How to play the Syntax Game

Luc Steels[1,2] and Emilia Garcia-Casademont[2]

[1]ICREA

[2] Institut de Biologia Evolutiva (UPF-CSIC)

Dr. Aiguader 88, Barcelona 08003, Spain

Abstract

This paper introduces the Syntax Game, a language game for exploring the origins of syntactic structure, specifically phrase structure. We define the game and propose a particular strategy for playing it. We show that this strategy leads to the emergence of a phrase structure grammar through the collective invention, adoption, and alignment of culturally established conventions.

Introduction

The topic of language evolution has been one of the many exciting research threads in Artificial Life since its beginning in the nineties. Most research so far has focused on the self-organization of vocabularies. The Naming Game, first published in the Alife journal in 1995 (Steels, 1995), emerged as the main model system, playing a similar role as the Prisoner's dilemma game for studying the origins of social cooperation. Many researchers have proposed strategies for playing the Naming Game, and studied the semiotic dynamics that unfolds, given particular strategies (Loreto et al., 2011). The Naming Game was also generalized to allow multiple words or deal with combinations of categories and there have been further experiments simulating the co-evolution of concepts and names and to implement Naming Games on real robots (Steels and Hild, 2012).

But human languages go far beyond words. They feature sophisticated grammars which have two functions: (i) to express additional information beyond individual words, e.g. information about tense, aspect, modality, determination, information structure (foreground/background), spatial perspective, etc. and (ii) to help listeners avoid combinatorial explosions and ambiguity, both for semantic interpretation, where combinatorial search and ambiguity unavoidably arise when multiple words are used without signalling how these words are semantically related, and for parsing, because words or patterns tend to have multiple possible functions, generating combinatorial search. In this paper we focus on (ii), i.e. on how grammar arises to dampen syntactic and semantic ambiguity and avoid combinatorial search.

Although there have been several suggestions and proposals to create language game models for the self-organization of syntax, we are lacking a clear game, similar to the Naming Game, that all researchers could use to devise, test and compare strategies leading to grammar and study the semiotic dynamics these strategies generate. The primary goal of this paper is to propose such a game. It is called the Syntax Game. The Syntax Game is very similar to the Naming Game. The key difference is that agents now need to convey semantic networks involving multiple objects, instead of one or more categories pertaining to a single object, and that they can use syntactic means, such as word order, to convey how the arguments of different predicates in the network relate to each other.

Section 2 introduces the game itself and section 3 and 4 discuss why grammar is needed. Section 5 introduces a possible strategy for playing the game and section 6 shows results of simulation experiments with this strategy.

The Syntax Game

A language game models the interaction between two individuals of the same language community. The Syntax Game is a game of reference similar to the Naming Game, i.e. the speaker tries to draw the attention of the hearer to an object in the world. Both agents are assumed to maintain a model of the current situation, called a world model, through perception and action (Spranger et al., 2012). In computer simulations, this world model is synthesized based on an ontology of possible predicates. The Syntax Game then involves the following steps:

1. The speaker selects an object from his world model to act as the topic.

2. The speaker chooses what meaning distinctively describes this object and uses his own lexicon and grammar to translate this meaning into an utterance. The utterance is transmitted to the speaker.

3. The hearer parses this utterance using his own lexicon and grammar in order to reconstruct a possible meaning.

4. The hearer interprets this meaning in terms of his own world model in order to find out what topic the speaker intended. The hearer then signals to the speaker which object he interpreted.

5. The speaker signals success if the topic identified by the hearer is the same as the topic originally chosen by the speaker. If they differ, the speaker signals failure but also points to the object he originally chose.

6. Both speaker and hearer then expand and align their lexicons and grammars based on the outcome of the game.

In classical Naming Game research, the topic is a single object, the meaning a single category, e.g. a color, and the utterance a single word. The Syntax Game allows meanings with several categories and relations implicating several objects, and the utterance can be a set of sequentially ordered words organized in phrases. The challenge of the Syntax Game is to find a language strategy that solves this problem, in other words, that leads to the self-organization of a shared grammar in the population which allows the agents to have communicative success while minimizing the effort in semantic interpretation and syntactic parsing.

