The Origins of Ontologies and Communication Conventions in Multi-Agent Systems

Luc Steels
Sony Computer Science Laboratory Paris
and
Artificial Intelligence Laboratory
Vrije Universiteit Brussel
Pleinlaan 2, B-1050 Brussels, Belgium
E-mail: steels@arti.vub.ac.be

November 19, 1997

Abstract

The paper proposes a complex adaptive systems approach to the formation of an ontology and a shared lexicon in a group of distributed agents with only local interactions and no central control authority. The underlying mechanisms are explained in some detail and results of some experiments with robotic agents are briefly reported.

Keywords: origins of language, self-organization, distributed agents, open systems.

1 Introduction

Agents cooperating in a multi-agent setting need a shared set of conventions [13]. The question addressed in this paper is where these conventions might come from. One approach is to agree upon a set of conventions and hence a

particular domain ontology in advance, and embed them in all future agent communication protocols. This is the approach underlying the standardisation efforts associated with Ontolingua [3] and KQML [1]. There are several reasons however why this may not be the best way to proceed.

1. It is hard to imagine how there could ever be a world-wide consensus about the ontologies and associated languages for every possible domain of multi-agent application.

2. Multi-agent systems are typically open systems. This means that the conventions cannot be defined once and for all but are expected to expand as new needs arise.

3. Multi-agent systems are typically distributed systems. There is no central control point. This raises the issue how evolving communication conventions might spread to agents which are already operational.

This paper explores an alternative to top-down design and global enforcement, namely self-organised emergence. I propose a mechanism by which a group of agents develop a shared lexicon for communicating a description, a mechanism by which agents develop their own ontology grounded in perception (but possibly grounded in other domains, e.g. social relations), and a co-evolutionary coupling so that the ontology and the language are tightly coordinated.

The main features of the proposed approach are:

1. There is no central controlling agency. Coherence arises in a bottom up, self-organised fashion.

2. The language community is open. New agents may enter at any time. They progressively adopt the conventions of the group and the group adopts new conventions that might be introduced by the new agent.

3. Conventions are adaptive. New meanings may enter at any time and the group develops the appropriate lexicalisations as needed.

4. The ontologies remain adaptive. New stimuli from the environment may require the formation of new distinctions.

These features are achieved without giving up the basic principles of an (autonomous) agent approach:

1. The agents have only limited knowledge. They cannot inspect the internal states of other agents.

2. The agents engage only in local interactions with other agents. No agent has a complete overview of what is happening.

3. The agents are autonomous. They acquire their own knowledge and decide for themselves how to communicate or divide up their world.

4. There is no global synchronisation. The system can operate in a fully distributed parallel fashion.

The proposed principles have been implemented in software simulations [6], [7], [8] and have been integrated in robotic agents, in which case the ontology is based on an embodied physical interaction with the environment [12], [9]. This research is strongly related to a growing body of work on the origins of (natural) languages, extensively reviewed in [10].

The rest of the paper is intended as a survey paper of our experiments with more details available in the cited papers. The basic components are presented in the next section (section 2). Section 3 focuses on ontology creation. Section 4 focuses on lexicon formation. Section 5 shows results of experiments in grounding.

2 Co-evolution of words and meanings

An interaction between two agents can profitably be modeled as a game. When the interaction involves language, it is a *language game*. A language game evolves a speaker and a hearer. The games that we will study by way of example, assume that the speaker wants to identify an object to the hearer given a particular context of other objects. In order to do this, the speaker must conceptualise the objects so as to find a description which distinguishes the topic from the other objects in the context. This requires an ontology, i.e. a set of distinctions. Then the speaker must find words to encode the distinctive features thus found, and transmit these words to the hearer. Next, the hearer receives the transmitted message, decodes it into one or more possible interpretations, and checks whether the interpretations are compatible with the present situation. The game succeeds if this is the

case. Failure may be due to (1) missing categories, or (2) missing or wrong linguistic conventions. In each case the agent can engage in a repair action which consists in extending his ontology, extending his lexicon (by creating a new word or by acquiring a word used by the speaker), or revising the lexicon. Agents record the use and success of words and prefer words that were used the most and had the most success in use. This causes coherence to emerge because the probability that a word is used increases if more agents adopt it.

The coordination of ontology creation and lexicon formation in a single agents and in a multi-agent system happens by co-evolution. There is an information flow and selectionist pressure in both directions. The categorisation produces ontologies which are lexicalised. Lexicalisation is successful if the word is also used by other agents. Feedback is established from the lexicon to the ontology if the agents prefer distinctions that have been successfully lexicalised, because then those categories will 'survive' that have become part of the language. This causes convergence of the ontology without a central control agency.

The coming sections contain more details of these various mechanisms followed by results from computational and robotic experiments showing that indeed a common lexicon and an ontology grounded in perceptual experiences emerges.

3 Ontology creation through discrimination games

Let there be a set of objects $\mathcal{O} = \{o_1, ..., o_m\}$ and a set of sensory channels $S = \{\sigma_1, ..., \sigma_n\}$, being real-valued partial functions over $\mathcal{O}$. Each function σ_j defines a value $0.0 \leq \sigma_j(o_i) \leq 1.0$ for each object o_i.

