

# Constructivist Development of Grounded Construction Grammars

Luc Steels

University of Brussels (VUB AI Lab)

SONY Computer Science Lab - Paris

6 Rue Amyot, 75005 Paris [steels@arti.vub.ac.be](mailto:steels@arti.vub.ac.be)

## Abstract

The paper reports on progress in building computational models of a constructivist approach to language development. It introduces a formalism for construction grammars and learning strategies based on invention, abduction, and induction. Examples are drawn from experiments exercising the model in situated language games played by embodied artificial agents.

## 1 Introduction

The constructivist approach to language learning proposes that "children acquire linguistic competence (...) only gradually, beginning with more concrete linguistic structures based on particular words and morphemes, and then building up to more abstract and productive structures based on various types of linguistic categories, schemas, and constructions." (TomaselloBrooks, 1999), p. 161. The approach furthermore assumes that language development is (i) grounded in cognition because prior to (or in a co-development with language) there is an understanding and conceptualisation of scenes in terms of events, objects, roles that objects play in events, and perspectives on the event, and (ii) grounded in communication because language learning is intimately embedded in interactions with specific communicative goals. In contrast to the nativist position, defended, for example, by Pinker (Pinker, 1998), the constructivist approach does not assume that the semantic and syntactic categories as well as the linking rules (specifying for example that the agent of an action is linked to the subject of a sentence) are universal and innate. Rather, semantic and syntactic categories as well as the way they are linked is built up in a gradual developmental process, starting from quite specific 'verb-island constructions'.

Although the constructivist approach appears to explain a lot of the known empirical data about child

language acquisition, there is so far no worked out model that details how constructivist language development works concretely, i.e. what kind of computational mechanisms are implied and how they work together to achieve adult (or even child) level competence. Moreover only little work has been done so far to build computational models for handling the sort of 'construction grammars' assumed by this approach. Both challenges inform the research discussed in this paper.

## 2 Abductive Learning

In the constructivist literature, there is often the implicit assumption that grammatical development is the result of observational learning, and several research efforts are going on to operationalise this approach for acquiring grounded lexicons and grammars (see e.g. (Roy, 2001)). The agents are given pairs with a real world situation, as perceived by the sensori-motor apparatus, and a language utterance. For example, an image of a ball is shown and at the same time a stretch of speech containing the word "ball". Based on a generalisation process that uses statistical pattern recognition algorithms or neural networks, the learner then gradually extracts what is common between the various situations in which the same word or construction is used, thus progressively building a grounded lexicon and grammar of a language.

The observational learning approach has had some success in learning words for objects and acquiring simple grammatical constructions, but there seem to be two inherent limitations. First, there is the well known poverty of the stimulus argument, widely accepted in linguistics, which says that there is not enough data in the sentences normally available to the language learner to arrive at realistic lexicons and grammars, let alone learn at the same time the categorisations and conceptualisations of the world implied by the language. This has lead

many linguists to adopt the nativist position mentioned earlier. The nativist position could in principle be integrated in an observational learning framework by introducing strong biases on the generalisation process, incorporating the constraints of universal grammar, but it has been difficult to identify and operationalise enough of these constraints to do concrete experiments in realistic settings. Second, observational learning assumes that the language system (lexicon and grammar) exists as a fixed static system. However, observations of language in use shows that language users constantly align their language conventions to suit the purposes of specific conversations (ClarkBrennan, 1991). Natural languages therefore appear more to be like complex adaptive systems, similar to living systems that constantly adapt and evolve. This makes it difficult to rely exclusively on statistical generalisation. It does not capture the inherently creative nature of language use.

