Lambek's computational approach to conjugation

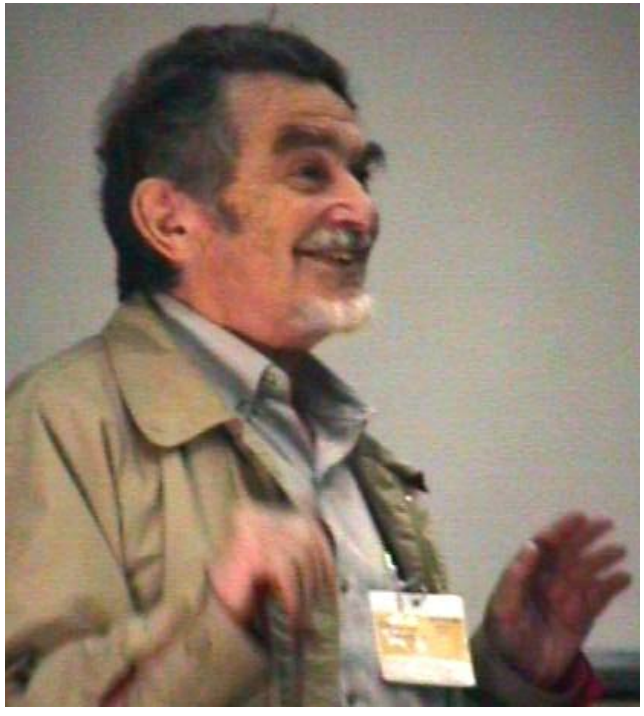
Silvia Ghilezan

University of Novi Sad

LAP 2014, IUC, Dubrovnik, September 22-26, 2014

Joachim, Jim, Lambek

1922, Leipzig - 2014, Montréal



"For more than 60 years, Jim Lambek has been a profoundly inspirational mathematician, with groundbreaking contributions to algebra, category theory, linguistics, theoretical physics, logic and proof theory... Jim Lambek's ideas keep inspiring upcoming generations of scholars."

(Festschrift on the occasion of Lambek's 90th birthday, 2014)

- Kosta Došen
- Djordje Čubrić
- Zoran Petrić
- Predrag Tanović
- S.G.
- Duško Pavlović

To celebrate his life

Jim Lambek was one of my postdoc supervisors.
He would ask a trivial question, and 3 months later my research would advance to the point where I would understand what he meant.
He made me understand that science is a conversation between people.
He would take me to a quiet sushi restaurant and teach me how to be rational, and how to use chopsticks.
Life makes sense because we meet such people.

Duško Pavlović

Seminar for Mathematical Linguistics during 1990s at McGill University, Montréal.

I have come to know Jim Lambek during my visit and work at the Seminar for Mathematical Linguistics in the spring of 1993 at McGill University, Montréal.

It was an inspiring project ran by Jim Lambek which linked between linguists and mathematicians (Ed Kennan, distinguished linguist from UCLA)

Overview

- This is an overview of Lambek's work on formal grammars for verb conjugation in several languages English, French, Latin, Turkish, Arabic, Hebrew and partly Serbian and Croatian.
- We will focus on the production grammar of conjugation of simple tenses in Serbian and Croatian, which is along the lines of Lambek's early work on Latin and French conjugation.
- His work was further developed and extended by Lambek and his co-authors to Turkish, Arabic and (Biblical) Hebrew.
- Lambek's approach was lately applied to Japanese.

Fundamental papers


J. Lambek,
A mathematician looks and French conjugation.
Theoretical Linguistics 2:203–214 (1975).


J. Lambek,
A mathematician looks and Latin conjugation.
Theoretical Linguistics 2/3:221–234 (1979).


S. Ghilezan,
Conjugation in SerboCroatian.
Linguistic Analysis 24:142–150 (1994).

The Verb - Simple Tenses

- Inflected languages are languages that changes the form or ending of some words when the way in which they are used in sentences changes.
- Latin, Greek, Sanskrit, Polish, Serbian, Croatian, Hungarian and Finnish are all highly inflected languages.
- English is weakly inflected

Lambek's method considers only simple tenses and disregards compound tenses.

Language	Inflected forms	Patterns	Simple tenses $\times$ Persons
Latin	90	3	5×6
French	42	1	7×6
Serbian	24	1(2)	4×6
Hebrew B	140	7	2×10
Spanish	54		

Lambek's production grammar

- Lambek's production grammar is a simple computational method for generating these conjugational forms step by step.

- The mathematical structure involved is:
  - *finitely generated partially ordered semi-group* or
  - *semi-Thue system* in mathematics
  - *rewriting system* in computer science
  - *production grammar* or
  - *Chomsky's Type zero language* in linguistics

Lambek's production grammar

- With each verb V, there is associated a $n \times m \times p$ matrix of conjugational verb-forms, $C_{ij}^{k}(V)$
  - $i = 1, ...n$ represents the (simple) tense
  - $j = 1, ..., m$ represents the person-number
  - $k = 1, ..., p$ represents the pattern

- A production grammar, in general, provides a method for calculating $C_{ij}^{k}(V)$ for a given (i, j, k, V).

Aspects - one pattern

Most verbs in Serbian and Croatian have two aspects:

- imperfective - the action is still in progress or it is being repeated
- perfective - the action has been completed, or that it is limited

Example. The two aspects of the verb *gledati*:

- *gledati* - "to look" (imperfective),
- *pogledati* - "to have a look" (perfective).