An agent a has a set of feature detectors $D_a = \{d_{a,1}, ..., d_{a,m}\}$. A *feature detector* $d_{a,k} = \langle p_{a,k}, V_{a,k}, \phi_{a,k}, \sigma_j \rangle$ has an attribute name $p_{a,k}$, a set of possible values $V_{a,k}$, a partial function $\phi_{a,k}$, and a sensory channel σ_j. The result of applying a feature detector $d_{a,k}$ to an object o_i is a feature written as a pair $(p_{a,k}\ v)$ where p is the attribute name and $v = \phi_{a,k}(\sigma_j(o_i)) \in V_{a,k}$ the value.

The *feature set* of a for o_i is defined as $F_{a,o} = \{(p_{a,k}\ v) \mid d_{a,k} \in D_a, d_{a,k} = \langle p_{a,k}, V_{a,k}, \phi_{a,k}, \sigma_j \rangle, v = \phi_{a,k}(\sigma_j(o_i))\}$. Two features $(a_1\ v_1)$, $(a_2\ v_2)$ are *distinctive* iff $a_1 = a_2$ *and* $v_1 \neq v_2$. A distinctive feature set $D^C_{a,o}$ is a set of

features distinguishing an object o_t from a set of other objects C. $D^C_{a,o} = \{f \mid f = (p\ v) \in F_{a,o}\ \textit{and}\ \forall o_c \in C\ \textit{either}\ / \exists f' = (p'\ v') \in F_{a,o}\ \textit{with}\ p = p'\ \textit{or}\ \exists f' \in F_{a,o}\ \textit{with}\ f\ \textit{and}\ f'\ \textit{distinctive}\}$. Clearly there can be several distinctive feature sets for the same o_t and C, or none.

A discrimination game $d =< a, o_t, C >$ involves an agent a, a topic $o_t \in C \subseteq \mathcal{O}$. C is called the context. The outcome of the game is twofold. Either a distinctive feature set could be found, $D^C_{a,o} \neq \emptyset$, and the game ends in success, or no such feature set could be found, $D^C_{a,o} = \emptyset$, and the game ends in failure.

As part of each game the repertoire of meanings is adjusted in the following way by the agent:

1. $D^C_{a,o} = \emptyset$, i.e. the game is unsuccessful. This implies that there are not enough distinctions and therefore $\exists o_c \in C,\ F_{a,o} \subseteq F_{a,o}$. There are two ways to remedy the situation:

   (a) If there are still sensory channels for which there are no feature detectors, a new feature detector may be constructed. This option is preferred.

   (b) Otherwise, an existing attribute may be refined by creating a new feature detector that further segments the region covered by one of the existing attributes.

2. $D^C_{a,o} \neq \emptyset$. In case there is more than one possibility, feature sets are ordered based on preference criteria. The 'best' feature set is chosen and used as outcome of the discrimination game. The record of use of the features which form part of the chosen set is augmented. The criteria are as follows:

   (a) The smallest set is preferred. Thus the least number of features are used.

   (b) In case of equal size, it is the set in which the features imply the smallest number of segmentations. Thus the most abstract features are chosen.

   (c) In case of equal depth of segmentation, it is the set of which the features have been used the most. This ensures that a minimal set of features develops.

The whole system is selectionist. Failure to discriminate creates pressure to create new feature detectors. However the new feature detector is not guaranteed to do the job. It will be tried later and only thrive in the population of feature detectors if it is indeed successful in performing discriminations.

The discrimination game defined above has been implemented and encapsulated as an agent. The programs create a set of sensory channels and an initial set of objects which have arbitrary values for some of the sensory channels. A typical example is the following list of objects and associated values for channels:

```
o-0: [sc-3:0.73] [sc-4:0.82] [sc-5:0.07]
o-1: [sc-0:0.89] [sc-3:0.02]
       [sc-4:0.56] [sc-6:0.48]
o-2: [sc-0:0.74] [sc-1:0.92] [sc-2:0.22]
       [sc-3:0.56] [sc-8:0.52] [sc-9:0.03]
o-3: [sc-2:0.36] [sc-3:0.09] [sc-4:0.14]
o-4: [sc-1:0.47] [sc-2:0.61] [sc-3:0.69]
       [sc-5:0.67] [sc-6:0.14] [sc-9:0.43]
...
```

A feature detector is a function assigning a value to a certain attribute. The name of the attribute indicates its nature. It is of the form $sc_i - n_1 -$ ... where i is the sensory channel followed by which one of the two segments has been chosen. For example, sc-5 is the name of an attribute whose feature detector operates on sc-5. sc-5-1 is a feature that identifies the second segment of sc-5. sc-5-1-0 identifies the first segment of the second segment of sc-5, etc. (sc-5-1-0 v-0) is a feature combining this attribute with the value v-0.

