A Dynamic Representation of Grammatical Relations

Siamak Rezaei Durroei, Matthew W. Crocker

University of Edinburgh

Proceedings of the LFG97 Conference

University of California, San Diego

Miriam Butt and Tracy Holloway King (Editors)

1997

CSLI Publications

http://www-csli.stanford.edu/publications/

Abstract

This paper presents a dynamic approach to modelling grammatical relations. For this purpose, we turn to dynamic theories in Computer Science such as CSP (Communicating Sequential Processes) [Hoare1978] and π-calculus [Milner1993] and by borrowing some concepts from these theories, we will discuss how communicating linguistic processes can be defined and constructed. Our approach, in spirit shares some similarities with parallel lexicalised theories such as LFG (Lexical Functional Grammar), but our main focus has been on defining the notion of grammatical channels for communication between processes, which will further be elaborated in this paper. We will also contrast some of our main design issues with some aspects of LFG.

1 Introduction

Modelling grammatical relations has played an important role in recent linguistic theories. In diverse linguistic theories such as Government Binding (GB), Lexical Functional Grammar (LFG) and Head Driven Phrase Structure Grammar (HPSG), grammatical relations are expressed differently.

In GB, the grammatical relations are derived from a combination of configurational positions and thematic roles in the theory, while within HPSG they are derived from SUBCAT list and the relative position of arguments in it. These two approaches contrasts with LFG in which the grammatical relations (functions) are primitive elements in the theory.

Whether we consider grammatical relations as primitive elements in the theory or we derive them from other principles of the theory, this decision will have further consequences for our future expansion of the theory. Here our main attempt is to add a dynamic aspect to the present approaches to linguistic modelling and we will show that choosing the former view of representing grammatical relations (i.e as primitive elements) is less problematic for our extension.

In the next section we will restrict our definition of dynamism. We also give the motivation for our work. We will briefly review some examples of dynamic approaches to linguistic analysis. In Section 3 the main idea will be presented and Section 4 gives more details of linguistic channels. Finally in the last section, possible extensions to this approach will be discussed and the main aspects of this approach will be compared with GB and LFG [Sells1985].

2 Dynamism, Communication and Competition

Recently, there have been attempts to introduce dynamic approaches based on logic of *information flow* for modelling some aspects of linguistic phenomena; theories such as situation theory [Cooper et al.1990] and channel theory [Barwise and Seligman1997] take this radical view that is based on flow of information from one object (process) to another. This contrasts with unification approaches which are *declarative* and have no explicit flow of information. The main application of these information based approaches has been in the domain of semantics and discourse and little work has been done in syntax.

Building a communicative model for syntax will help us to add another dimension to representation of language as flow of information and by this we can develop a full-fledged theory based on communication for language modelling. A further motivation for our approach is extending grammatical relations with probability measures and to develop a framework for modelling competition for these as grammatical resources.

Existing attempts to implement traditional theories in a parallel and competitive framework (e.g. [Stevenson1993] and [Choi1996]) will force us to change and modify some of the basic assumptions in our present theories. For example [Bresnan1996] and [Choi1996] in working on optimal LFG argue for full specification of under-specified grammatical functions (GFs) in LFG. This optimal extension to LFG is in conflict with the assumption of underspecification in LFG. It is not clear how long distance scrambling can be captured in such optimal LFG.

We think that considering dynamism, communication and competition in the linguistic representation will further help us to build better models of natural language processing in a competitive and parallel distributed framework.

From a theoretical and mathematical perspective, the notion of dynamism in our model is based on the actor model of computation [Agha and Hewitt1987]. This model combines object-oriented methodology with concurrency and distribution. The model assumes that a collection of independent objects (actors) communicate via asynchronous message passing. In this model a process can be thought of as an object with a state that can be changed by the process. For changing the state of an object a message can be sent to that object and an object may send messages to other objects. Objects can create instances of themselves or different objects. But what is the relevance of this concurrency model to linguistics?