This paper explores an alternative approach, which assumes a much more active stance from language users based on the Peircian notion of abduction (Fann, 1970). The speaker first attempts to use constructions from his existing inventory to express whatever he wants to express. However when that fails or is judged unsatisfactory, the speaker may extend his existing repertoire by inventing new constructions. These new constructions should be such that there is a high chance that the hearer may be able to guess their meaning. The hearer also uses as much as possible constructions stored in his own inventory to make sense of what is being said. But when there are unknown constructions, or the meanings do not fit with the situation being talked about, the hearer makes an educated guess about what the meaning of the unknown language constructions could be, and adds them as new hypotheses to his own inventory. Abductive constructivist learning hence relies crucially on the fact that both agents have sufficient common ground, share the same situation, have established joint attention, and share communicative goals. Both speaker and hearer use themselves as models of the other in order to guess how the other one will interpret a sentence or why the speaker says things in a particular way.

Because both speaker and hearer are taking risks making abductive leaps, a third activity is needed, namely induction, not in the sense of statistical generalisation as in observational learning but in the

sense of Peirce (Fann, 1970): A hypothesis arrived at by making educated guesses is tested against further data coming from subsequent interactions. When a construction leads to a successful interaction, there is some evidence that this construction is (or could become) part of the set of conventions adopted by the group, and language users should therefore prefer it in the future. When the construction fails, the language user should avoid it if alternatives are available.

Implementing these visions of language learning and use is obviously an enormous challenge for computational linguistics. It requires not only cognitive and communicative grounding, but also grammar formalisms and associated parsing and production algorithms which are extremely flexible, both from the viewpoint of getting as far as possible in the interpretation or production process despite missing rules or incompatibilities in the inventories of speaker and hearer, and from the viewpoint of supporting continuous change.

### 3 Language Games

The research reported here uses a methodological approach which is quite common in Artificial Life research but still relatively novel in (computational) linguistics: Rather than attempting to develop simulations that generate natural phenomena directly, as one does when using Newton's equations to simulate the trajectory of a ball falling from a tower, we engage in computational simulations and robotic experiments that create (new) artificial phenomena that have some of the characteristics of natural phenomena and hence are seen as explaining them. Specifically, we implement artificial agents with components modeling certain cognitive operations (such as introducing a new syntactic category, computing an analogy between two events, etc.), and then see what language phenomena result if these agents exercise these components in embodied situated language games. This way we can investigate very precisely what causal factors may underly certain phenomena and can focus on certain aspects of (grounded) language use without having to face the vast full complexity of real human languages. A survey of work which follows a similar methodology is found in (CangelosiParisi, 2003).

The artificial agents used in the experiments driving our research observe real-world scenes through their cameras. The scenes consist of interactions between puppets, as shown in figure 1. These

scenes enact common events like movement of people and objects, actions such as push or pull, give or take, etc. In order to achieve the cognitive grounding assumed in constructivist language learning, the scenes are processed by a battery of relatively standard machine vision algorithms that segment objects based on color and movement, track objects in real-time, and compute a stream of low-level features indicating which objects are touching, in which direction objects are moving, etc. These low-level features are input to an event-recognition system that uses an inventory of hierarchical event structures and matches them against the data streaming in from low-level vision, similar to the systems described in (SteelsBaillie, 2003).



Figure 1: Scene enacted with puppets so that typical interactions between humans involving agency can be perceived and described.

In order to achieve the communicative grounding required for constructivist learning, agents go through scripts in which they play various language games, similar to the setups described in (Steels, 2003). These language games are deliberately quite similar to the kind of scenes and interactions used in a lot of child language research. A language game is a routinised interaction between two agents about a shared situation in the world that involves the exchange of symbols. Agents take turns playing the role of speaker and hearer and give each other feedback about the outcome of the game. In the game further used in this paper, one agent describes to another agent an event that happened in the most recently experienced scene. The game succeeds if the hearer agrees that the event being described occurred in the recent scene.