In normal operation, the agent continuously goes through a loop performing the following activities:

1. A context is delineated. The context consists of the objects currently in the field of attention of the agent.

2. One object in this context is chosen randomly as topic.

3. The feature sets of the topic and the other objects in the context are derived.

4. An attempt is made to find possible discriminating feature sets.

We now show some typical situations for an agent a-5, which starts from no features at all. In the first game, a-5 tries to differentiate the object o-5 from o-3. The agent does not have a way yet to characterise the topic and creates a new attribute operating on sc-5.

```
a-5: o-5 <-> {o-3 }
Topic: NIL
Not enough features topic
New attribute: sc-5
```

The next game to distinguish o-5 from o-9 and o-1 is already successful, because o-5 is again the topic. The context contains objects that do not have any response for sc-5, and thus no features can be constructed:

```
a-5: o-5 <-> {o-9 o-1 }
Topic: ((sc-5 v-1))
Context: (NIL NIL)
Success: ((sc-5 v-1))
```

The next game is also sucessful because o-6 has value v-0 for sc-5, o-2 has nothing and o-5 has v-1.

```
a-5: o-6 <-> {o-2 o-5 }
Topic: ((sc-5 v-0))
Context: (NIL ((sc-5 v-1)))
Success: ((sc-5 v-0))
```

In the following game the attributes are not sufficiently distinctive and therefore a new attribute is created. As long as there are possibilities to focus on additional sensory channels, existing attributes will not be refined. The new attribute operates on sc-3.

```
a-5: o-7 <-> {o-1 o-2 }
Topic: ((sc-1 v-1))
Context: (NIL ((sc-1 v-1)))
No distinctive features but new
   one possible: (sc-2 sc-3 sc-8)
New attribute: sc-3
```

When uncovered sensory channels are no longer available, more refined feature detectors for existing attributes start to be made. In the following example, o-0 fails to be distinguished from o-8 and 0-1, even though a set of features is available to characterise each object. A refinement of the attribute operating over sc-5 is chosen.

```
a-5: o-0 <-> {o-8 o-1 }
Topic: ((sc-3 v-1)(sc-4 v-1)(sc-5 v-0))
Context: (((sc-0 v-1)(sc-1 v-0)(sc-3 v-1)
           (sc-4 v-0)(sc-5 v-0)))
          ((sc-0 v-1)(sc-3 v-0)(sc-4 v-1)))
No distinctive features but refinements possible.
Refining attribute: sc-5 => sc-5-0, sc-5-1
```

After a sufficient number of discrimination games the set of features stabilises. For the set of objects given above, the following is a stable discrimination tree. For each attributes the possible values are listed with their corresponding regions as well as the number of times a feature has been used.

```
sc-5:
  v-0: [0.00 0.50] 358.
   sc-5-0:
     v-0: [0.00 0.25] 31.
      sc-5-0-0:
       v-0: [0.00 0.12]
        sc-5-0-0-0:
         v-0: [0.00 0.06] ; v-1: [0.06 0.12] 3.
       v-1: [0.12 0.25]
     v-1: [0.25 0.50] 22.
  v-1: [0.50 1.00] 309.
sc-1:
  v-0: [0.00 0.50] 651. ;  v-1: [0.50 1.00] 628.
sc-3:
  v-0: [0.00 0.50] 713. ; v-1: [0.50 1.00] 733.
sc-8:
  v-0: [0.00 0.50] 15. ; v-1: [0.50 1.00] 8.
sc-2:
  v-0: [0.00 0.50] 99. ; v-1: [0.50 1.00] 112.
```

```
sc-0:
  v-0: [0.00 0.50] ;  v-1: [0.50 1.00] 42.
sc-4:
  v-0: [0.00 0.50] 223.
   sc-4-0-0:
     v-0: [0.00 0.25]
     v-1: [0.25 0.50] 1.
      sc-4-0-0-1:
        v-0: [0.25 0.37] 5.;  v-1: [0.37 0.50] 5.
  v-1: [0.50 1.00] 215.
   sc-4-1:
     v-0: [0.50 0.75] 1.
     v-1: [0.75 1.00] 2.
      sc-4-1-1:
        v-0: [0.75 0.87] 5. ; v-1: [0.87 1.00] 2.
sc-6:
  v-0: [0.00 0.50] 2. ; v-1: [0.50 1.00]
```

We see that more abstract features, like *(sc-1 v-0)*, are used more often. For some, like *(sc-5 v-0)*, there is a deep further discrimination. For others, like *(sc-5 v-1)*, there is none. Some features, like *(sc-6 v-1)*, have not been used at all and could therefore be eliminated. Another experiment with the same objects but for a different agent a-6 yields a different discrimination tree. In one example, some sensory channels (such as sc-6) were not used, sc-4 was no longer refined, etc. Usually there are indeed many different possibilities and an important question for further study is how optimal the discrimination trees obtained with the proposed mechanism are.

When new objects enter the environment, the agent should construct new distinctions if they are necessary. This is effectively what happens. If new sensory channels become available, for example because a new sensory routine has become active, then it will be exploited if the need arises.

