Computational considerations for the reconstruction of Proto-Slavic

Charles Zhang

September 18, 2021

1 Introduction

This paper is a case study of applying formal computational methods to aid the reconstruction of Proto-Slavic, specifically using the framework of the Reconstruction Engine [1]. The reconstruction of Slavic poses a set of novel challenges to such methods and I will investigate how well the meta-language of the program developed works to represent historical Slavic phenomena (such as the fall of yers and its effects on the law of open syllables in Common Slavic), how the program apparatus can be adapted or extended to deal with such challenges, and more broadly how well the neo-grammarian approach works for the reconstruction of Slavic, that is, whether sound laws seem to apply unexceptionally or precisely to what degree there appear to be non-'vertical' developments such as borrowing, dialect diffusion, and other wave-model-like phenomena that obscure the development of the daughter languages by means of regular sound law.

We chiefly focus on the phonological reconstruction of Proto-Slavic; the program implements the comparative method quite precisely using Czech, Polish, OCS, BCS, Slovenian, and Russian as the languages carrying the reflex forms, with the relevant sound correspondences and rules compiled to the best of the author's ability. The most noteworthy result of the phonological analysis is the encoding of the syllable canon and phonotactic structure of Common Slavic. While we currently ignore more difficult issues such as accent and syllable length which are more complex and difficult to represent formally for the purposes of the currently implemented algorithm's metalanguage, the version of the syllable canon used for Slavic in the program is a regular expression capable of expressing the polysyllabic, open syllable structure of Proto-Slavic and the distribution of strong and weak yers by Havlík's law, enabling non-trivial inferences and reconstructions by the program.

2 The data

We describe in this section the data-flow of the algorithm as applied to Slavic. There are various pieces of input data to the program. The lexical data is a set of 6 lexicons in the modern Slavic languages (Bosnian-Croatian-Serbian, Czech, Old Church Slavonic, Polish, Russian, Slovenian) cleaned and represented in a way most beneficial to diachronic analysis. The data consisting of the syllable canon and table of correspondences are what is used to perform diachronic analysis on the input data. The table of correspondences are a set of correspondences between sound forms of the daughter languages corresponding to a proto-sound form in a certain phonetic and phonotactic environment. The syllable canon specifies the syllable structure of the proto-language as a *regular expression*, and is also used to encode the phonotactic constraints which are necessary to guide the combinatorial reconstruction of the proto-language using the table of correspondences. For Slavic in particular, the syllable canon is useful to encode positional information which help for example in specifying phonotactic environments that determine the distribution of weak and strong yers as stipulated by Havlík's law, as well as codifying the law of open syllables in Common Slavic.

An excerpt of some of the rows of correspondences used are as follows:

ID	Context	Syllable Type	*	cz	pl	bcs	sl	rus	ocs
č		C	č	č	cz	č	č	č	č
d		C	d	d	d	d	d	d	d
d-front	_F	C	d	d	dzi	d	d	d,d′	d
d′		C	d′	z	dz	đ	j	ž	žd
dl-medial	V_V	CC	dl	dl	dl	l	l	l	l
ablaut-1-pl	_d,n,t,r,l,s	V	e	e	o,ó	e	e	e,é	e
g-soft	_F	C	g	h	gi	g	g	g,g′	g
g-v-soft	_F	CC	gv	hv	gwi	zv	zv	zv	zv
ę		V	ę	a, ě	ę, ą	e	e	ja,já	ę
ę̄		V	ę̄	á, í	ą	e	e	ja,já	ę
i		V	i	i,í	i	i	i	i,í	i
ĭ-strong		Y	ĭ	e	e	a	a,e	e,é	ĭ
ĭ-weak		y	ĭ	∅	∅	∅	∅	∅	ĭ
m		C	m	m	m	m	m	m	m
m-front	_F	C	m	m	mi	m	m	m	m
n		C	n	n	n	n	n	n	n
n-front	_F	C	n	n	ń, ni	n	n	n, n′	n
n′	_F	C	n′	ň	ń, ni	nj	nj	n′	n′
ŭ-strong		Y	ŭ	e	o,ó	a	a,e	o,ó	ŭ
ŭ-weak		y	ŭ	∅	∅	∅	∅	∅	ŭ
v		C	v	v	w	v	v	v	v
v-front	_F	C	v	v	wi,w	v	v	v,v′	v

The symbols F and V are *sound classes*, that is, they are a short-hand notation for specifying a ”natural” sound class. The algorithm has no actual phonetic knowledge and simply explicitly defines the classes in this way:

- V (vowel) = a, e, ě, ę, ę̄, i, ĭ, o, ǫ, ǭ, u, ŭ, y
- F (front) = ĭ, i, e, ě, ę, ę̄'
- S (soft) = l′, ř, č, ž, š, j

From the above excerpt of the table of correspondences we note the various representational considerations manifest in the lexical data and correspondences.