#### 4 The Lexicon

Visual processing and event recognition results in a world model in the form of a series of facts describing the scene. To play the description game, the

speaker selects one event as the topic and then seeks a series of facts which discriminate this event and its objects against the other events and objects in the context. We use a standard predicate calculus-style representation for meanings. A semantic structure consists of a set of units where each unit has a referent, which is the object or event to which the unit draws attention, and a meaning, which is a set of clauses constraining the referent. A semantic structure with one unit is for example written down as follows:

[1] unit1 | ev1 | fall(ev1,true), fall-1(ev1,obj1), ball(obj1) where unit1 is the unit, ev1 the referent, and fall(ev1,true), fall-1(ev1,obj1), ball(obj1) the meaning. The different arguments of an event are decomposed into different predicates. For example, for “John gives a book to Mary”, there would be four clauses: give(ev1,true) for the event itself, give-1(ev1, John), for the one who gives, give-2(ev1,book1), for the object given, and give-3(ev1,Mary), for the recipient. This representation is more flexible and makes it possible to add new components (like the manner of an event) at any time.

Syntactic structures mirror semantic structures. They also consist of units and the name of units are shared with semantic structures so that cross-reference between them is straightforward. The form aspects of the sentence are represented in a declarative predicate calculus style, using the units as arguments. For example, the following unit is constrained as introducing the string “fall”:

[2] unit1 | string(unit1, “fall”)

The rule formalism we have developed uses ideas from several existing formalisms, particularly unification grammars and is most similar to the Embodied Construction Grammars proposed in (BergenChang, 2003). Lexical rules link parts of semantic structure with parts of syntactic structure. All rules are reversible. When producing, the left side of a rule is matched against the semantic structure and, if there is a match, the right side is unified with the syntactic structure. Conversely when parsing, the right side is matched against the syntactic structure and the left side unified with the semantic structure. Here is a lexical entry for the word “fall”.

[3] ?unit | ?ev | fall(?ev,?state), fall-1(?ev,?obj)

⇔ ?unit | string(?unit, “fall”)

It specifies that a unit whose meaning is fall(?ev,?state), fall-1(?ev,?obj) is expressed with the string “fall”. Variables are written down with a

question mark in front. Their scope is restricted to the structure or rule in which they appear and rule application often implies the renaming of certain variables to take care of the scope constraints. Here is a lexical entry for “ball”:

[4] ?unit | ?obj | ball(?obj)  
 $\iff$  ?unit | string(?unit, “ball”)

Lexicon lookup attempts to find the minimal set of rules that covers the total semantic structure. New units may get introduced (both in the syntactic and semantic structure) if the meaning of a unit is broken down in the lexicon into more than one word. Thus, the original semantic structure in [1] results after the application of the two rules [3] and [4] in the following syntactic and semantic structures:

[5] unit1 | ev1 | fall(ev1,true), fall-1(ev1,obj1)  
 unit2 | obj1 | ball(obj1)

—  
 unit1 | string(unit1, “fall”)  
 unit2 | string(unit2, “ball”)

If this syntactic structure is rendered, it produces the utterance “fall ball”. No syntax is implied yet. In the reverse direction, the parser starts with the two units forming the syntactic structure in [5] and application of the rules produces the following semantic structure:

[6] unit1 | ?ev | fall(?ev,?state), fall-1(?ev,?obj)  
 unit2 | ?obj1 | ball(?obj1)

The semantic structure in [6] now contains variables for the referent of each unit and for the various predicate-arguments in their meanings. The interpretation process matches these variables against the facts in the world model. If a single consistent series of bindings can be found, then interpretation is successful. For example, assume that the facts in the meaning part of [1] are in the world model then matching [6] against them results in the bindings:

[7] ?ev/ev1, ?state/true, ?obj/obj1, ?obj1/obj1

When the same word or the same meaning is covered by more than one rule, a choice needs to be made. Competing rules may develop if an agent invented a new word for a particular meaning but is later confronted with another word used by somebody else for the same meaning. Every rule has a score and in production and parsing, rules with the highest score are preferred.