Fig 1. shows a typical example where an agent builds up a repertoire of feature detectors, starting from scratch. We start from a set of 10 objects and gradually add new objects in a probabilistic fashion, to reach a total of 50 objects. We see that the feature repertoire is extended occasionally. The average discrimination success remains close to the maximum (1.0) because new objects are only encountered occasionally and the feature detectors al-

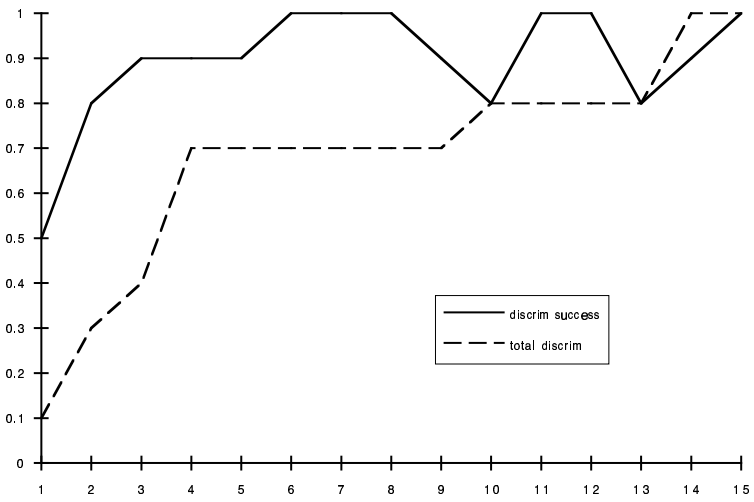

Figure 1: The graph shows the evolution of the discriminatory capacities of a single agent. The total number of objects (10) is fixed. There are 5 sensory channels. The average success in discrimination games as well as the global success is shown on the y-axis. The number of discrimination is mapped on the x-axis (scale 1/10). All objects can be discriminated after 150 discrimination games.

ready constructed are general. Fig 2. shows how the system copes with new objects.

When performing multi-agent experiments, each of the agents is running the same ontology creation mechanisms. Even if they are in the same environment they will end up with different ontologies. Similarities are uniquely due to the fact that the agents share the same context. The coupling to lexicon formation discussed in the next section pushes the ontologies towards greater coherence because it is a collective activity with feedback between words and meanings.

4 Lexicon formation

Let a word be a sequence of letters drawn from a finite shared alphabet. An expression is a set of words. A lexicon L is a relation between feature sets and words. A single word can have several associated feature sets and a given

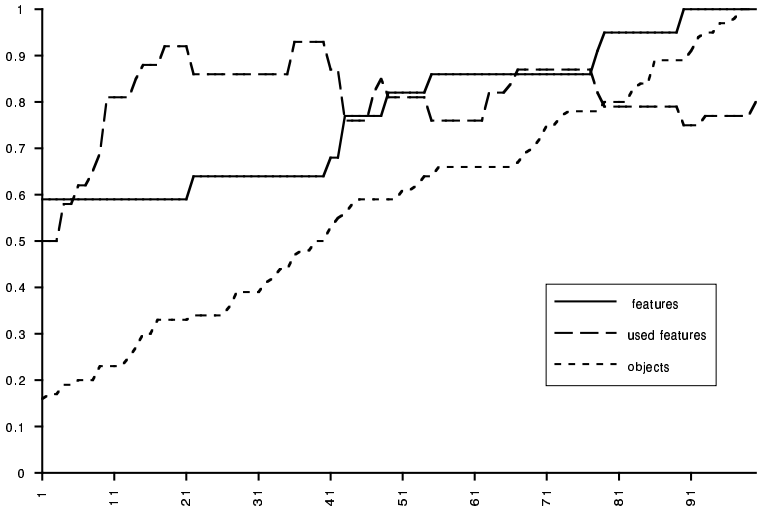

Figure 2: Graph showing how an agent handles a steady increase in the number of objects. The graph shows on the y-axis the number of objects (as a percentage of the total reached at the end, i.e. 50), the discriminatory success which remains close to the maximum, and the number of features (as a percentage of the total reached at the end, i.e. 35). The x-axis plots the number of discrimination games (scale 1/10).

feature set can have several associated words. For each word-meaning pair, the use and success in use is recorded. Words that are used more and have more success are prefered. This establishes a positive feedback loop pushing the group towards coherence.

Each agent $a \in A$ is assumed to have a single lexicon L_a which is initially empty. A feature set of a word in L is denoted as $F_{w,L}$. The following functions can be defined:

- $cover(F, L)$ defines a set of expressions, where each expression U is such that $\forall f \in F, \exists w \; such \; that \; f \in F_{w,L} \; and \; f \in U$
- $uncover(u, L)$ defines a set of feature sets, where each feature set K is such that $\forall u \in U, \exists f \; such \; that \; f \in F_{w,L} \; and \; f \subset K$

A language game involves a dialog between two agents, a speaker and a hearer, within a particular contextual setting which consists of objects of which one is chosen as the topic. The agents perceive these objects through the observational channels and construct features through the discrimination trees discussed earlier.

The scenario for a language game is as follows:

1. A speaker and hearer as well as a context consisting of a set of objects is randomly identified.
2. The speaker selects one object as the topic and points to this object so that the hearer shares the topic.
3. Both speaker and hearer identify possible distinctive feature sets using the data for the observational channels and the discrimination trees associated with each channel.
4. If there is at least one set of discriminating features, the speaker selects such a set and translates it to words using the cover function. Words that have been used the most and have been most successful in use are prefered.
5. The hearer interprets this expression using the uncover function and compares it with his expectations, i.e. the distinctive feature sets determined in step 3.

As a side effect of such a language game, various language formation steps take place:

1. *No differentiation possible* (step 3 fails): In this case new features are created as discussed in the previous section.

2. *The speaker does not have a word* (step 4 fails): In this case at least one distinctive feature set S is detected but the speaker s has no word(s) yet to express it. The language game obviously fails. However the speaker may create a new word (with a probability typically $w_c = 0.05$) and associate it in his lexicon with S.