The traditional approach in linguistics views objects in syntactic constructs as passive data structures which are manipulated by a set of rules, or global constraints, principles or modules. There are very few works that treat syntactic constructs as active objects (processes) that communicate with each other and compete for grammatical resources.

[Trehan and Wilk1988] is one of the earliest approaches based on actor model, which attempts to introduce dynamism into parsing and syntax. Trehan tries to implement a chart parser in a parallel environment. For this purpose he treats incomplete phrases as active processes which are looking for inactive processes (i.e. completed phrases on the left hand side of the rules or words). For example in:

VP_2 -> VP_1, NP

VP_1 is an incomplete process (edge) which is looking for an NP to become completed. After attachment of an NP to VP_1, a VP_2 will be generated as a completed process. Trehan uses Context Free Grammar (CFG) rules for expressing the relationship between processes. In his approach the phrases are treated as processes, but the channels of communications between processes are not part of the linguistic theory and the existence of channels in the implementation is an implementational issue and is specific to the parser architecture.

ParseTalk [Broker et al.1994] is a recent parser based on actor model which is based on dependency grammar and finally [Fujinami1996] is another recent process based approach to language analysis.

All these models try to implement one of the present linguistic models in a concurrent and dynamic framework, without trying to introduce the concurrency into the linguistic model. In all these models the medium of communication among linguistic processes are relevant to the parsing architecture and they are not part of the linguistic model.

But what should be the domain for interaction of these processes and what is the medium for their communication? Is there any notion in linguistic theory that can be used as communication channels?

3 Process Structures and Grammatical Channels

In the following we will try to answer some of those questions. The major building block in our model is process structure. We assume that structures or constituents like NP or PP exist in languages and they are the product of recurrent patterns in a language.

But by referring to these constituents or structures as process structures we view these structures dynamically and associate time-period, locality, activation and other measures with these structures. In this view the structures can communicate with each other and interact, hence we define communicating and interacting process structures [Rezaei-Durroei1997]. Our main attempt in this paper was to look at linguistic representation from this perspective and define linguistic process structures that communicate with each other and highlight the role of communication in defining the syntax of language.

We will distinguish between three kinds of process structures. First there are the clause processes in which the communication can occur. Inside a clause there are process structures like unmarked NPs and marked NPs (with preposition or postposition) which compete with each other for the grammatical resources of the clause, such as subject and object.

The resources are offered by another type of process structure such as verb[1].
This production and consumption resources occurs inside a clause process structure. The grammatical resources are transfered via channels of communication (with the same name as resources) between verbs and arguments. Each channel has two variables: the producer, which will bind to a verb process and the consumer which will refer to an NP process. subj(Producer, Consumer) and obj(Produver, Consumer) are instances of such channels. The channels in our model are for communication between two process and hence we are following CSP in this respect[2].

To deal with Long Distance Scrambling (LDS)[3] we use a mechanism which is analogous to functional uncertainty in theories like LFG. Under certain conditions (e.g. barriers theory of GB) some channels can be passed or exported from one clause to an embedded one. Since we use time indices to refer to processes (starting time and ending time), upon exporting a channel there will be no need for renaming the variables of the channel.

To sum up, in this section we introduced linguistic processes as phrases (sequence of words) which communicate through grammatical channels. Channels are instances of grammatical relations. Each process has a time duration which marks the starting and ending word of the process in the sentence. To capture long distance scrambling, we introduced the notion of mobile channels which are exported from one clause to another.

In the next section, we will have a closer look at channels and discuss the competition of processes for channels.

4 Channel Based Modelling

In the following we will elaborate on different data structures, operators and constraints that we have designed for channels. For channels, we have considered two stages: marking and acquisition. The first step is channel marking, when the pre/post positions case mark a bare NP, at this stage the possible channels that an NP-process can compete for are specified. At the final stage, only one of these channels can be acquired by the process and this stage is channel acquisition.