- Explicit soft signs are used in the transcription of Russian only when not implied by the following vowel.
- The acute accent ´ represents vowel length in Czech and stress in Russian. Neither of these features are very meaningfully distinguished in the table of correspondences besides the rows for *ę and *ę̄, meaning that these suprasegmental features do not factor much in the diachronic analysis. This is because I don't want to tackle the issue of trying to reconstruct Proto-Slavic accent quite yet in this paper.

- The representation of the lexical data is close to being orthography-based (using Latin transcription for Cyrillic-spelled languages); in general, standard orthography is phonemic enough in most Slavic languages for the purposes of the algorithm while also retaining useful premerger historical features (such as the distinction between *i* and *y* in Czech and the distinctions between *ż* and *rz* and between *ó* and *u* in Polish) that aid diachronic analysis. This reduces the need for transcription and makes it easier to obtain data for a uniform and standardized given variety.

The output data of the Reconstruction Engine is a list of cognate sets with their associated possible proto-form reconstructions together with the corresponding rows of correspondences used to determine those reconstructions.

An example of a cognate set that would be produced by the program, along with the correspondences used to derive the set, clearly representing the Proto-Slavic etyma for 'meat':

- **męso* `m-front ę s o`
  - bcs *meso* 'meat'
  - cz *maso* 'meat'
  - ocs *męso* 'meat'
  - pl *mięso* 'meat'
  - rus *mjáso* 'meat'
  - sl *meso* 'meat'

There is also a facility in the program to incorporate semantics to help filter out unlikely cognate sets. We ignore this facility for the purposes of the paper because we are mainly interested in the computational phonological challenges of the reconstruction of Proto-Slavic. Such a facility is also less important in Slavic than it is for mono-syllabic language groups like Tamangic [1], because there is less homophony and the phonological history of Slavic is such that relatively speaking there are not as many instances of phonologically distinct Proto-Slavic words becoming homophonous reflexes in the daughter languages. However the semantic component is still relevant, and there are a few instances in the Swadesh list data, especially with shorter forms, where the semantics helps pick apart cognate sets where vowel mergers have conflated two distinct etyma in the various daughter languages.

One important aspect of the Reconstruction Engine is that the algorithms used to produce the output from the inputs work purely on the abstract symbolic representations used in the input. That means that there is no knowledge of phonetics built into the program and also that the algorithms are not alignment-based, as is the case for most other programs. Instead, the Reconstruction Engine framework follows the neo-grammarian tradition and works off of abstract correspondences and rules which manipulate those symbols, where phonetics is used to help initially postulate the rules, but the meat of the comparative method executed by the program to get outputs from inputs is logical inference, a.k.a 'triangulation', to determine the best reconstruction from candidate reconstructions, for example, using the fact that reconstruction A is supported by a set of reflexes which is a strict superset of the set of reflexes supporting reconstruction B. Relevant for Slavic, the fact that the program is not alignment based means special care needs to be taken care when it comes to erosion of segments such as yers. Instead of aligning segments in the daughter languages, the program instead uses a combinatorial approach where a deletion rule means that every possible reconstruction that could have led to a daughter reflex is generated. That is, every position that could have held the deleted segment (subject to the phonotactic and contextual constraints) is a candidate for inserting the deleted sound and all combinations of those possible insertions are generated. While this is a big source of overgeneration in the sense that many different proto-forms are generated for a given set of cognates, this can be mitigated by the tighter specification of the phonological and phonotactic environment in which the deletion can occur in. This is how the program avoids any kind of alignment analysis which can be imprecise and complex.

