# The Match-Extend Serialization Algorithm in Multiprecedence

**Maxime Papillon**
Classic, Modern Languages and Linguistics, Concordia University
papillonmaxime@gmail.com

## Abstract

Raimy (1999; 2000a; 2000b) proposed a graphical formalism for modeling reduplication, originallymostly focused on phonological overapplication in a derivational framework. This framework is now known as Precedence-based phonology or Multiprecedence phonology. Raimy's idea is that the segments at the input to the phonology are not totally ordered by precedence. This paper tackles a challenge that arose with Raimy's work, the development of a deterministic serialization algorithm as part of the derivation of surface forms. The Match-Extend algorithm introduced here requires fewer assumptions and sticks tighter to the attested typology. The algorithm also contains no parameter or constraint specific to individual graphs or topologies, unlike previous proposals. Match-Extend requires nothing except knowing the last added set of links.

## 1 Introduction

This paper provides a general serialization algorithm for all morphological structures in all languages. The challenge of converting non-linear structures of linguistic representation into a format ready to be handled in production is one that matters to both morphosyntax and morphophonology. Reduplication is a phenomenon at the frontier of morphology and phonology that has drawn a lot of attention in the last few decades. Reduplication's non-concatenative nature and the fact that it manifests long-distance dependencies among segments set it apart from the 'standard' word-formation that most theories are designed to handle. These properties have often pushed theoreticians to propose expansive systems such as copying procedures on top of traditional linear segmental phonology to make the system powerful enough to handle these dependencies. The multiprecedence model expanded upon here builds on properties that are already implicit in all approaches to phonological representation, and actually gets rid of some standard assumptions. It accounts for attested patterns and predicts an unattested reduplication pattern to be impossible.

## 2 Multiprecedence

The theory of Multiprecedence seeks to account for reduplication representationally via loops in a graph. Eschewing correspondence statements and copying procedures, Multiprecedence treats reduplication as fundamentally a structural property created by the addition of an affix, whose serialization has the effect of pronouncing all or part of the form twice.

Consider a string like Fig. 1a, the standard way of representing the segments that constitute a phonological representation. An alternative way to encode that same information is in the form of a set of immediate precedence statements like Fig. 1b. For legibility the set of pairs in Fig. 1b can be represented in the form of a graph. Adding the convention that of using # and % for the START and END symbols respectively we get the picture in Fig. 1c. In general I will refer to this as the graph representation.

a.kæt
b. { ⟨ START, k ⟩, ⟨ k,æ ⟩,⟨ æ,t ⟩,⟨ t,END ⟩}
c. # → k → æ→ t → %

Figure 1: Phonological representations of the word *cat* as a string (a), ordered pairs (b), and a graph (c).

The graph representation should highlight an important detail. There is no a priori logical reason in this representation why forms should be linear, with one segment following another in a chain. This is only an assumption that we impose on the structure when assuming strings. This assumption is what Multiprecedence abandons. Multiprecedence proposes that asymmetry and irreflexivity are not relevant to phonology. A segment can precede or follow multiple segment, two segments can transitively precede each other, and a segment can precede itself. A valid multiprecedence graph is not restricted by topology, a term a will use for the pattern of the graph independent from the content of the nodes.

Using this view of precedence, affixation is the process of combining the graph representations of different morphemes. A word is a graph consisting of the edges and vertices (precedence relations and segments) of one or more morphemes. An example of the suffixation of the English plural is shown in Fig. 2a, and the infixation of the Atayal animate focus morpheme is given in Fig. 2b. Full root reduplication, which expresses the plural of nouns in Indonesian is shown in Fig. 2c. There are two things to notice in Fig. 2c. First, that a precedence arrow is added, without any segmental material: the reduplicative morpheme consists of just that arrow. Second, although Fig. 2a and Fig. 2b each offer two paths from the START to the END of the graph, Fig. 2c contains a loop that offers an infinite number of paths from START to End. The representation itself does not enforce how many times the arrow added by the plural morpheme should be traversed. All three of these structures have to be handled by a *serialization* algorithm in order to be actualized by the phonetic motor system, which selects a path through the graph to be sent to the articulators. A correct serialization algorithm must be able to select the correct of the two paths in Fig. 2a and Fig. 2b and the path going through the back loop only once in Fig. 2c.