Reducing semantic uncertainty

Let us first introduce the sort of meanings speakers and hearers should be able to express. A *world model* consists of a set of objects corresponding to real world entities, e.g. a ball, a pyramid, a person. The objects are labeled as *o-1*, *o-2*, etc. The world model furthermore consists of a conjunction of facts which are true for the current world state, e.g. that the ball is moving, the pyramid is red, and the person is pushing the ball towards the pyramid. World models are represented using standard First Order Logic, that means in terms of predicates with arguments. Predicates are decomposed into an attribute and a value, such as color/red or material/plastic and written in prefix notation. The distinction between attributes and values is needed to generate semantic categories to be used by the grammar (as explained later). Unary predicates are written down as

 (attribute value object)

as in

 (color red o-1) or *(material plastic o-1)*

N-ary predicates, which represent relations, are decomposed. For example a predicate, *moving-away-from* with two arguments, for a mover and for the object the mover moves away from, is decomposed into three predicates, as in the following example:

 (moving away o-3)
 (mover-moving-away o-3 o-1)
 (moving-away-from o-3 o-2)

There is one predicate *moving* with the value *away* for the relation itself which thus becomes reified as *o-3*. This reification is needed because the relation can also be a topic. For example, if the speaker says "the ball moves to the block", the referent of the whole sentence is the moving relation. The other predicates (*mover-moving-away* and *moving-away-from*) explicitly introduce the arguments of the relation. The attribute is the name of the argument, the value is the relation itself, i.e. *o-3*, and the object being predi-

cated is the filler of the argument, e.g. *o-1* in the case of the mover. This decomposition has a number of advantages, and is quite common in AI representations. One advantage is that all facts are tuples with 3 elements: the attribute, the value, and the predicated object.

The different facts in the world model form a *semantic network*. The nodes in the network are facts and if an object occurs more than once in different facts a link is established between them. For example, Figure 1 represents an event where a small paper moves away from a wooden table.
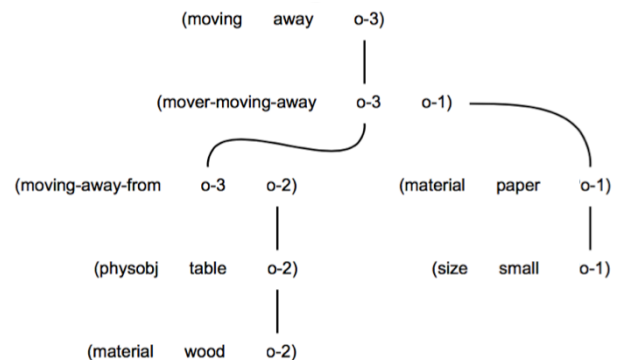

Figure 1: Example of a semantic network representing the world model of a speaker or a hearer.

The speaker in the Syntax Game chooses one object in such a network as the topic (for example, the wooden table *o-2* or the moving event *o-3*) and selects a subnetwork as the meaning of his utterance. This subnetwork can have varying degrees of complexity depending how many properties and relations are involved. For example for the world-model in Figure 1 we could have:

1. *The table* (topic = o-2)
2. *The wooden table* (topic = o-2)
3. *The paper moving away from the table* (topic = o-1)
4. *(I want) the small paper moving away* (topic = o-3), etc.

The speaker knows which specific objects are involved (o-1, o-2, etc.) but the hearer does not. A word like "wooden" signals that there is a wooden object in the scene but not which object is intended. "Table" introduces another object but we do not know whether it is the same as the one introduced by "wooden", because there could be another wooden object, e.g. a block on the table. So after performing lexicon-lookup, the hearer can only derive a set of disconnected semantic subnetworks where the arguments are variables as opposed to objects (Figure 2).

Variables are written with a question-mark in front, as in *?o-1*, *?o-2*, etc. Note also that a fact for the topic has been added. The attribute is called *topic* and the value is either speaker or hearer. The argument of the predicate is the object chosen as the topic, which is here still a variable, namely ?o-8.