3. *The hearer does not have a word*: At least one distinctive feature set S is detected and the speaker s can construct an expression to express it, i.e. $\exists u \; where \; cover(S, L_s) = u$. However, the hearer does not know the word. Because the hearer has a hypothesis about possible feature sets that might be used, he is able to extend his lexicon to create associations between the word used and each possible feature set. If there is more than one possibility, the hearer cannot disambiguate the word and the ambiguity is retained in the lexicon.

4. *The speaker and the hearer know the word*: In this case there are two possible outcomes:

   (a) *The meanings are compatible with the situation*: The dialog is a success and both speaker and hearer achieve communicative success. Note that it is possible that the speaker and the hearer use different feature sets, but because the communication is a success there is no way to know this. Semantic incoherences persist until new distinctions become important and disambiguate.

   (b) *The meanings are not compatible with the situation*: The same situation as before may arise, except that the feature set uncovered by the hearer is not one of the feature sets expected to be distinctive. In this case, there is no communicative success, neither for the speaker or the hearer.

The language games described above have been implemented and encapsulated in software agents. The simulation experiments consistently show

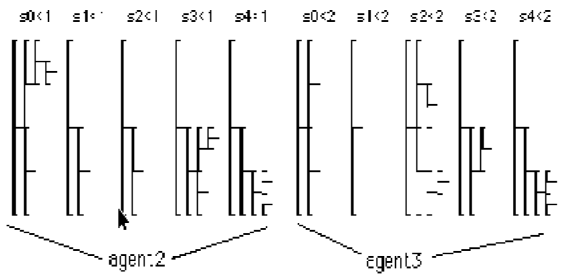

Figure 3: Evolution of the discrimination trees for a set of agents, where each agent has 4 channels. The evolution of communicative success (evolving towards 1.0) and discriminatory failure (evolving towards 0.0) is shown.

that a shared repertoire of word-meaning pairs develops conjointly with the development of a feature repertoire.

Fig 3 shows for a group of agents both the decreasing failure in discrimination (which evolves towards 0.0) and the increased success in communication (which evolves towards 1.0). Fig 4. shows the discrimination trees developed by agents operating in the same environment. Some interesting observations can be made: We see that there are on the one hand strong similarities between the agents. For example, the discrimination trees for channel 4 look almost identical. At the same time we see that there are differences. For example, for channel 2 we see that agent 3 has an elaborate discrimination tree whereas agent 2 has almost no distinctions. Also when we inspect the lexicons of the different agents we see important similarities (how else could there be complete communicative success) but at the same time we see differences. These differences are maintained because the environment allows multiple possibilities for discrimination and the same word may therefore be associated with different features without being noticed. New objects coming into the environment may disambiguate words or may cause some of the agents to develop distinctions shared by others.

Both the feature formation and language formation processes are open. The (distributed) lexicon adapts itself when new features are created. New words are created and multi-word sentences appear. The system is also open

Figure 4: The figure shows the evolution of the communicative success in a fluctuating population of agents. New agents enter with a probability 0.00005 and the depart with the same probability. The population is able to cope with the flux.

with respect to the number of agents: New agents may enter at any time in the population. The new agent will gradually take over words already present but is also a new source of novelty (see fig 4).

5 Grounding experiments

The self-organised coherence in lexicons and ontologies has been well-established in a large number of software experiments. Based on this success, we decided to see whether the mechanisms would also work on physically instantiated robotic agents. This is even more challenging because it forces us to test the robustness of the proposed mechanisms in real world settings and to see whether ontology creation can handle the rich variation present in real-world data.

Figure 5: Two robots have approached each other and are now facing each other. The robots are equiped with a dozen low-level sensors. The discrimination trees are based on output from these sensory channels. Note the other objects in the environment surrounding the robots, which will be the subject of the conversation.

5.1 Language games on mobile robots

A first experiment (reported more extensively in [12]) was conducted on fully mobile robots. The robots are small Lego-vehicles which have a variety of sensors (infrared, visible light, sound, touch, etc.), actuators for moving around in the environment, batteries, and on board processors. The robots operate in a physical ecosystem in which they have opportunities to recharge their batteries but also competitors which have to be countered by performing work [5] (fig 5).

The observational channels contain the real world data obtained from the physical sensors. An example of such data is given in fig 6. The sensors are always located on the body in pairs, for example left infrared and right infrared sensor, left and right visible light sensing, etc., so that the robot has a center of perception (as most animals). An object is in this center of

Figure 6: Sensory data streams taken from physical robot. The channels include left and right infrared and visible light sensing and motor speeds.

perception when the left and right sensory data cross over. Thus if the robot turns left towards the visible light emitted by the charging station, it will be centered on the charging station when the left visible light peak decreases and crosses the increasing right visible light peak. The sensory values at each crossing point act as input to the discrimination games.

The protocol for engaging in language games has been implemented on the physical robots by a combination of physical gestures and communications through a radio link between the robots. Two robots engage in a communication when both are facing each other. Then each robot makes a 360 degree turn to develop a panoramic sensory view of the environment. The pointing is implemented by a gesture: The speaking robot emits 4 infrared beams while moving towards the topic, so that the other robot can observe in which direction it moves. The speaking robot halts when it is facing the object that it wants to see as the topic of the conversation. The listening robot detects the topic by consulting its own sensory map. Then the language game starts as described above.