I will assume here that these forms are constructed by the attachment of an affix morpheme onto a stem as in Fig. 3. English speakers have a graph as a lexical item for the plural as in Fig. 3a and a lexical item for CAT as in Fig. 3b, which combine as in Fig. 3c. The moniker “last segment” is an informal way to refer to that part of the affix that is responsible for attaching it to the stem in the right location. This piece of the plural affix will attach onto the last segment, the one preceding the end of the word %, of what it combined with, and onto %, yielding Fig 3c. Similarly the Atayal form in Fig. 2 is built from a root *#hŋuʔ%* ‘soak’ and an infix *-m-* marking the animate actor focus and attaching between the first and the second segment. For details on the mechanics of attachment see Raimy (2000a, §3.2), Samuels (2009, p.177-87), and Papillon (2020, §2.2). It suffices here to say that at vocabulary insertion an affix can target certain segments of the stem for attachment. Raimy (2000a) shows how this representation can generate the reduplicative patterns from numerous languages as well as account for such phenomena as over- and under-application of phonological processes in reduplication.

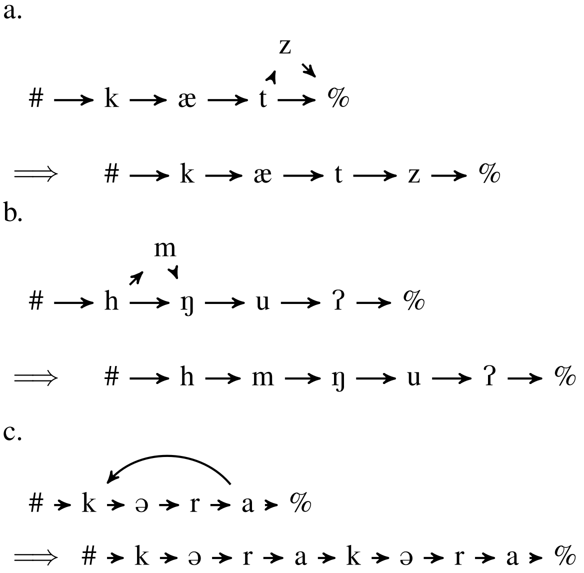

Figure 2: Affixation in Multiprecedence. Suffixation (a), infixation (b), reduplication (c).

a. [last segment] → z → %
b. # → k → æ→ t → %
c.

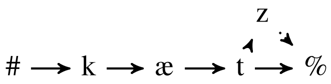

Figure 3: Affix (a) and root (b) combined in (c).

Given the assumption that a non-linear graph cannot be pronounced, phonology requires an algorithm capable of converting graph representations into strings like in Fig. 2. Two main families of algorithms have been proposed. Raimy (1999) proposed a stack-based algorithm which was expanded upon by Idsardi and Shorey (2007) and McClory and Raimy (2007). This algorithm traverses the graph from # to % by accessing the stack. This idea suffers the problem of requiring parameters on individual arcs. Every morphologically-added precedence link must be parametrized as to its priority determining whether it goes to the top or the bottom of the stack. This is necessary in this system because when a given arc is traversed is not predictable on the basis of when it is encountered in a traversal. This parametrization radically explodes the range of patterns predicted to be possible much beyond what is attested. Fitzpatrick and Nevins (2002; 2004) proposed a different constraint-base algorithm which globally compares paths through the graph for completeness and economy but suffers the problem of requiring ad hoc constraints targeting individual types of graphs, lacking generality. In the rest of this article I will present a new algorithm which lacks any parameter and whose two operations are generic and not geared towards any specific configuration.

## 3 The Match-Extend algorithm

This section will present the Match-Extend algorithm and follow up with a demonstration of its operation on various attested Multiprecedence topologies.

The input to the algorithm is the set of pairs of segments corresponding to the pairs of segments in immediate precedence relation without the affix, e.g. {#k,kæ,æt,t%} for the English stem *kæt*, and the set of pair of segments corresponding to the precedence links added by the affix, e.g. {tz,z%} when the plural is added.