When the speaker performs lexicon lookup and rules were found to cover the complete semantic structure, no new rules are needed. But when some part is uncovered, the speaker should create a new

rule. We have experimented so far with a simple strategy where agents lump together the uncovered facts in a unit and create a brand new word, consisting of a randomly chosen configuration of syllables. For example, if no word for ball(obj1) exists yet to cover the semantic structure in [1], a new rule such as [4] can be constructed by the speaker and subsequently used. If there is no word at all for the whole semantic structure in [1], a single word covering the whole meaning will be created, giving the effect of holophrases.

The hearer first attempts to parse as far as possible the given sentence, and then interprets the resulting semantic structure, possibly using joint attention or other means that may help to find the intended interpretation. If this results in a unique set of bindings, the language game is deemed successful. But if there were parts of the sentence which were not covered by any rule, then the hearer can use abductive learning. The first critical step is to guess as well as possible the meaning of the unknown word(s). Thus suppose the sentence is “fall ball”, resulting in the semantic structure:

[8] unit1 | ?ev | fall(?ev,?state), fall-1(?ev,?obj)

If this structure is matched, bindings for ?ev and ?obj are found. The agent can now try to find the possible meaning of the unknown word “ball”. He can assume that this meaning must somehow help in the interpretation process. He therefore conceptualises the same way as if he would be the speaker and constructs a distinctive description that draws attention to the event in question, for example by constraining the referent of ?obj with an additional predicate. Although there are usually several ways in which obj1 differs from other objects in the context. There is a considerable chance that the predicate ball is chosen and hence ball(?obj) is abductively inferred as the meaning of “ball” resulting in a rule like [4].

Agents use induction to test whether the rules they created by invention and abduction have been adopted by the group. Every rule has a score, which is local to each agent. When the speaker or hearer has success with a particular rule, its score is increased and the score of competing rules is decreased, thus implementing lateral inhibition. When there is a failure, the score of the rule that was used is decreased. Because the agents prefer rules with the highest score, there is a positive feedback in the system. The more a word is used for a particular meaning, the more success that word will have.

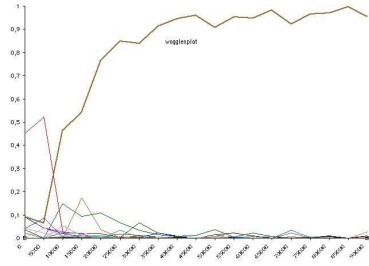


Figure 2: Winner-take-all effect in words competing for same meaning. The x-axis plots language games and the y-axis the use frequency.

Scores rise in all the agents for these words and so progressively we see a winner-take-all effect with one word dominating for the expression of a particular meaning (see figure 2). Many experiments have by now been performed showing that this kind of lateral inhibition dynamics allows a population of agents to negotiate a shared inventory of form-meaning pairs for content words (Steels, 2003).

## 5 Syntactisation

The reader may have noticed that the semantic structure in [6] resulting from parsing the sentence “fall ball”, includes two variables which will both get bound to the same object, namely  $?obj$ , introduced by the predicate  $fall-1(?ev,?obj)$ , and  $?obj1$ , introduced by the predicate  $ball(?obj1)$ . We say that in this case  $?obj$  and  $?obj1$  form an equality. Just from parsing the two words, the hearer cannot know that the object involved in the fall event is the same as the object introduced by ball. He can only figure this out when looking at the scene (i.e. the world model). In fact, if there are several balls in the scene and only one of them is falling, there is no way to know which object is intended. And even if the hearer can figure it out, it is still desirable that the speaker should provide extra-information about equalities to optimise the hearer’s interpretation efforts.