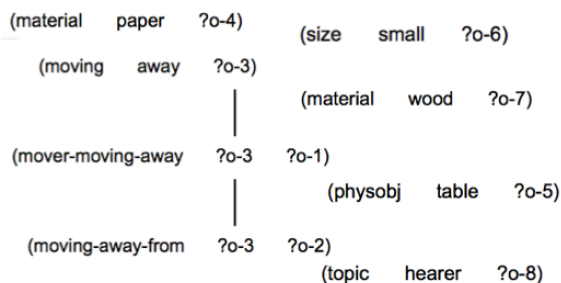Semantic interpretation consists in finding bindings for all

Figure 2: Disconnected network fragments resulting from lexicon lookup by the hearer for the utterance "wooden small moving-away table paper" (no syntax intended).

these variables. Given the world model in Figure 1, the following is a valid set of bindings for the network in Figure 2: {(?o-1 o-1) (?o-4 o-1) (?o-6 o-1) (?o-2 o-2) (?o-5 o-2) (?o-7 o-2) (?o-3 o-3)}. Note that some variables (such as ?o-1 and ?o-5) get bound to the same object. They are said to be *co-referential*.

The hearer can discover the bindings by matching the network that he derived from lexicon-lookup against his own world-model. But this approach, although in principle feasible and undoubtly used by human listeners, has a number of short-comings:

(a) The hearer needs to consider a rapidly exploding set of possible hypotheses $\mathcal{H}_n$. Concretely, H_n, the number of hypotheses, is equal to the number of partitions of the set D of words in an utterance of size n, where a partition of D is defined as a set of nonempty, pairwise disjoint subsets of D whose union is D. H_n is known as the Bell number and defined using the following equation (Bell, 1938):

$$H_{n+1} = \sum_{k=0}^{n} \binom{n}{k} H_k \qquad (1)$$

with $H_0 = H_1 = 1$. So H_n grows double exponentially with the number of words in the utterance. It means that as soon as an utterance is longer than a few words, it is not feasible anymore to rely exclusively on the world model.
(b) But even if the world model would give a possible set of bindings, there can still be remaining semantic ambiguity if there is more than one set of bindings compatible with the hearer's world model. This happens quite often in real dialog because the speaker is not distinctive enough in selecting the subnetwork of his world model that could uniquely identify the topic or assumes facts to be present in the hearer's world model which are not there.
(c) Interpretation through the world model is not possible in the case of displaced communication, where speaker and hearer do not share the same physical context and hence do

not have a common world model (for example when speaking on the phone). And even if they share the same situation, the world-models of situated embodied agents are always different due to differences in perception, a different perspective on the scene, and a different focus of attention.
(d) And finally, even if bindings for all variables could be deduced by matching with the world model, the hearer might still not know which object is intended to be the topic.

Syntax can help because it can signal that two variables have to be bound to the same object, i.e. that they are co-referential. Indeed, if the speaker says "the small paper moves away from the wooden table", he signals that the facts introduced by "small" and "paper" as well as "wooden" and "table" pertain to the same object, i.e. that the variables introduced in the meanings of these words are co-referential. The hearer does this by grouping the words together and order them sequentially to express a noun-phrase pattern. The speaker also signals through the ordering of the constituents at the sentence level and the agreement for number and person between the subject and the verb that "the small paper" (the subject) is the mover and "the wooden table" (the direct object) is the object the paper moves away from. So syntax provides information to link network fragments into a globally coherent network, as in Figure 3. This network can then be matched against the world model to find the actual bindings, but, depending on how effective syntax has been, there will be almost no combinatorial search needed. But often some residual uncertainty remains and has to be disambiguated through the world-model. For example, in the sentence "Maria wants the pyramid on top of the block" it is not clear whether Maria wants the pyramid itself or wants to see it on top of the block.
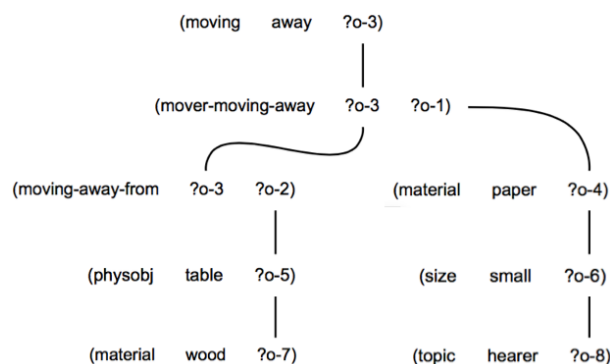

Figure 3: Co-referential links in the hearer network have been linked prior to matching with the world model and a topic has been added.