Intuitively the algorithm starts from the morphologically added links and extends outwards by following the precedence links in the StemSet, the set of all precedence links in the stem to which the morpheme is being added. If there is more than one morphologically added link, they all extend in parallel and collapse together if one string ends in one or more segment and the other begins with the same segment or segments. A working version of this algorithm coded in Python will be included as supplementary material.

### 3.1 Match-Extend in action

Consider first total reduplication as in Fig. 2c above. Fig. 3.1 shows the full derivation of *kəra-kəra* with total reduplication. As there is only one morphologically-added link, no Match step will happen.

Let us turn to more complex graphs discussed in the literature. Raimy discusses a process of CV reduplication in Tohono O'odham involving reduplicated pattern such as *babað* to *ba-b-bað*, and *čipkan* to *či-čpkan* requiring graphs as in Raimy (2000a, p.114). Although there are multiple plausible paths through this graph, only one is attested and this path requires traversing the graph by following the backlink before the front-link, even though the front-link would be encountered first in a traversal.

1 .The precedence links of the stem begin in a set StemSet.
2. The morphologically added links begin in a set WorkSpace.
3. Whenever two strings in the WorkSpace match such that the end of one string is identical to the end of the other, the operation Match collapses the two into one string such that the shared part appears once. E.g. `abcd` and `cdef` to `abcdef`. A Match along multiple characters is done first.
4. When there is no match within the WorkSpace, the operation Extend simultaneously lengthens all strings in the WorkSpace to the right and left using matching precedence links of the stem. StemSet remains unchanged.
5.Steps 3 and 4 are repeated until # and % have been reached by Extend and there is a single string in the WorkSpace.

**Algorithm 1:** The Match-Extend Algorithm (informal version).

| StemSet | { #k , kə , ər , ra , a%} |
|---|---|

| WorkSpace | ak |
|---|---|
| Extend | rakə |
| Extend | ərakər |
| Extend | kərakəra |
| Extend | #kərakəra% |

Figure 4: Match-Extend derivation of *kəra-kəra*.

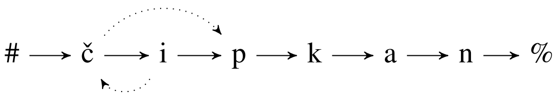

Figure 5: Tohono O'odham *či-čpkan*.

The match-Extend algorithm will correctly derive the correct form as shown in . Right away the strings `ič` and `čp` match, as one starts with the node c and the other ends with the same node. The two are collapsed as ičp and then keep extending.

| StemSet | {#č, či, ip, pk, ka, an, n%} |
|---|---|
| WorkSpace | čp<br>ič |
| Match | ič̲p |
| Extend | čičpk |
| Extend | #čičpka |
| Extend | #čičpkan |
| Extend | #čičpkan% |

Figure 6: Match-extend derivation of Tohono O'odham *čičpkan*.

A similarly complex graph is needed in Nancowry. Raimy (2000a, p.81) discusses examples like Nancowry reduplication of the last consonant toward the beginning of the word, e.g. *sut* 'to rub' to *ʔit-sut* which requires a graph as in Fig. 7. However here the opposite order of traversal must be followed, not skipping the first forward link. I assume here, like Raimy, that the glottal stop is epenthetic and added after serialization. Here, not taking the first link would also result in the wrong output [*sutsut]. So this form requires the first morphologically-added link to be taken to produce the correct form.

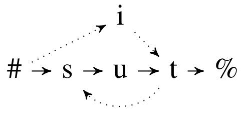

Figure 7: Nancowry *ʔit-sut*.

Again Match-Extend will serialize Fig. 7 without any further parameter as in Fig. 8. The three strings #i, it, and ts can match right away into a single string #its which will keep extending.

As these examples illustrate, Match-Extend does not need to be specified with look-ahead, global considerations, or graph-by-graph specifications of serialization to derive the attested serialization of graphs like Fig. 5 or Fig. 7. The serialization starts in parallel from two added links that extend until they reach each other in the middle, and this will work regardless of the order in which 'backward' and 'forward' arcs are located. They will meet in one direction and serialize in this order.

| StemSet | {#s, su, ut, t%} |
|---|---|
| WorkSpace | #i<br>it<br>ts |
| Match | #i̲t<br>ts |
| Match | #it̲s |
| Extend | #itsu |
| Extend | #itsut |
| Extend | #itsut% |

Figure 8: Derivation of Nancowry *ʔitsut*.