A major thesis of the present paper is that resolving equivalences between variables is the main motor for the introduction of syntax. To achieve it, the agents could, as a first approximation, use rules like the following one, to be applied after all lexical rules have been applied:

[9]  $?unit1 \mid ?ev1 \mid fall-1(?ev1,?obj2)$   
 $?unit2 \mid ?obj2 \mid ball(?obj2)$   
 $\iff$   
 $?unit1 \mid string(?unit1, "fall")$

$?unit2 \mid string(?unit2, "ball")$

This rule is formally equivalent to the lexical rules discussed earlier in the sense that it links parts of a semantic structure with parts of a syntactic structure. But now more than one unit is involved. Rule [9] will do the job, because when unifying its right side with the semantic structure (in parsing)  $?obj2$  unifies with the variables  $?obj$  (supplied by “fall”) and  $?obj1$  (supplied by “ball”) and this forces them to be equivalent. Note that  $?unit1$  in [9] only contains those parts of the original meaning that involve the variables which need to be made equal.

The above rule works but is completely specific to this case. It is an example of the ad hoc ‘verb-island’ constructions reported in an early stage of child language development. Obviously it is much more desirable to have a more general rule, which can be achieved by introducing syntactic and semantic categories. A semantic category (such as agent, perfective, countable, male) is a categorisation of a conceptual relation, which is used to constrain the semantic side of grammatical rules. A syntactic category (such as noun, verb, nominative) is a categorisation of a word or a group of words, which can be used to constrain the syntactic side of grammatical rules. A rule using categories can be formed by taking rule [9] above and turning all predicates or content words into semantic or syntactic categories.

[10]  $?unit1 \mid ?ev1 \mid semcat1(?ev1,?obj2)$   
 $?unit2 \mid ?obj2 \mid semcat2(?obj2)$

$\iff$

$?unit1 \mid syncat1(?unit1)$   
 $?unit2 \mid syncat2(?unit2)$

The agent then needs to create sem-rules to categorise a predicate as belonging to a semantic category, as in:

[11]  $?unit1 \mid ?ev1 \mid fall-1(?ev1,?obj2)$   
 $\implies$

$?unit1 \mid ?ev1 \mid semcat1(?ev1,?obj1)$

and syn-rules to categorise a word as belonging to a syntactic category, as in:

[12]  $?unit1 \mid string(?unit1, "fall")$   
 $\implies$

$?unit1 \mid ?ev1 \mid syncat1(?unit1)$

These rules have arrows going only in one direction because they are only applied in one way.<sup>1</sup> During production, the sem-rules are applied first, then the lexical rules, next the syn-rules and then the gram-

<sup>1</sup> Actually if word morphology is integrated, syn-rules need to be bi-directional, but this topic is not discussed further here due to space limitations.

mathematical rules. In parsing, the lexical rules are applied first (in reverse direction), then the syn-rules and the sem-rules, and only then the grammatical rules (in reverse direction). The complete syntactic and semantic structures for example [9] look as follows:

```
[13] unit1 | ?ev1 | fall(?ev1,?state), fall-1(?ev1,?obj),
      semcat1(?ev1,?obj)
      unit2 | ?obj1 | ball(?obj1), semcat2(?obj1)
      —
      unit1 | string(unit1, "fall"), syncat-1(unit1)
      unit2 | string(unit2, "ball"), syncat-2(unit2)
```

The right side of rule [10] matches with this syntactic structure, and if the left side of rule [10] is unified with the semantic structure in [13] the variable ?obj2 unifies with ?obj and ?obj1, thus resolving the equality before semantic interpretation (matching against the world model) starts.

How can language users develop such rules? The speaker can detect equalities that need to be resolved by re-entrance: Before rendering a sentence and communicating it to the hearer, the speaker re-parses his own sentence and interprets it against the facts in his own world model. If the resulting set of bindings contains variables that are bound to the same object after interpretation, then these equalities are candidates for the construction of a rule and new syntactic and semantic categories are made as a side effect. Note how the speaker uses himself as a model of the hearer and fixes problems that the hearer might otherwise encounter. The hearer can detect equalities by first interpreting the sentence based on the constructions that are already part of his own inventory and the shared situation and prior joint attention. These equalities are candidates for new rules to be constructed by the hearer, and they again involve the introduction of syntactic and semantic categories. Note that syntactic and semantic categories are always local to an agent. The same lateral inhibition dynamics is used for grammatical rules as for lexical rules, and so is also a positive feedback loop leading to a winner-take-all effect for grammatical rules.

