NJ: Prentice-Hall.
- HOLLAND, John, H., Booker, Lashon, B., Colombetti, Marco, Dorigo, Marco, Goldberg, David, E., Forrest, Stephanie, et al. (2000). *What Is a Learning Classifier System?* Springer-Verlag.
- HUTCHINS, E, & Hazlehurst, B. (1995). How to invent a lexicon: the development of shared symbols in interaction. In G. N. Gilbert & R. Conte (Eds.), *Artificial Societies*. London: UCL Press.
- KING, Leslie J. (1985). *Central Place Theory*: Sage.
- KOVACS, Tim. (2002). *Two Views of Classifier Systems*. Paper presented at the Fourth International Workshop on Learning Classifier Systems – IWLCS-2001, San Francisco, California, USA.
- MALINOWSKI, B. (1922 [1978]). *Argonauts of the Western Pacific*. London: Routledge and Kegan Paul.
- MARKMAN, E. (1989). *Categorization and naming in children*. Cambridge, Ma.: MIT Press.
- PREMACK, D., & Woodruff, G. (1978). Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4), 515–526.
- QUINE, W. V. O. (1960). *Word and object*: Cambridge University Press.

SMITH, Andrew, D. M. (2003). Intelligent Meaning Creation in A Clumpy World Helps Communication. *Artificial Life*, 9(2), 559--574.

SMITH, Andrew. D. M. (2005). *Mutual Exclusivity: Communicative Success Despite Conceptual Divergence*. Paper presented at the Language Origins: perspectives on evolution, Oxford.

SMITH, Bruce. (1995). *The Emergence of Agriculture*. New York: Scientific American Library.

STEELS, L. (1996). *Emergent adaptive lexicons*. Paper presented at the From Animals to Animats 4, Cambridge, MA.

STEELS, L. (1997). *Language Learning and Language Contact*. Paper presented at the Workshop Notes of the ECML/MLnet Familiarization Workshop on Empirical Learning of Natural Language Processing Tasks, Prague.

STEELS, L., & Kaplan, F. (1999). *Situated Grounded Word Semantics*. Paper presented at the Proceedings of IJCAI 99.

SUTTON, R., & Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Cambridge: MIT Press.

TOMASELLO, M. (1999). *The cultural origins of human cognition*: Harvard University Press.

VOGT, P. (2003). Anchoring of semiotic symbols. *Robotics and Autonomous Systems*, 43(2), 109--120.

VOGT, P. (2005). The emergence of compositional structures in perceptually grounded language games. *Artificial Intelligence* 167(1-2):206 - 242.

VOGT, P., & Divina, F. (2005). *Language evolution in large populations of autonomous agents: issues in scaling*. Paper presented at the Proceedings of AISB 2005, Hatfield, UK.

VOGT, P., & Coumans, H. (2003). Investigating social interaction strategies for bootstrapping lexicon development. *Journal for Artificial Societies and Social Simulation*, 6(1) <http://jasss.soc.surrey.ac.uk/6/1/4.html>.

VOGT, P., & Smith, A. D. M. (2005). Learning colour words is slow: a cross-situational learning account. *Behavioral and Brain Sciences* 28(4): 509-510.

WOOLDRIDGE, M., & Jennings, N. R. (1995). Intelligent Agents: Theory and Practice. In *Knowledge Engineering Review*, 10(2), 1-62.

---

[Return to Contents of this issue](#)

© [Copyright Journal of Artificial Societies and Social Simulation, \[2006\]](#)