An example of a language game between two robots (r1 and r0) at the earliest stages is as follows. Three objects are encountered by r1 and 7 by r0. For each of these objects, the data are given followed by the features that have been extracted based on the discrimination trees developed so far. Although the speaker has a distinctive feature set namely { sc0-1 }, it has no words yet for it. The game therefore fails. The speaker creates a new word.

```
r1:
Objects:
- o0=[0] [1,0,0,0] -> {sc0-1}
- o1=[46] [0,2,12,3] -> {sc1-127,sc2-127,sc3-127}
- o2=[96] [0,1,0,193] -> {sc1-127,sc3-127}
r1: Topic=o0

r0:
Objects:
- o0=[0] [1,0,0,0] -> {sc0-1}
- o1=[6] [0,86,12,169] -> {sc1-127,sc2-127,sc3-127}
- o2=[7] [0,81,9,168] -> {sc1-127,sc2-127,sc3-127}
- o3=[9] [0,82,12,171] -> {sc1-127,sc2-127,sc3-127}
- o4=[20] [0,37,29,167] -> {sc1-127,sc2-127,sc3-127}
- o5=[67] [0,1,4,195] -> {sc1-127,sc2-127,sc3-127}
- o6=[72] [0,0,4,217] -> {sc2-127,sc3-127}
r0: Topic=o5

r1: DFS={sc0-1}
Encoded expression r1: (nil)
Decoded expression r0: D={}
failure
```

Another language game much further in the process (after about 1000 games) is as follows: Both speaker and hearer have a distinctive feature set (sc0-1 and sc3-190 respectively) to distinguish the topic from the other objects. The speaker uses the word "(c d)" which is recognised as compatible by the hearer with what he expects. The game succeeds.

```
r0:
Objects:
- o0=[0] [1,0,0,0] -> {sc0-1}
- o1=[7] [0,152,10,190] -> {sc1-149,sc2-10,sc3-189}
- o2=[45] [0,1,7,181] -> {sc1-1,sc2-7,sc3-186}
r0: Topic=o0

r1:
Objects:
```

Figure 7: These graphs show the average communicative success per 5 games (top) and the coherence that arises (bottom)

```
- o0=[0] [1,0,0,0] -> {sc0-1}
- o1=[6] [0,3,0,115] -> {sc1-3,sc3-115}
- o2=[19] [0,3,0,208] -> {sc1-3,sc3-209}
- o3=[36] [0,3,0,29] -> {sc1-3,sc3-30}
- o4=[118] [0,4,0,192] -> {sc1-4,sc3-190}
r1: Topic=o4

r0: DFS={sc0-1}
Encoded expression r0: (c d)
Decoded expression r1: D={sc3-190}
success
```

The graphs in fig 7 show the evolution of the shared lexicon. We have now demonstrated in a large number of experiments, that even in these very difficult circumstances coherence and successful communication emerges. The circumstances are difficult because every step in the process may fail: A robot may loose its orientation in constructing a panoramic view, the pointing may fail, the data is to some extent erratic, they may loose radio contact during the communication, etc.

5.2 The Talking Heads experiment

A second experiment in physical grounding of language formation processes is known as the *talking heads experiment*. It is reported more extensively in [10]. The experiment is based on two robotic heads which can track moving objects based on visual inputs. The heads watch a static or dynamic scene. A typical example of a scene as seen through one of the heads is contained in fig 8. Low level visual processing identifies coherent segments in the image, for example based on difference matching between consecutive images or comparisons with respect to the average intensity of the rest of the image. The observational channels now contain data extracted for each fragment, such as the area of the bounding box, the ratio of the fragment area compared

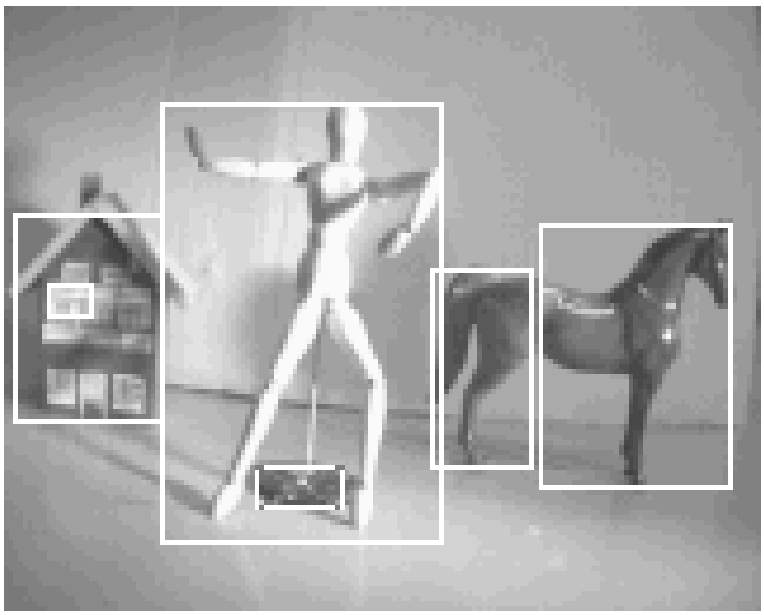

Figure 8: View through the camera of one of the heads. The consecutive bitmaps are segmented and here visualised by a bounding box around each segment.

to the bounding box area, the average light intensity within a bounding box, etc. Then the creation of a perceptually grounded ontology and of a lexicon expressing distinctive feature combinations necessary to identify an object proceeds as outlined in earlier sections.