## 6 Hierarchy

Natural languages heavily use categories to tighten rule application, but they also introduce additional syntactic markings, such as word order, function words, affixes, morphological variation of word forms, and stress or intonation patterns. These markings are often used to signal to which category

certain words belong. They can be easily incorporated in the formalism developed so far by adding additional descriptors of the units in the syntactic structure. For example, rule [10] can be expanded with word order constraints and the introduction of a particle "ba":

```
[14] ?unit1 | ?ev1 | semcat1(?ev1,?obj2)
      ?unit2 | ?obj2 | semcat2(?obj2)
      ⇔
      ?unit1 | syncat1 (?unit1)
      ?unit2 | syncat2(?unit2)
      ?unit3 | string (?unit3, "ba")
      ?unit4 | syn-subunits ({?unit1, ?unit2, ?unit3}),
      precedes(?unit2, ?unit3)
```

Note that it was necessary to introduce a superunit ?unit4 in order to express the word order constraints between the ba-particle and the unit that introduces the object. Applying this rule as well as the syn-rules and sem-rules discussed earlier to the semantic structure in [5] yields:

```
[13] unit1 | ev1 | fall(ev1,true), fall-1(ev1,obj),
      semcat1(ev1,obj)
      unit2 | obj1 | ball(obj1), semcat2(obj1)
      —
      unit1 | string(unit1, "fall"), syncat-1(unit1)
      unit2 | string(unit2, "ball"), syncat-2(unit2)
      unit3 | string(unit3, "ba")
      unit4 | syn-subunits({unit1,unit2,unit3}),
      precedes(unit2,unit3)
```

When this syntactic structure is rendered, it produces "fall ball ba", or equivalently "ball ba fall", because only the order between "ball" and "ba" is constrained.

Obviously the introduction of additional syntactic features makes the learning of grammatical rules more difficult. Natural languages appear to have meta-level strategies for invention and abduction. For example, a language (like Japanese) tends to use particles for expressing the roles of objects in events and this usage is a strategy both for inventing the expression of a new relation and for guessing what the use of an unknown word in the sentence might be. Another language (like Swahili) uses morphological variations similar to Latin for the same purpose and thus has ended up with a rich set of affixes. In our experiments so far, we have implemented such strategies directly, so that invention and abduction is strongly constrained. We still need to work out a formalism for describing these strategies as meta-rules and research the associated learning mechanisms.

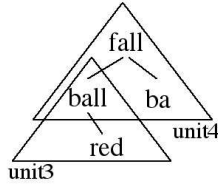


Figure 3: The graph shows the dependency structure as well as the phrase-structure emerging through the application of multiple rules

When the same word participates in several rules, we automatically get the emergence of hierarchical structures. For example, suppose that two predicates are used to draw attention to obj1 in [5]: ball and red. If the lexicon has two separate words for each predicate, then the initial semantic structure would introduce different variables so that the meaning after parsing "fall ball ba red" would be:

[15] fall(?ev,?state), fall-1(?ev,?obj), ball (?obj), red(?obj2)

To resolve the equality between ?obj and ?obj2, the speaker could create the following rule:

[14] ?unit1 | ?obj | semcat3(?obj)

?unit2 | ?obj | semcat4(?obj)

↔

?unit1 | syncat3(?unit1)

?unit2 | syncat4(?unit2)

?unit3 | syn-subunits ({unit1,unit2}), preceeds(unit1,unit2)

The predicate ball is declared to belong to semcat4 and the word "ball" to syncat4. The predicate red belongs to semcat3 and the word "red" to syncat3. Rendering the syntactic structure after application of this rule gives the sentence "fall red ball ba". A hierarchical structure (figure 3) emerges because "ball" participates in two rules.