In a GB framework for Parsing Warlpiri, [Kashket1986] distinguishes between case marking and case assignment. Our notions of channel marking and acquisition, in spirit, refers to the same mechanism. There are certain constraints on channel marking and acquisition in different languages. For fixed word order languages the location and positions are very

[1] In general, a process structure may receive and/or offer a number of channels at the same time.

[2] Unlike CSP, the communication over these channels are asynchronous and numbers and indices are passed over them.

[3] In LDS a constituent will be moved across the boundary of two clause boundaries.

important parameters for channel marking. In general channel marking is of two types: deterministic and non-deterministic. In nondeterministic the marked process may choose among multiple channels. In contrast, for deterministic marking, the process is marked for a unique channel.

For cases where there are multiple choices, competition determines which channel is acquired by a process. For each clause we consider a channel set for representing competition of channel acquisition. A channel set consists of channel sequences to which probability values are attached. We refer to each of these sequences as a channel path. subj(–,–).obj(–,–) and obj(–,–).subj(–,–) are two examples of channel paths.

4.1 Channel Constraints

The word order constraints in our model are expressed for channels and we deviate from frameworks such as LFG which use an ID/LP notation. In our framework the word order constraints are defined local to a clause and not for rules, and they specify the precedence relations between two channels in a channel path.
The precedence relations are probabilistic and each channel order has a probability measure attached to it. We have two types of channel order constraints which are applied to channel paths: hard and soft. The hard constraints cannot be violated, while the soft ones can be violated. The violation of a hard constraint makes the channel path inactive, while the violation of a soft constraint reduces the level of activity of that specific channel path. For simplicity we assume that the activity level is the same as probability number. This is not true and the activity level is a fuzzy/possibility measure and we have modelled it based on possibility theory [Dubois and Prade1993]. This mechanism provides us with a notion of graded grammaticality. An analogous notion of grammaticality has been defined based on optimality theory (see [Keller1996]). But our framework is more flexible than optimality theory (OT).

In OT, there is a hierarchy of constraints. In a hierarchy, for any two constraints c1 and c2 we can have c1 $\ll$ c2 or c2 $\ll$ c1, but not both. Here $\ll$ shows priority of one constraint over another. OT uses a GEN function that generates an array of candidate surface representations and the constraints are applied to all the candidates. Since in OT constraints can be violated, we can have optimal and suboptimal candidates for which one or more number of constraints doesn't apply. The competition criteria is that the candidate that satisfies a higher ranking constraint will win. The main criticism to OT is that it doesn't allow a set of low ranking constraints to conspire to override a high ranking constraint. In our model, which shares greater similarity with Harmony Grammar (HG) [Smolensky, Legendre, and Miyata1992], this behaviour can be expressed.

In HG there is no explicit notion of hierarchy of constraints, and here we consider for each constraint a positive and negative contribution pair. When a constraint is satisfied

	C1	C2	C3	C1 [- 2 + 0]	C2 [- 3 + 0]	C3 [- 5 + 0]
a	****	**		4(-) 1(+)	2(-) 2(+)	1(+)
b	*		*	1(-) 4(+)	4(+)	1(-)
c			*	5(+)	4(+)	1(-)

Table 1: A comparison of OT and our approach

by a candidate, the corresponding plus contribution of the constraint will be added up to the activity measure of the candidate. But if a constraint is violated, then the minus contribution of the constraint will be added to the activity measure of the candidate. Depending on the values of the + and – contributions and the semantics of addition and subtraction operators we can express different relationships among constraints and it is possible for small constraints to "gang up" against bigger constraints. An example of this is shown in Table 1. Assume C1 $\ll$ C2 $\ll$ C3 as the constraint hierarchy.
In OT version, the candidates will be ranked as a, c, b. While in HG version, the candidates will be ranked as:

a: (4*(-2) + 2*(-3))= -14
b: (1*(-2) + 2*(-5) = -12
c: (2*(-5)) = -10

Hence the ordering will be c,b,a. The numbers have been chosen so that a default ordering between the constraints also exists.