Here are some examples of language games which imply the formation of new categories, and then of words by speaker and hearer.

```
0 ++> Speaker head-16
Failure: INSUFFICIENT-FEATURES
=> Extend categories: FILL-RATIO AVERAGE [-1.0 1.0]: v-81 v-82
Failure

1 ++> Speaker head-16
Failure: INSUFFICIENT-FEATURES
=> Extend categories: VISIBILITY SLOPE [-1.0 1.0]: v-83 v-84
Failure
```

```
...

4 ++> Speaker head-16
Failure: MISSING-LEMMA-SPEAKER
Failure: MISSING-WORD-FORM
=> Extend word repertoire: <WORD E98E44>
=> Extend Lexicon:
<SCHEMA E98EE4>
Function:(((VISIBILITY AVERAGE) v-88))
Form:((D U))
Failure

...

6 ++> Speaker head-16
Categorial Perception: (((VISIBILITY AVERAGE) v-88))
Conceptualisation:(((VISIBILITY AVERAGE) v-88))
Lemmas:(<SCHEMA E98EE4>)
Expression: ((D U))
++> Hearer head-17
Failure: INSUFFICIENT-FEATURES
=> Extend categories: VISIBILITY SLOPE [-1.0 1.0]: v-91 v-92
Failure
...
```

The first successful game happens after 47 games:

```
47 ++> Speaker head-16
Categorial Perception:
  (((FILL-RATIO AVERAGE) v-81)((VISIBILITY AVERAGE) v-109)
   ((AREA AVERAGE) v-108))
Conceptualisation:
  (((VISIBILITY AVERAGE) v-109)
   ((AREA AVERAGE) v-108)((FILL-RATIO AVERAGE) v-81))
Lemmas:(<SCHEMA E8B908>)
Expression: ((K I))
```

```
++> Hearer head-17
Categorial Perception:
  (((FILL-RATIO AVERAGE) v-125)((INTENSITY AVERAGE) v-134)
   ((AREA AVERAGE) v-132))
Expression: ((K I))
Lemmas:(<SCHEMA E8FC98>)
Meaning:(((FILL-RATIO AVERAGE) v-125)((INTENSITY AVERAGE) v-134)
         ((AREA AVERAGE) v-132))
Success
```

Here is another example of a successful game:

```
52 ++> Speaker head-16
Categorial Perception:
  (((FILL-RATIO AVERAGE) v-81)((VISIBILITY AVERAGE) v-109)
   ((AREA AVERAGE) v-108))
Conceptualisation:
   (((VISIBILITY AVERAGE) v-109)((AREA AVERAGE) v-108)
    ((FILL-RATIO AVERAGE) v-81))
Lemmas:(<SCHEMA E8B908>)
Expression: ((K I))
++> Hearer head-17
Categorial Perception:
  (((FILL-RATIO AVERAGE) v-125)((INTENSITY AVERAGE) v-134)
   ((AREA AVERAGE) v-132))
Expression: ((K I))
Lemmas:(<SCHEMA E9008C>) (<SCHEMA E9008C>)
Meaning:
  (((FILL-RATIO AVERAGE) v-125)((INTENSITY AVERAGE) v-134)
   ((AREA AVERAGE) v-132))
  (((FILL-RATIO AVERAGE) v-125)((INTENSITY AVERAGE) v-134)
   ((AREA AVERAGE) v-132))
Success
```

A snapshot of the lexicon of one agent is as follows:

```
<SCHEMA E86958>
   Function:(((VISIBILITY AVERAGE) v-88))
```

```
   Form:((D U))
<SCHEMA E86C50>
   Function:(((FILL-RATIO AVERAGE) v-82))
   Form:((T E))
<SCHEMA E872B8>
   Function:
      (((FILL-RATIO AVERAGE) v-81)((AREA AVERAGE) v-86)
       ((VISIBILITY AVERAGE) v-87))
   Form:((L E))
<SCHEMA E87628>
   Function:(((INTENSITY AVERAGE) v-90))
   Form:((N A))
<SCHEMA E87F1C>
   Function:
      (((FILL-RATIO AVERAGE) v-81)((AREA AVERAGE) v-86)
       ((INTENSITY AVERAGE) v-89))
   Form:((P U))
<SCHEMA E884C4>
   Function:(((INTENSITY AVERAGE) v-89))
   Form:((M I))
<SCHEMA E885E4>
   Function:(((FILL-RATIO AVERAGE) v-81)((AREA AVERAGE) v-108)
             ((VISIBILITY AVERAGE) v-109))
   Form:((K I))
<SCHEMA E88CF4>
   Function:(((AREA AVERAGE) v-85))
   Form:((F I))

...
```

Fig 9 shows the increased success in communication as the agents continue to build up a shared lexicon and the increase in complexity of the lexicons.