## 7 Re-use

Agents obviously should not invent new conventions from scratch every time they need one, but rather use as much as possible existing categorisations and hence existing rules. This simple economy principle quickly leads to the kind of syntagmatic and paradigmatic regularities that one finds in natural grammars. For example, if the speaker wants to express that a block is falling, no new semantic or syntactic categories or linking rules are needed but block can simply be declared to belong to semcat4 and "block" to syncat3 and rule [14] applies.

Re-use should be driven by analogy. In one of

the largest experiments we have carried out so far, agents had a way to compute the similarity between two event-structures by pairing the primitive operations making up an event. For example, a pick-up action is decomposed into: an object moving into the direction of another stationary object, the first object then touching the second object, and next the two objects moving together in (roughly) the opposite direction. A put-down action has similar sub-events, except that their ordering is different. The roles of the objects involved (the hand, the object being picked up) are identical and so their grammatical marking could be re-used with very low risk of being misunderstood. When a speaker reuses a grammatical marking for a particular semantic category, this gives a strong hint to the hearer what kind of analogy is expected. By using these invention and abduction strategies, semantic categories like agent or patient gradually emerged in the artificial grammars. Figure 4 visualises the result of this experiment (after 700 games between 2 agents taking turns). The x-axis (randomly) ranks the different predicate-argument relations, the y-axis their markers. Without re-use, every argument would have its own marker. Now several markers (such as "va" or "zu") cover more than one relation.

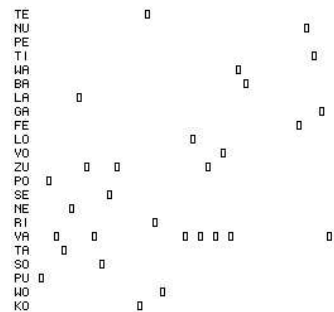


Figure 4: More compact grammars result from re-use based on semantic analogies.

## 8 Conclusions

The paper reports significant steps towards the computational modeling of a constructivist approach to language development. It has introduced aspects of a construction grammar formalism that is designed to handle the flexibility required for emergent developing grammars. It also proposed that invention, abduction, and induction are necessary and sufficient for language learning. Much more technical work remains to be done but already significant experimental results have been obtained with embod-

ied agents playing situated language games. Most of the open questions concern under what circumstances syntactic and semantic categories should be re-used.

Research funded by Sony CSL with additional funding from ESF-OMLL program, EU FET-ECAgents and CNRS OHLL.

## References

- Bergen, B.K. and N.C. Chang. 2003. *Embodied Construction Grammar in Simulation-Based Language Understanding*. TR 02-004, ICSI, Berkeley.
- Cangelosi, and D. Parisi 2003. *Simulating the Evolution of Language*. Springer-Verlag, Berlin.
- Clark, H. and S. Brennan 1991. *Grounding in communication*. In: Resnick, L. J. Levine and S. Teasley (eds.) *Perspectives on Socially Shared Cognition*. APA Books, Washington. p. 127-149.
- Fann, K.T. 1970. *Peirce's Theory of Abduction*. Martinus Nijhoff, The Hague.
- Roy, D. 2001. *Learning Visually Grounded Words and Syntax of Natural Spoken Language*. *Evolution of communication* 4(1).
- Pinker, S. 1998. *Learnability and Cognition: The acquisition of Argument Structure*. The MIT Press, Cambridge Ma.
- Steels, L. 2003 *Evolving grounded communication for robots*. *Trends in Cognitive Science*. Volume 7, Issue 7, July 2003 , pp. 308-312.
- Steels, L. and J-C. Baillie 2003. *Shared Grounding of Event Descriptions by Autonomous Robots*. *Journal of Robotics and Autonomous Systems* 43, 2003, pp. 163-173.
- Tomasello, M. and P.J. Brooks 1999. *Early syntactic development: A Construction Grammar approach* In: Barrett, M. (ed.) (1999) *The Development of Language* Psychology Press, London. pp. 161-190.