Interconnection:

- imperfective may become perfective by means of a prefix.
- perfective may become imperfective by:
  - changing the tone (4 tones not written)
  - dropping the prefix and prolonging and changing the root of the word.

One pattern for conjugation- both aspects use the same pattern.

Four groups of rules

1. Stem
2. Tense
3. Persone-Number
4. Morphographemic

1. The Stem

$Stem_1$ and $Stem_2$ are operations which assign to each verb its present and infinitive (aorist) stem.

- $Stem_1$ assigns to each verb the third person singular of the present tense
- $Stem_2$ of a verb is obtained from the first person singular of the aorist of the verb by leaving out the ending *oh* or *h* when preceded by another vowel.
- $Stem_3$, $Stem_4$.

	$Stem_1$	$Stem_2$	
gledati	gleda	gleda	(to look)
zvati	zove	zva	(to call)
voleti	voli	vole	(to love)

2. The Tense Signs

Each aspect has the following simple tenses:

T_1=present tense,

T_2=aorist,

T_3=imperfect,

T_4=future.

:

In order to obtain the infix corresponding to the appropriate tense we stipulate the following 4 rules (2a-2d):

2a. $T_1 \rightarrow \emptyset$,

2b. $T_2 \rightarrow \begin{cases} os_1, & \textit{after a consonant}, \\ s_2, & \textit{otherwise} \end{cases}$

2c. $T_3 \rightarrow \begin{cases} *\text{as}, & \textit{after a}, \\ *j\bar{a}s, & \textit{after ne, je and i}, \\ *ij\bar{a}s, & \textit{otherwise} \end{cases}$

2d. $T_4 \rightarrow \dagger\acute{c}$.

3. The Person-Number Signs

We present the six persons as follows:

Singular	Plural
P_1 = first person	P_4 = first person
P_2 = second person	P_5 = second person
P_3 = third person	P_6 = third person

In order to obtain the sufix corresponding to the appropriate person-number we stipulate the following 6 rules (3a-3f):

$$\text{3a. } P_1 \to \begin{cases} m, & \textit{after a vowel}, \\ u, & \textit{after ć or g}, \\ \dagger h, & \textit{otherwise}. \end{cases}$$

$$\text{3b. } P_2 \to \begin{cases} \check{s}, & \textit{after a vowel}, \\ {}^{+}e\check{s}, & \textit{after ć or g}, \\ \oplus(e), & \textit{otherwise}. \end{cases}$$

$$\text{3c. } P_3 \to \begin{cases} \emptyset, & \textit{after a vowel}, \\ \oplus(e), & \textit{otherwise}. \end{cases}$$

$$\text{3d. } P_4 \to \begin{cases} {}^{+}emo, & \textit{after ć or g}, \\ mo, & \textit{otherwise}. \end{cases}$$

$$\text{3e. } P_5 \to \begin{cases} {}^{+}ete, & \textit{after ć or g}, \\ te, & \textit{otherwise}. \end{cases}$$

$$\text{3f. } P_6 \to \begin{cases} *u, & \textit{after a, e, s or g}, \\ *e, & \textit{otherwise} \end{cases}$$

4. Morphographemic Rules

In order to take into consideration the morphographemic aspect of the language needed for generation, we stipulate the following 4 groups of rules (4a-4d):

4b. The second group of rules:
$s^{+}e \rightarrow še$,
$č^{+}u \rightarrow ku$,
$k^{+}e \rightarrow če$, $g^{+}e \rightarrow že$.

Verb Forms and One Word Sentences

The form C_{ij} of a verb can be generated by the following rule:

$$C_{ij}(V) \to Stem_i(V)\, T_i P_j.$$

We proceed from left to right by the application of rules.

As in Latin, each conjugational verb form can be regarded as one-word sentence: "plivam" means "I swim".

These sentences S are generated in the following way:

$$S \to C_{ij}(V).$$

This rule converts a sentence S into a verb form via $Stem_i(V)\, T_i P_j$.

$$
\begin{array}{lcll}
C_{16}(brinuti) & \rightarrow & Stem_1(brinuti)\,T_1P_6 & \\
 & \rightarrow & brine\,T_1P_6 & (1a) \\
 & \rightarrow & brine P_6 & (2a) \\
 & \rightarrow & brine * u & (3f) \\
 & \rightarrow & brinu & (4a)
\end{array}
$$

Implementation

- A **Prolog** implementation program of the present work has been developed by Miroslav Martinović.
- The main predicate (logic grammar symbol)

  conjugate (Inf, Tense, Per-Num, Form)

  Inf, *Tense*, *Per-Num* are given.
  Form is to be derived.
- A sort of lexicon needed for this program also contains '$Stem_1$' and '$Stem_2$' forms of the verbs. A version of this program is envisaged in which both of these operations under consideration will be implemented.

Further developments of Lambek's method


D. Bargelli, J. Lambek,
A Computational View of Turkish Conjugation
Linguistic Analysis 29:248–256 (1999).


D. Bargelli, J. Lambek,
A Computational Approach to Arabic Conjugation
Linguistic Analysis 30:1-22 (2001/2002).


C. Casadio, B. Coecke, M. Moortgat, P. Scott, P. (Eds.)
Categories and Types in Logic, Language, and Physics
Essays dedicated to Jim Lambek on the Occasion of his 90th Birthday.
Lecture Notes in Computer Science 8222 Subseries: Theoretical Computer Science and General Issues (2014).


Kumi Cardinal
An algebraic study of Japanese grammar
PhD thesis, McGill University (2002).