3 Determining the strength of a yer: Implementation of Havlík's law

In this section we describe the mechanism the algorithm uses to constrain reconstructions with respect to the strength and position of yers proposed therein. We encode into the Proto-Slavic syllable canon Havlík's law, which is a rhythmic law describing the distribution of strong and weak yers in a Proto-Slavic word [5]. The syllable canon is implemented with the following *regular expression*:

```
V?((CC?C?V)|((CC?C?Y)?(CC?C?y)))+
```

where

- `C` = consonant
- `V` = full vowel
- `Y` = strong yer
- `y` = weak yer

This encodes the constraint that, counting backwards from the last yer in a word, the last yer being weak, the yers alternate in strength until a full vowel is reached, and then the pattern repeats with alternating weak and then strong yers. The law of open syllables is also manifest in this representation, since each possible syllable type in the regular expression is open and ends with either a full vowel, strong yer, or weak yer. The initial `V?` expression represents the fact that it is possible to have vowel initial words in Proto-Slavic; such as the word **ono* 'it'.

With the help of the syllable canon, the algorithm is able to constrain possible reconstructions to fit this phonotactic structure. Candidate reconstructions which have a strong final yer will be thrown out and not considered, for example. Because both weak front and weak back yers disappeared in the modern Slavic languages, the algorithm will try to insert a front or back yer in every possible position in a word when working from a modern reflex. Hence, limiting the possible positions a yer can appear in a reconstruction greatly reduces the amount of overgeneration of possible proto-forms.

4 Determining the quality of a yer

We describe the mechanism used for determining the quality of any proposed yers during reconstruction. While the phonotactic environment constrains the strength and position of yers in a potential proto-form, it leaves the quality of any proposed yers unspecified. For strong yers, the correspondence rules

ID	Context	Syllable Type	*	cz	pl	bcs	sl	rus	ocs
ĭ-strong		Y	ĭ	e	e	a	a,e	e,é	ĭ
ŭ-strong		Y	ŭ	e	o,ó	a	a,e	o,ó	ŭ

are enough to determine the quality of the yers as long as the daughter languages with distinct front and back reflexes support a given etymon. Otherwise, the algorithm will propose forms with both possible yer qualities. For example, given only the evidence of Czech and Croatian, which show little palatalization effects and do not show distinct reflexes for yers, Czech *pes* 'dog' and Croatian *pas* 'dog' will reconstruct to the possible forms *pĭsĭ*, *pĭsŭ*, *pŭsĭ*, and *pŭsŭ*. The weak yer must exist by Havlík's law, but the quality of it cannot be determined either.

Unlike with strong yers, the weak yers do not systematically show reflexes in any attested Slavic language besides OCS as part of the outgoing Common Slavic process of the loss of yers. However, the quality of a proposed weak yer can sometimes be deduced from the context (i.e. phonological environment) of the yer, so that effects such as palatalization on preceding consonants can determine the quality of a proposed yer, allowing the algorithm to make weak yer quality deductions without a supporting OCS form.

Altogether, this allows the algorithm to propose a single best proto-form for the etyma 'day' out of four possible proto-forms even without the support of OCS.

The word for 'day' in the daughter languages besides OCS are:

1. bcs *dan*

2. cz *den*

3. pl *dzień*

4. rus *den′*

5. sl *dan*

Then the following 'steps' of deduction are made simultaneously by the algorithm to deduce the quality of each yer:

1. The correspondence of the vowel reflexes imply a strong front yer in the first syllable.

2. The syllable canon (the law of open syllables and Havlík's law) implies there must be a final weak yer to make the previous yer strong.

3. The palatalization of *n* implies the final weak yer is a front yer.

Thus the algorithm arrives at the correct form **dĭnĭ* even with just the yerless daughter languages. Of course, the OCS word for 'day' *dĭnĭ* is in fact attested, but some lexical items present in other Slavic languages do not have an attested OCS cognate, hence the utility of these deductions.

The final set constructed by the algorithm:

- **dĭnĭ* `d-front ĭ-strong n-front ĭ-weak`
  - bcs *dan* 'day'
  - cz *den* 'day'
  - ocs *dĭnĭ* 'day' <— not necessarily needed for reconstruction
  - pl *dzień* 'day'
  - rus *den′* 'day'
  - sl *dan* 'day'

We see that the reconstruction of yers is double-faceted: The *phonotactic environment*, specified as a correspondence's 'syllable (canon) type', helps determine the possible positions and strengths of yers in a reconstruction, while the *phonological environment*, specified as a correspondence's 'context', helps determine the quality of any proposed yer, when possible. These two components form the basis of the computational phonology performed by the program, and is well-adapted to solving the problem of the reconstruction of yers in Proto-Slavic.

5 Results

5.1 General Observations

I make following qualitative judgements about the generated cognate sets:

- Encoding Havlík's law and effects of yers was relatively successful. The cases of overgeneration specifically to do with yers is controlled well by the syllable canon. Instances where multiple possibilities are generated in terms of yers seem to all be genuine ambiguities resulting from insufficient data necessary to construct the correct yer.