To model "ganging up" phenomenon, we need both plus/minus measures, because it is the sum of positive and negative contributions which counts and we need to know how many constraints are satisfied and how many are not satisfied. But the plus/minus contributions can have different values and can be even equal or zero. This approach is more flexible than OT and it can also adapt to ill-formed data by lowering the contribution of a constraint or making it zero - for example in cases where a constraint is violated all the time. It is worth investigating whether the combination of HG and the notion of channel path will be more suitable for tackling reanalysis and examples of garden path in psycholinguistics research.

The possible channel combinations are restricted by channel order constraints which are imposed on channel sequences. They are of the form:

(1) chnl1 $\prec^{no+}_{no-}$ chnl2

The no- contributes to lowering the activity of a path, while a no+ contributes to raising the activeness of the path. A channel resource cannot be allocated twice in a channel sequence. We represent this in our framework as a hard precedence constraint. E.g. subj $\succ\!\!-\!\!\prec$ subj. Violating a hard constraint in a path makes the path inactive and hence closed. This contributes to closing down of some paths and reducing the number of alternatives.

The channel sets provide the mechanism for a process to compete for one or more channels at the same time and a set of paths can progress in parallel. The competition strategy that we have adopted is partial commitment strategy. We are following neither committed choice nor incremental commitment strategies. In committed choice strategy (e.g. in Parlog), a process must commit itself to one of the successful choices and discard (de-activate) the others, while in incremental commitment strategies (in NLP) a single choice is committed to and in case of deadlock or failure, by backtracking or reanalysis another choice can be adopted. In contrast we use a partial commitment strategy and all active channel paths are partially active at the same time, but the most active path will win at the end.

We argued that the choice for possible channel paths are restricted by channel order constraints and resource limitation constrains. Hence these context dependent constraints reduce the range of possibilities and make the strategy decidable. Put another way, we have introduced a notion of partial and soft commitment, which is fitted into the general model.

We will let all competing paths be active in parallel and will commit to one path as late as possible[4]. The path with highest activity will be the winning path, if its activity level doesn't go down.

4.2 Why Imperfect Channels

Our approach to modelling grammatical relations by channels diverges from LFG and the channels in our model are imperfect. In other words there are probabilities/possibilities assigned to them. In current probabilistic approaches to LFG the probabilities are assigned to feature structures and the stochastic model becomes very complex and there is no automatic mechanism for collecting these probabilities from corpus; or it is not straightforward. It should be noted that in LFG the functional uncertainty mechanism creates many theoretical obstacles to a probabilistic extension of LFG. To my knowledge no satisfactory probabilistic method for modelling Long Distance Scrambling (LDS) for LFG exist. LDS is captured in LFG by functional uncertainty.

One general criticism to apply probabilistic techniques to under-specified structures or categories is that the two notions are not consistent with each other. In probabilistic approaches, we specify categories and assign numbers (i.e. probabilities) to these non-overlapping categories, while in underspecification, we avoid specific categories and employ general and under-specified categories that can potentially represent a range of categories.

[4]This is the clause boundary position, where a choice is committed to.

This notion of general unspecified category and in contrast the non-overlapping specific categories (with probabilities) are sometimes in conflict with each other. This is the main turning point in our framework for having imperfect (fuzzy) fully specified grammatical relations (i.e. channels) and away from present discussions in LFG that go from under-specified grammatical relations towards more under-specified notions (such as +o -o).

From a probabilistic point of view, the introduction of imperfect channels also helps to specify the probability of pro-dropness in languages such as Persian and Japanese which are highly pro-drop. It introduces another level of abstraction for representing the present flat probabilities for verb argument attachment. This contributes to a better notion of probability for pro-drop languages.