Another interesting topology is found in the analysis of Lushotseed. Fitzpatrick & Nevins (2002; 2004) observed that in cases where multiple reduplication processes of different size happen to the same form, with multiple morphologically added arrows forking away from the same segment, these graphs are seemingly universally serialized such that they follow the shorter arc first. They discuss Lushotseed forms with both distributive and Out-Of-Control (OOC) reduplication. They argue on the basis of the fact that in either scope order the form is serialized in the same way, suggesting that they are serialized simultaneously. This implies forms like *$g^w$ad*, 'talk', surfacing in the distributive OOC or the OOC distributive as *$g^w$ad-ad-$g^w$ad*, requiring a graphs like Fig. 9.

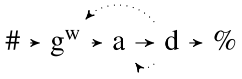

Figure 9: Lushotseed *$g^w$ad-ad-$g^w$ad*.

Fitzpatrick & Nevins (2002; 2004) proposed an ad hoc constraint to handle this type of scenario, the constraint SHORTEST, enforcing serializations that follow the shorter arrow first. But Match-Extend derives the attested pattern without any further assumptions. Consider the derivation of the Lushotseed form in Fig. 9. After one Extend step, the two strings adg$^w$a and adad match along the nodes ad. You might notice that the two strings also match in the other order with the node a, so we must assume the reasonable principle that in case of multiple matches, the best match, meaning the match along more nodes, is chosen. From that point on adadg$^w$a extends into the desired form.

It is somewhat intuitive to see why this works: because Match-Extend applies one step of Extend at a time and must Match and collapse separate

| StemSet | { #$g^w$, $g^w$a, ad, d%} |
|---|---|
| WorkSpace | d$g^w$<br>da |
| Extend | ad$g^w$a<br>adad |
| Match | ad$\underline{\text{adg}}^{w}$a |
| Extend | $g^w$adad$g^w$ad |
| Extend | #$g^w$adad$g^w$ad% |

Figure 10: Derivation of Lushotseed *$g^w$adad$g^w$ad.*

strings from the WorkSpace immediately when a Match is found, two arcs added by the morphology will necessarily match in the direction in which they are the closest. The end of the d→a arc is closer to the beginning of the d→$g^w$ one than vice-versa, and hence the two will join in this direction and therefore surface in this order. This can be generalized as Fig. 11.

- If the graph contains two morphologically added links $\alpha \rightarrow \beta$ and $\gamma \rightarrow \delta$, and
  - ▶ There is a unique path X from $\beta$ to $\gamma$ not going through $\alpha \rightarrow \beta$ or $\gamma \rightarrow \delta$, and
  - ▶ There is a unique path Y from $\delta$ to $\alpha$ not going through $\alpha \rightarrow \beta$ or $\gamma \rightarrow \delta$,
- Then the Match-Extend algorithm will output a string containing:
  - ▶ ...$\alpha\beta$...$\gamma\delta$... if X is shorter than Y
  - ▶ ...$\gamma\delta$...$\alpha\beta$... if Y is shorter than X

Figure 11: Closest Attachment in Match-Extend.

Note that this is not a new assumption: this is a theorem of the model derivable from the way Match and Extend interact with multiple morphologically added arcs. This can allow us to work out some serializations without having to do the whole derivation.

Consider for instance the Nlaka'pamuctsin distributive+diminutive double reduplication, e.g. *sil*, 'calico', to *sil-si-sil*, (Broselow, 1983). This pattern requires the Multiprecedence graph to look as in Fig. 12.

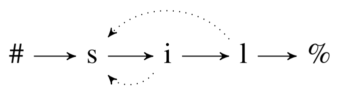

Figure 12: Nlaka'pamuctsin *sil-si-sil*.

The graph in Fig. 12 is simply the transpose graph of a graph where SHORTEST would apply like Fig. 9, but it does not actually fit the pattern of SHORTEST as its two 'backward' arrows do not start from the same node. In fact if anything SHORTEST would predict the wrong surface form, as **si-sil-sil* would be the form if the shorter path were taken first. In Match-Extend and Closest Attachment Fig. 11 the prediction is clear: it is predicted to serialize as *sil-si-sil* because the path from l→s to i→s is shorter than the path from i→s to l→s, thus deriving the correct string.