- The actual reconstructed cognate sets are overall pretty good; the lack of other deletion effects helped with the quality of the reconstructions, leading to cleaner cognate sets. The lack of phonological erosion and polysyllabic structure of Proto-Slavic reduced instances of semantically unrelated words merging in daughter languages, reducing the need to further sort and process cognate sets with semantics. The computational diachronic phonology tools at the disposal of the program (that is, a context-aware metalanguage for stating correspondences together with a syllable canon encoding phonotactics via regular expressions) are sufficiently equipped to handle enough of the phonological phenomena in Slavic to produce realistic cognate sets without much overgeneration.

- Lack of chronology/family grouping sometimes made accumulated sound changes hard to express and therefore reduced the quality of cognate sets; for example the various palatalization effects would be much more cleanly expressed with chronology. Polish palatalization assimilation in particular presents an issue, where in a form like *jeść* 'to eat' the palatalization on *ś* is secondary; without duplicating each correspondence row for *s* just to express a secondary palatalization effect in the environment of preceding a Proto-Slavic sound which corresponds to a palatalized development (primary or secondary), the program is unable to deduce that *jeść* is able to fall into the cognate set with the reconstruction **(j)ěsti*.

- OCS is very useful for yers; sets without the support of OCS suffered in terms of ambiguity, as expected. It is impossible to know the position of yers without the help of OCS in many cases.

5.2 The case of 'wing'

The algorithm gives the following two different cognate sets for the proto-etymon meaning 'wing':

- **křidlo* `k r-soft i dl-medial o`
    - bcs *krilo* 'wing'
    - cz *křidlo* 'wing'

  - ocs _krilo_ ’wing’
  - sl _krilo_ ’wing’

- _*krydlo_, _*krylo_ `k r y dl-medial/l-hard o`
  - bcs _krilo_ ’wing’
  - rus _kryló_ ’wing’
  - sl _krilo_ ’wing’

We observe that these forms are all cognate, but the algorithm refuses to create one cognate set that captures all the forms, because there is no sequence of correspondences that simultaneously explains _every_ reflex. Hence, because BCS, Russian, and Slovenian all do not show reflexes for medial _-dl-_ clusters, the algorithm has no choice but to reconstruct two potential proto-forms for the second cognate set. Only with the help of a West Slavic language would the algorithm be able to determine whether the cluster _dl_ or _l_ existed. The problem here is that the OCS and Czech reflexes are mutually exclusive with the Russian reflex in terms of which correspondences explain which proto-form. The problematic segment in question is _*i_ and _*y_.

The correspondences for _*i_ and _*y_ look like the following:

ID	Context	Syllable Type	*	cz	pl	bcs	sl	rus	ocs
i		V	i	i, í	i	i	i	i,í	i
y		V	y	y, ý	y	i	i	y	y

These correspondences explain the discrepancy; original _*i_ would reflect an _i_ vowel in Russian, whereas an _*y_ vowel would reflect an _y_ vowel in OCS and Czech (this is purely historical orthography in the case of Czech, as _y_ and _i_ now represent the same vowel, but note that the presence of _ř_ in the Czech form allows the ’vowel’ _i_ to be internally reconstructed anyway. This is another instance where the level of abstraction used in the textual representation to represent modern forms has somewhat of an impact on the diachronic analysis).

This represents a case where the supplied sound laws do not explain the whole story. The correct historical form in this case _is_ actually _*křidlo_ [3,4]. One suggested explanation is that in this specific instance the _y_ vowel actually appeared in Russian as an instance of _conflation_, where it is probable that Proto-Slavic noun _*krydlo_ meaning ’covering’ subsumed the similar sounding _*křidlo_. Other instances of non-neogrammarian changes can be explained by phenomena such as _analogical borrowing_ or _dialectal borrowing_ as well. We discussed another instance of this in class with the example of the alternation between _moloko - mlečnyj puť_ in Russian.