Reducing syntactic uncertainty

Now we turn to the issue of grammar. In principle, the Syntax Game is neutral with respect to which framework is adopted, as long as the grammar is bi-directional: it should

support the transformation of meaning (in this case a semantic network) into an utterance for language production, and the reconstruction of the meaning of an utterance for language comprehension.

There is a broad consensus in linguistics that the bi-directional mapping of meaning to form proceeds through the intermediary of various linguistic units corresponding to words and phrases and syntactic and semantic categorizations of these units (see Figure 4). Examples of syntactic categorizations are parts of speech also known as lexical categories (Noun, Verb, etc.), agreement features (such as person, gender and number), temporal categories (tense, aspect, modality), syntactic functions (subject, direct-object, etc.), syntactic cases (nominative, accusative), etc. Examples of semantic categorizations include semantic categories (such as event-type, animacy), semantic roles (agents, patients) and case frames.
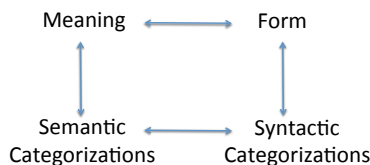

Figure 4: The bi-directional mapping between meaning and form goes through the intermediary of syntactic and semantic categorizations associated with linguistic units such as words and phrases.

There is also a consensus that the mappings of meaning, form, and unit categorizations are packaged in terms of constructions (Fillmore, 1988) and several computational formalisms have been developed recently to operationalize language processing in terms of such constructions. For the simulations reported later, we have used Fluid Construction Grammar (FCG) (Steels, 2011).

Thus a lexical construction creates a unit for a word and associates a word string with meaning and syntactic and semantic categorizations. For example, a lexical construction for the word "paper" associates the string "paper" with the meaning *(material paper ?obj)* and it categorizes the word from a syntactic point of view as a singular noun and from a semantic point of view as being material and having the referent *?obj*. In FCG notation, the lexical construction for "paper" is written as follows:

On the right hand side are the conditions for activating this construction either in production (above the line) or in parsing (below the line). On the left hand side are the features

that are to be added when the construction has become active.

A grammatical construction creates higher order units and determines their syntactic and semantic properties. For example, the sequential occurrence of an adjective and a noun introduces a new unit categorized as a noun phrase, and the meanings of the different constituents are related to each other by establishing co-referential relationships between their variables. Grammatical constructions also assign one of the constituents to be the head and this choice determines other properties (such as the referent) of the higher-order unit. A construction may also introduce new meaning. For example, the di-transitive construction, which underlies sentences such as "he bakes her a cake", adds the meaning that the indirect object ("her") is the recipient of a transaction caused by the subject ("he") and involving the direct object ("a cake") (Goldberg, 2006).

Here is an example of a (very simplified) grammatical construction in FCG-notation: It combines an adjective *?word-unit-1* and a noun *?word-unit-2* into a new hierarchical unit *?np-unit*:

The details of this formalism are not important here. The reader should just remember that the grammar operates through packages of associations between meaning, form, and unit categorizations and that the emergence of a grammar implies not only that agents come up with new grammatical constructions but also that they decide what hierarchical units are needed and what possible syntactic and semantic categories their grammar employs.

Language processing uses a basic data structure (often called the *transient structure*) to represent all units and features for one hypothesis on how to parse or produce an utterance. Language production starts from a transient structure which contains only the meaning of what needs to be expressed and then different constructions are applied until the transient structure contains enough information to articulate the utterance. Language comprehension starts from a transient structure that contains only information about what could be observed in the speech or written form and constructions are again applied until the meaning of the utterance can be extracted and interpreted by matching against the world model. In both cases search becomes unavoidable

as soon as more than one construction gets triggered for the same transient structure. It is well known that this kind of search is also combinatorially explosive.