Although the physical embodiment of the Talking Heads experiment is quite different from the mobile robots, we see the same phenomena: steady increase and adaptation of a perceptually grounded ontology, and progressive build up and self-organised coherence of a shared lexicon. The Talking Heads experiment is somewhat easier because visual perception provides a

Figure 9: Graph showing the increase in average communicative success (top) as well as the increase in the number of words in the vocabularies of two robotic heads (bottom).

richer source for building an ontology and the communication and perceptual conditions are more stable.

6 Conclusions

This paper has discussed mechanisms for the creation of ontologies in the form of discrimination trees of perceptually grounded categories and the formation of a lexicon expressing a feature structure using these categories. The mechanisms exploit three principles known from biology: self-organisation, selectionism, and co-evolution. Self-organisation appears when there is a positive feedback loop between an emergent structure (in this case a shared lexicon) and future behavior. Selectionism occurs when a system generates spontaneous variation which is selected for under environmental pressure. In the present case, the spontaneous variation occurs through the (relatively) random expansion of the discrimination trees which will be positively selected for if they are relevant in future games. Co-evolution occurs when two selectionist systems are coupled in the sense that selectionist pressure flows from one to the other and vice-versa. Because agents prefer words that have shown more success in the past, the more successful words will propagate in the population. Because the success of a word feeds back to the survival of the distinctions underlying this word, a shared ontology will emerge. The sharing is always incomplete and dynamic. It is incomplete because agents may have success in communication even though they use different categories or they have different meanings for the same word which are nevertheless compatible with the situations in which they find themselves. The sharing is dynamic because new distinctions or new words may be created as required by the circumstances.

The mechanisms proposed here are generally applicable both to software agents and to robotic agents. It is sufficient to identify the observational channels, and to set up the appropriate feedbacks from the environment (for example, initially some form of pointing to establish a shared context).

Although results obtained with the presented mechanisms are very encouraging, many open issues remain. The issue of syntax and its origins has not been discussed even though some progress in this area has been made (see [10]). Syntax becomes necessary when the meaning to be conveyed is more complex and when the agents want to press more information in a

single expression and thus optimise communication and make it more reliable. It is also clear that natural languages have a much more flexible way to match meaning against a lexicon, occasionally using analogical reasoning. This implies that a flexible inference machinery is integrated in lexicon lookup. These and other issues are the subject of intense current research.

7 Acknowledgement

The software simulations discussed in this paper have been developed at the Sony Computer Science Laboratory in Paris by Luc Steels, Angus McIntyre and Frederic Kaplan. Several robot builders at the VUB AI Lab (soft and hardware) have contributed to the grounding experiments, including Andreas Birk, Tony Belpaeme, Peter Stuer and Danny Vereertbrugghe. The grounding experiments on mobile robots were carried out by Paul Vogt. Joris Van Looveren and Tony Belpaeme have been important in the physical implementation of the Talking Heads experiment.

References

[1] Arpa Knowledge Sharing Initiative. Specification of the KQML agent-communication language. External Interfaces Working group working paper, July 1993.

[2] Edelman, G.M. 1987. *Neural Darwinism: The Theory of Neuronal Group Selection.* New York: Basic Books.

[3] Genesereth, M. and S. Ketchpel (1994) Software agents. Communications of the ACM. 37(7). p. 48-53, 147.

[4] Kohonen, T. (1988) *Self-Organization and Associative Memory.* Springer Series in Information Sciences. Vol 8. Springer Verlag, Berlin.

[5] Steels, L. (1994) The Artificial Life Roots of Artificial Intelligence. *Artificial Life Journal* 1(1), pp. 89-125.

[6] Steels, L. (1996a) Emergent Adaptive Lexicons. In: Maes, P. (ed.) (1996) *From Animals to Animats 4: Proceedings of the Fourth International*

Conference On Simulation of Adaptive Behavior, The MIT Press, Cambridge Ma.

[7] Steels, L. (1996b) Perceptually grounded meaning creation. In: Tokoro, M. (ed.) (1996b) *Proceedings of the International Conference on Multi-Agent Systems.* The MIT Press, Cambridge Ma.

[8] Steels, L. (1996c) Self-organising vocabularies. In Langton, C. (ed.) *Proceedings of Artificial Life V.* Nara,1996. The MIT Press, Cambridge Ma.

[9] Steels, L. (1997a) Constructing and Sharing Perceptual Distinctions. In van Someren, M. and G. Widmer (eds.) *Proceedings of the European Conference on Machine Learning* Springer-Verlag, Berlin, 1997.

[10] Steels, L. (1997b) The origins of syntax in visually grounded robotic agents. In: Pollack, M. (ed.) *Proceedings of the 10th IJCAI, Nagoya* AAAI Press, Menlo-Park Ca. 1997. p. 1632-1641.

[11] Steels, L. (1997c) The synthetic modeling of language origins. *Evolution of Communication*, 1(1):1-35. Walter Benjamins, Amsterdam.

[12] Steels, L. and P. Vogt (1997) Grounding language games in robotic agents. In Harvey, I. et.al. (eds.) *Proceedings of ECAL 97*, Brighton UK, July 1997. The MIT Press, Cambridge Ma., 1997.

[13] Wooldridge, M. and N.R. Jennings (1995) Intelligent Agents: Theory and Practice. *Knowledge Engineering Review*, 10(2).