5.3 The case of ’fog’

Another interesting example the algorithm produces which warrants a second look by the linguist is the etyma for ’fog’ in Slavic. The program creates two cognate sets in this instance:

- _*mĭgĭla, mŭgĭla, mĭgŭla, mŭgŭla_ `m ĭ/ŭ-strong g ĭ/ŭ-weak l-hard a`
  - bcs _magla_ ’fog’
  - sl _magla_ ’fog’

- _*mĭgla_ `m ĭ-weak g l-hard a`
  - ocs _mĭgla_ ’fog’

  - pl *mgła* ’fog’

The first set seemingly shows a strong yer for the root vowel, as supported by the full vowel reflex present in the BCS and Slovenian forms, while the second set shows a weak yer reflex, as supported by the Polish form where the yer has disappeared and the OCS form. We would naturally like to propose that these forms derive from the same etyma, and so this discrepancy leads the linguist to investigate and provide an explanation. In this case, the proposed sound changes are incomplete; Browne states that in BCS, weak yers developed into *a* in some obstruent-sonorant environments [2]. Presumably a similar development occurred in Slovenian. This allows us to propose new correspondences for the development of yers that allows the algorithm to propose just a single set **mĭgla* supported by all of OCS, Polish, BCS and Slovenian.

After adding the additional correspondences we get a ’fuller’ cognate set:

- **mĭgla* `m ĭ-weak/son-obst g l-hard a`
  - bcs *magla* ’fog’
  - ocs *mĭgla* ’fog’
  - pl *mgła* ’fog’
  - sl *magla* ’fog’

Hence this shows the utility of such a program for reconstruction in that the linguist can more easily narrow down instances where the proposed sound changes or correspondences are incomplete in some way, then make a modification, and then recheck the modification, in addition to narrowing down instances where regular sound correspondences cannot explain the history of a word at all.

6 Conclusion

This investigation shows that computational methods such as this one is useful for historical linguists, even for well-studied languages like Slavic. Even though the program cannot itself come up with the rules (an exceptionally hard problem), the formal verification of those rules against the lexical data give the linguist more confidence in their rules and reconstructions, as well as giving the rapid ability of noticing exactly when the given sound correspondences are not exceptionless or underspecified as in the case of missing reflexes from a given cognate set in the output of the program. The computational representation of Havlík’s law is a computational encoding of a complex phonotactic phenomena that further demonstrates the expressiveness of regular expressions for the representation of syllable canons, and can be applied to other language families with similar prosodic phenomena.

This program was also originally developed to work on languages with only one or two syllable words, where the syllable canon can make many more explicit assumptions about the entire-word prosodic structure, because entire words are either one or two syllables. We find that it is definitely feasible to encode polysyllabic syllable canons and do full reconstructions with this computational framework.

7 Further work

There are still various aspects of this project that can be improved. While the encoding of Havlík’s law presented in this paper is a big step for representing the phonological reconstruction of Slavic computationally, more investigation is needed to tackle the problem of reconstructing Proto-Slavic accent, taking into account vowel length, stress, and even South Slavic tone, which may require machinery more sophisticated than regular expressions, such as a general framework for diachronic computational autosegmental phonology.

There was also no subgrouping hypothesis used in this reconstruction. We observed earlier that a way to represent chronology would have captured some phenomena more cleanly. Perhaps a reconstruction subgrouping West Slavic, East Slavic, and South Slavic would produce cleaner cognate sets, as that injections two steps of chronology into the analysis.

One other aspect not touched upon in this investigation is the morphological analysis of words in Slavic. Since most morphology, specifically morphological endings, are parallel in Slavic languages, there was not much of a need to isolate morphemes in the hope to detect more cognates. However, Slavic morphology is complex, and it would be interesting to also apply the program's methodology to propose cognate sets for individual morphemes or morphophonological processes.

7.1 References

[1] Lowe, John and Martine Mazaudon. 1995. ”The reconstruction engine: a computer implementation of the comparative method,” Association for Computational Linguistics Special Issue in Computational Phonology 20.3 (September 1994) pp. 381-417.
[2] Browne, Wayles. (1993). Serbo-Croat. In Bernard Comrie and Greville G. Corbett, eds., The Slavonic Languages. London & New York: Routledge, pp. 306–387.
[3] Derksen, Rick, Etymological Dictionary of the Slavic Inherited Lexicon (Leiden Indo-European Etymological Dictionary Series; 4), Leiden, Boston: Brill, 2008: “*krīdlò”, Entry page 247: “n. o (b) ‘wing’”
[4] Olander, Thomas, Common Slavic accentological word list, Copenhagen: Editiones Olander, 2001: “kridlo”
[5] Townsend, Charles and Laura Janda (1996). Common and Comparative Slavic Phonology and Inflection: Phonology and Inflection: With Special Attention to Russian, Polish, Czech, Serbo-Croatian, Bulgarian. Bloomington, USA: Slavica. ISBN 0-89357-264-0.