The world model can be used to reduce search. A transient structure, even if it has not yet processed all words, can often already be partly interpreted and some possible branches can then already be cut off because they are incompatible with the current world model. Human language users certainly use this approach, relying also partially on an ontology to weed out semantically incoherent combinations. But a hearer will encounter the same problems as discussed earlier for semantic interpretation. It would be much better if the grammar is more tight, i.e. uses additional grammatical features or provides additional syntactic marking, to avoid syntactic search. For example, the subject of a sentence is normally the first constituent of the sentence, but this can be made more explicit by requiring agreement between the subject and the verb for number and person.

A constructivist strategy for the Syntax Game

Strategies for building language cannot use most of the standard statistical machine learning techniques because there is no corpus to learn from and learning must be incremental: agents have to build up new grammar through successive situated language games. Instead, we use a constructivist strategy, where agents build hypotheses about what the communal language should or could look like, and adjust their hypotheses based on further interactions. The consecutive application of constructions expands the transient structure to go from meaning to form or vice versa. But occasions will arise when no construction can be applied. At such a point, a strategy for dealing with this impasse should become active (Figure 5). The strategy decomposes into 3 types of meta-operators: Semantic and syntactic meta-operators that try to repair the impasse based on the world model or on stretching existing constructions for new purposes, and learning meta-operators that become active at the end of processing and store what was learned
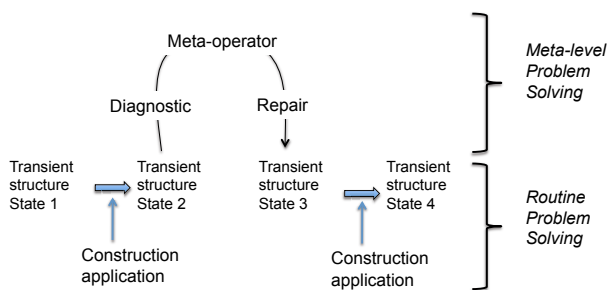

Figure 5: When agents do not have a construction to further expand a transient structure, they move to a meta-level, diagnosing the situation, possibly using the world-model or expansions of the grammar, and then continue routine processing based on constructions.

Semantic meta-operators

When no (partially) matching constructions can be found, it is possible to use the world model and combine units for words or phrases based on semantics, specifically:

+ **Build-or-extend-group**: If two words or word groups expressing unary predicates refer to the same object, they can be combined. For example, if there is a group for *wooden table* (based on an existing construction) and the utterance is *small wooden table*, the word-unit for *small* can be linked in with the group-unit for *wooden table*. The group-unit retains the same referent.

+ **Build-Hierarchy** When a relational word is encountered, i.e. a word which introduces a predicate with more than one argument, such as *moves-away-from*, and no constructions are available to handle this word, then the Build-Hierarchy meta-operator looks in the world-model to detect which object plays which roles and then combines the units for these objects into a new hierarchical unit.

The Build-Hierarchy meta-operator also decides which of the arguments is going to be the referent depending on the role of the arguments in the rest of the semantic network and determines on that basis the phrasal category of the hierarchical unit as well as its semantic category. For example, suppose the speaker must express an on-top-of relation with two arguments for the top and the bottom, then the head will be the unit introducing the top if the top is the object that links into the rest of the semantic network as in "the pyramid on top of the block". The hierarchical unit will be a noun-phrase (because the pyramid is already a noun-phrase) and its semantic category will be physical object.

Note that the Build-Hierarchy meta-operator leads to recursive syntax, because a noun-phrase (such as the block in "the pyramid on top of the block") is itself a constituent of a noun-phrase and this can go on at several levels as in "the pyramid on top of the block sitting on the table standing on the floor inside the room".