In (2) we have shown the classical probabilistic approach to modelling verb argument attachment. Here #Prob, P,N and V correspond to probability, preposition, head noun of NP and the verb respectively. (2) is the result of joining (3) and (4) tuples and deleting the channel part.

(2) (#Prob, P, N, V)

(3) (#Prob, P, N, Channel)

(4) (#Prob, V, Channel)

Conceptually we need this extra third relation for pro-drop languages which is represented naturally in our framework. Our model can further be extended to represent long distance scrambling in a probabilistic approach.

5 Further Work

In our framework, long distance scrambling was represented by mobile channels and exporting channels into embedded clauses while local scrambling is captured by competition inside channel sets.

In Table 2 we have compared some of the features of our framework with GB and LFG. This is one of our first attempts to investigate the possibility of representing linguistic phenomena as a set of communicating processes that compete with each other for channel resources[5].

The present framework suggests a new approach to modelling linguistic phenomena using a communicative theory of language which is based on concurrent processes. The performance issues have been incorporated into the foundations of the framework and not added

[5] See [Rezaei and Crocker1995] for a previous work.

GB	LFG	CPS based
Case	?	Channel marking
Theta theory	Functional uncertainty	Channel acquisition
Move$-\alpha$	Functional uncertainty	(Mobile) channels
Barriers	?	Constraints on mobility
Control	Control	Conditional channels
Optimality	probabilistic feature structure	Fuzzy channels

Table 2: A Comparison of our framework with GB and LFG

on top of the theory. It is worthwhile investigating the possibility of developing a channel algebra for this framework. The algebra will be an instance of discrete time probabilistic process algebras. The CFG rules and the mobility of channels can be represented with a CFG extension to a subset of π-calculus operators. [Burkart and Steffen1992] is an example of a CFG process algebra.

The channels in our model are binary communication links and hence are more restricted than π-calculus channels. In addition an uncertainty number is associated with each of them that shows the level of activity of the channel (path). We have used sequence, parallel and choice operators for constructing channel paths. The sequence operator is needed for constructing a sequence of channels and the parallel operator is needed for capturing the possible parallelism among channel paths. We need to have choice operator to represent the choice between two competing channel paths. In addition, we have used a time precedence binary operator to represent the channel precedence constraints.

The main problem is to specify the notion of fuzzy (imperfect) channel which can fail to communicate and communication is optional over it. This will form the basic building block for specifying a semantics for this channel algebra which can model linguistic concurrent processes. Finally one interesting aspect of the model is restricting communication and constraints on communication in localised domains such as clause. This brings a better notion of interaction (See [Abramsky1996]).

To sum up, in this paper we tried to show that considering grammatical relations as primitive elements in theories like LFG will help us to incorporate a further domain of dynamism into the static structure of present linguistic theories. But we argued against underspecification in LFG which creates many theoretical obstacles in extending LFG with probabilistic notions or incorporating optimality into LFG. Instead we proposed that it will be fruitful to add probabilities to LFG and extend such framework by concentrating on dynamic aspects of grammatical relations. In other words, we consider grammatical relations as imperfect channels for communication between linguistic processes. Finally we have proposed that in a dynamic framework, we can use the notion of channel exporting for capturing long distance scrambling to substitute underspecification. Our approach suggests a new approach to modelling linguistic knowledge as a network of distributed and communicating process structures. This needs further research.

References

[Abramsky1996] Abramsky, Samson. 1996. Retracting some paths in process algebra. In *Proceedings of CONCUR 96, number 1119 in Lecture Notes in Computer Science*, pages 1–17, Springer Verlag.

[Agha and Hewitt1987] Agha, Gul A. and Carl Hewitt. 1987. Concurrent programming using actors. In *A. Yonezawa; M. Tokoro (eds) Object Oriented Concurrent Programming*, Cambridge, Massachusetts. MIT Press.