Syntactic meta-operators

When partially matching constructions can be found, it is possible to handle the impasse by either coercing words to fit into the partially matching construction (syntactic coercion) or to expand the applicability of the construction by making it more general (extension). More specifically,

+ **Coercion:** A construction is found that is semantically compatible but one word does not have the appropriate lexical category (as in the example of "googled" where a noun occurs in a context where a verb is expected). The Coercion meta-operator then adds the lexical category to the word-unit in the transient structure and the construction can apply.

+ **Extension:** A construction is found for which the syntactic constraints match but a required semantic category is missing. The Extension meta-operator then adds this semantic category to the construction so that it can apply.

Learning Operators

When an utterance could be successfully parsed or produced after one or more repairs, the learner activates learning operators to integrate the insights that were obtained into his construction inventory. The following learning operators have been implemented:

+ Memoization: This learning operator acts on the result of the semantic operators (build-or-extend-group and build-hierarchy). It builds a new construction, which should trigger when observing the relevant subunits and create a new superunit. There are two cases. (i) When the units are words expressing unary predicates, the head is the unit with the most referential power. (The hierarchy of referential power is provided by the ontology, e.g. physobj has more referential power than color.) The syntactic category of the superunit is equal to the phrasal variant of the lexical category of the head if it is a word, e.g. noun-phrase if the head is a noun, or the same phrasal category, if the head is already a phrase. And the lexical categories of the respective subunits are either randomly chosen from the existing categories, or, if there are no such categories associated already with the implicated word in the lexicon, a new lexical category is created and stored. The new categories are labeled *syn-cat-1*, *syn-cat-2*, etc. (ii) When the superunit is formed to handle a relational word, the head is the chosen referent, and the syntactic category of the superunit is the phrasal variant associated with the head (e.g. prepositional-phrase if the head is a preposition).

+ Enact-Coercion: This learning operator records the result of coercion by storing the assumed lexical category in the lexical construction of the relevant word.

+ Enact-Extension: This learning operator records the result of extension by expanding the set of semantic categories of the construction that was extended.

Alignment

The meta-operators and learning-operators are hypotheses made by the speaker and the hearer. Because neither of them has an absolute overview of the language and cannot inspect the internal states of the other agents, some hypotheses may be erroneous. The agents therefore need an additional mechanism to progressively discard wrong hypotheses based on further interactions. Theoretical research on the Naming Game has shown that it is best to let only the hearer adjust these scores De Vylder and Tuyls (2006) and to use a lateral inhibition learning rule, one of the strategies commonly used for the Naming Game Steels (1998). Knowing which constructions c_i need an increased score is easy: they are the constructions that were used on the path towards the final transient structure. We use the following update rule: $\sigma_{c_i} \leftarrow \sigma_{c_i}(1 - \gamma) + \gamma$, with $\gamma = 0.1$ a constant.

Competing constructions c_j need to be decreased using the following update rule: $\sigma_{c_j} \leftarrow \sigma_{c_j}(1 - \gamma)$. How can the agent determine competing constructions? First of all, they

include all constructions that started off a wrong branch in the search space during comprehension, i.e. a branch which was not on the path towards the final solution. When such constructions are discarded, this will minimize the syntactic uncertainty S_t. Second, the listener can produce himself the utterance based on the meaning deduced from the comprehension process and then find all constructions that would start off a wrong branch in producing, i.e. a branch that would not lead to the utterance produced by the speaker. Their scores need to be decreased as well.

Results

Because the Syntax Game focuses on grammar, we start simulations from a pre-defined set of lexical constructions associating word strings with meanings. Semantic categories are directly derived from the attributes of the meaning. The agents must autonomously introduce and assign lexical categories to the words and introduce new hierarchical units, categories for these units and grammatical constructions that build them. Each agent has to do this independently but a shared common grammar has to emerge through self-organization.

The first simulation experiment tests whether an agent is able to learn an existing grammar in one-on-one interactions. The tutor is initialized with a lexicon of 40 lexical constructions and a grammar with 30 grammatical constructions. The tutor grammar includes adverbs, adjectives, nouns, verbs, prepositions, pronouns and relative pronouns as well as noun phrases of different levels of complexity, verb phrases, main clauses and relative clauses. The constructions produce utterances in a reduced English, without articles and without grammatical agreement. Each sentence describes a particular topic (object or event) in a scene. Some example utterances are "Paul sees (the) red carpet (that) Emilia wants", "big red table on small stone" or "Paul believes (that) Emilia wants (the) carpet on (the) big wooden table". The learner is initialized with the same lexicon and endowed with the various operators described above, but without any grammatical constructions or lexical and phrasal categories. Each experiment is carried out for 5 different tutor-learner pairs, using random choices from a set 20 situations, so that results are comparable.

Words have potentially more than one lexical category (as in human languages), e.g. "paper" can be a noun as well as an adjective (as in a paper towel), although only one lexical category can fit with a grammatical construction. On the other hand a grammatical construction can accept more than one semantic category for a particular unit, which then makes the construction more general from a semantic point of view.

Figure 6 shows the result of a tutor-learner experiment. It shows 500 consecutive language games. The running average of four different measures is shown for 2 different experiments:

(i) The *semantic uncertainty*, which measures the number of times the world model of the situation is used to generate or block hypotheses, divided by the number of variables introduced by the lexicon for the utterance. More precisely, when grammar or syntactic meta-operators have been used, it measures the number of hypotheses that were not plausible according to the world model and when semantic meta-operators have been used, it measures the number of times the situation was used to generate possible hypotheses.

(ii) The *syntactic uncertainty*, which measures the number of extra hypotheses that grammatical constructions generated during processing. This is the number of splits in the search space that were due to multiple results of applying a grammatical construction, divided by the number of words in the utterance.

(iii) The *communicative success*, which measures whether the hearer was able to identify the topic without speaker feedback.

(iv) The *alignment* which measures whether the hearer would express the same meaning using the same utterance as the speaker.

(v) The *grammatical constructions* measure, which measures how fast the grammar is acquired by measuring the percentage of constructions learned of the final grammar.

Thanks to the grammar we see that both types of uncertainty get drastically reduced, which proves that the grammar achieves its desired purpose of minimizing uncertainty.
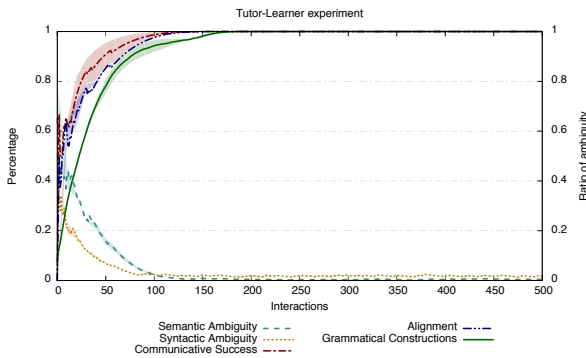
Figure 6: Tutor-learner experiment. The learner rapidly (after 150 interactions) acquires the grammar of the tutor, and a total alignment and communicative success with the tutor. On the other side, semantic and syntactic uncertainties of the learner decrease drastically as long as grammar is learned.

Figure 7 shows the results of an experiment in which a population of 5 agents develops a new grammar from scratch. The agents start with a shared lexicon but no syntactic categories and no grammatical constructions. They build very quickly a common grammar that provides high communicative success, total alignment as well as a reduction of syntactic and semantic uncertainty.

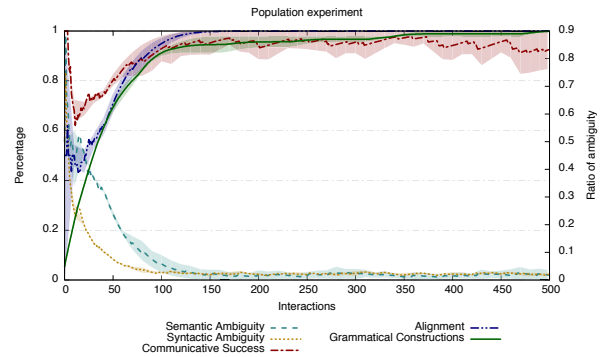What about the complexity of the grammar? We ob-

Figure 7: Grammar emergence experiment. The same measures as in 6 are shown.

serve that the agents construct noun-phrase-like constructions with categories reminiscent of adjectives and nouns. They also build relational constructions with lexical categories similar to prepositions and verbs as well as relative clauses such as "(the) red carpet (where) emilia wants (the) table (to be) on." There are many examples of recursive structures, particularly embedded clauses and embedded noun-phrases.