[Barwise and Seligman1997] Barwise, Jon and Jerry Seligman. 1997. Information flow in distributed systems. Cambridge University Press (In Press).

[Bresnan1996] Bresnan, Joan. 1996. Optimal syntax: Notes on projection, heads, and optimality. MS. Stanford University.

[Broker et al.1994] Broker, Norbert, Micheal Strube, Susanne Schacht, and Udo Hahn. 1994. Coarse-grained parallelism in natural language understanding: Parsing as message passing. In *Proceedings of the International Conference on New Methods in Language Processing (NeMLaP)*, pages 182–189, Manchester, UK, Sept.

[Burkart and Steffen1992] Burkart, Olaf and Brenhard Steffen. 1992. Model checking for context-free processes. In *Proceedings of CONCUR 92, number 630 in Lecture Notes in Computer Science*, pages 123–137, Springer Verlag.

[Choi1996] Choi, Hye-Won. 1996. *Optimizing Structure in Context: Scrambling and Information Structure.* Ph.D. thesis, Stanford University, August.

[Cooper et al.1990] Cooper, Robin, Kuniaki Mukai, , and John Perr (eds). 1990. *Situation Theory and its Applications, Volume1.* CSLI, Stanford.

[Dubois and Prade1993] Dubois, Didier and Henri Prade. 1993. Possibility theory, belief revision and nonmonotonic logic. In *Lecture Notes in Artificial Intelligence, IJCAI 93 workshop, Anca L. Ralescu Eds.*

[Fujinami1996] Fujinami, Tsutomu. 1996. *A Process Algebraic Approach to Computational Linguistics.* Ph.D. thesis, Centre for Cognitive Science, University of Edinburgh.

[Hoare1978] Hoare, C. A. 1978. Communicating Sequential Processes. In *Communication of the ACM, 21(8)*, pages 666–677, August.

[Kashket1986] Kashket, Michael B. 1986. Parsing a Free-Word Order Language: Warlpiri. In *24th proceedings of the ACL*, pages 60–66.

[Keller1996] Keller, Frank. 1996. Extraction from complex noun phrases. a case study in graded grammaticality. Master's thesis, Institute for Computational Linguistics, University of Stuttgart.

[Milner1993] Milner, Robin. 1993. The polyadic π-calculus: a tutorial. In *F. L. Bauer, W. Brauer, and H. Schwichtenberg, editors, Logic and Algebra of Specification*, pages 203–246, Springer Verlag.

[Rezaei and Crocker1995] Rezaei, Siamak and Matthew Crocker. 1995. A Distributed Architecture for Parsing Persian. In *Proceedings of ICSCS*, Sharif Univ. of Technology, Tehran, December.

[Rezaei-Durroei1997] Rezaei-Durroei, Siamak. 1997. Linguistic communicating process structures. Australasian Natural Language Processing workshop.

[Sells1985] Sells, Peter. 1985. *Lectures on Contemporary Syntactic Theories.* CSLI, Stanford.

[Smolensky, Legendre, and Miyata1992] Smolensky, Paul, Geraldine Legendre, and Yoshiro Miyata. 1992. Principles for an integrated connectionist/symbolic theory of higher cognition. Technical Report Report CU-CS-600-92, Computer Science, University of Colorado at Boulder, July.

[Stevenson1993] Stevenson, Suzanne. 1993. A Competition-Based Explanation of Syntactic Attachment Preferences and Garden Path Phenomena. In *Proceedings of the 31st Meeting of the Association for Computational Linguistics*, pages 266–273, Columbus, Ohio, June.

[Trehan and Wilk1988] Trehan, R. and P. F. Wilk. 1988. A parallel chart parser for the commited choice non-deterministic(CCND) logic language. Technical Report AIAI-TR-36, Artificial Intelligence Applications Institute, January.

siamakr@cogsci.ed.ac.uk
mwc@cogsci.ed.ac.uk
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh EH8 9LW
Scotland, UK