It is not straightforward to compare the grammars of two agents directly because agents have different syntactic categories. To visualise nevertheless the similarity between syntactic categories we have used the multidimensional scaling method. In order to apply it, we have defined a vector space to describe every syntactic categories as a vector in the following way: $Cat_i \in \{0, 1\}^n$ where n equals the number of words in the lexicon and position j is 1 when word w_j can be used as category Cat_i, and 0 when it cannot. Finally we have applied the method to these corresponding sets of vectors by using the Euclidean distance. Figure 8 shows results from the Tutor-Learner experiment where we see that the learner is able to form categories similar to the syntactic categories that the tutor uses.

Conclusion

This paper introduced the Syntax Game, a minimal language game for exploring how grammar can arise in a population of agents. The game is a variant of the Naming Game but introduces more complex semantics. Grammar is needed to avoid semantic and syntactic uncertainty which lead, respectively, to a combinatorial explosion in semantic interpretation and in parsing. Hence a language cannot scale up to utterances beyond a few words without minimizing these sources of utterance in language processing. We discussed an example of a language strategy, which allows agents to self-organize a phrase structure grammar that uses sequen-
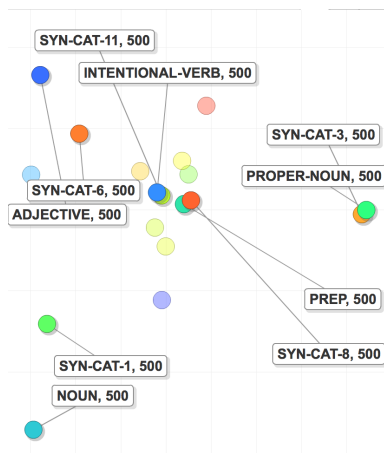
Figure 8: MDS plot, useful for showing how syntactic categories of two agents become similar.

tial ordering and grouping of words into phrases in order to avoid semantic and syntactic uncertainty. This strategy uses a set of meta-operators and alignment based on a lateral inhibition learning rule. Experimental results show that the strategy achieves the desired results.

Acknowledgments

References

Bell, E. (1938). The iterated exponential integers. *The Annals of Mathematics*, 39.

De Vylder, B. and Tuyls, K. (2006). How to reach linguistic consensus: A proof of convergence for the naming game. *Journal of Theoretical Biology*, 242(4):818–831.

Fillmore, C. J. (1988). The mechanisms of "Construction Grammar". In *Proceedings of the Fourteenth Annual Meeting of the Berkeley Linguistics Society*, pages 35–55, Berkeley CA. Berkeley Linguistics Society.

Goldberg, A. E. (2006). *Constructions at Work, The Nature of Generalization in Language*. Oxford University Press, Great Clarendon Street, Oxford.

Loreto, V., Baronchelli, A., Mukherjee, A., Puglisi, A., and Tria, F. (2011). Statistical physics of language dynamics. *JOURNAL OF STATISTICAL MECHANICS: THEORY AND EXPERIMENT*, P04006.

Spranger, M., Loetzsch, M., and Steels, L. (2012). A Perceptual System for Language Game Experiments. In

Steels, L. and Hild, M., editors, *Language Grounding in Robots*, pages 89–110. Springer.

Steels, L. (1995). A self-organizing spatial vocabulary. *Artificial Life Journal*, 2(3):319–332.

Steels, L. (1998). The origins of ontologies and communication conventions in multi-agent systems. *Journal of Agents and Multi-Agent Systems*, 1(2):169–194.

Steels, L., editor (2011). *Design Patterns in Fluid Construction Grammar*. John Benjamins.

Steels, L. and Hild, M., editors (2012). *Language Grounding in Robots*. Springer, New York.