Fitzpatrick and Nevins (2002) report some forms with graphs like Fig. 12 which must be linearized in ways that would contradict Match-Extend, such as *sax$^w$* to *sa-sax$^w$-sax$^w$* in Lusotsheed Diminutive+Distributive forms. But contrary to the Distributive+OOC forms discussed earlier there is no independent evidence here for the two reduplications being serialized together. I therefore assume that those instances consist of two separate cycles, serialized one at a time: *sax$^w$* to *sax$^w$-sax$^w$* to *sa-sax$^w$-sax$^w$*. Match-Extend therefore relies on cyclicity, with the graph built up through affixation and serialized multiple times over the course of the derivation.

### 3.2 Non-Edge Fixed Segmentism

Fixed segmentism refers to cases of reduplication where a segment of one copy is overridden by one or more fixed segments. A well known English example is *schm*-reduplication like *table* to *table-schmable* where *schm*-replaces the initial onset. I will call Non-Edge Fixed Segmentism (NEFS) the special case of fixed segmentism where the fixed segment is not at the edge of one of the copies. These are the examples where the graph needed is like Fig. 13 or Fig. 14.

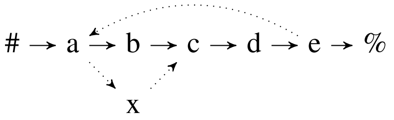

Figure 13: NEFS 'early' in the copy.

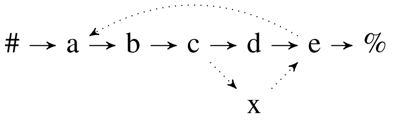

Figure 14: NEFS 'late' in the copy.

Closest Attachment in Match-Extend predicts that if a fixed-segment is added towards the beginning of the form, it should surface in the second copy, and if it is added toward the end of the form, it should surface in the first copy. Or in other words the fixed segment will always occur in the copy such that the fixed segment is closer to the juncture of the two copies. The graph in Fig. 13 will serialize as `abcde-a`**`x`**`cde` and the graph in Fig. 14 will serialize as `abc`**`x`**`e-abcde`. This follows from the properties of Match and Extend: as the precedence pairs of the overwriting segment and the precedence pair of the backward link extend outward, it will either reach the left or right side first and this will determine the order in which they appear in the final serialized form.

This prediction is borne out by many examples of productive patterns of reduplication with NEFS such as Marathi *saman-s**u**man* (Alderete et al., 1999, citing Apte 1968), Bengali *sajɹa-s**u**jɹa* (Khan, 2006, p.14), Kinnauri migo-məgo (Chang, 2007).

Apparent counterexamples exist, but have other plausible analyses. A major one worth discussing briefly is the previous multiprecedence analysis of the Javanese Habitual-Repetitive as described by Yip (1995; 1998). Most forms surface with a fixed /a/ in the first copy as in *el**a**ŋ-eliŋ* ‘remember’. This requires a graph such as Fig. 15 which serializes in comformity with Match-Extend.

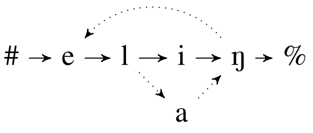

Figure 15: Javanese *el**a**ŋ-eliŋ*.

However when the first copy already contains /a/ as the second vowel the form is realized with /e/ in the second copy as *udan-ud**e**n* ‘rain’. Idsardi and Shorey (2007) and McClory and Raimy (2007) have analyzed this as a phonologically-conditioned allomorph with fixed segment /e/ that must be serialized differently from the /a/ allomorph, with the overwriting vowel in the second copy, i.e. a graph such as Fig. 16 that does not serialize in comformity with Match-Extend. Idsardi and Shorey (2007) and McClory and Raimy (2007) use this example to argue for a system of stacks that serialization must read from the top-down. Precedence arcs in turn can be lexically parametrized as to whether they are added on top or at the bottom of the stack upon affixation, thus deriving *elaŋeliŋ* from the /a/ allomorph being on top of the stack and traversed early and *udanuden* from the /e/ allomorph being at the bottom of the stack and traversed late. This freedom of lexical specification grants their system the power to enforce any order needed, including the capacity to handle the ‘look-ahead’ and ‘shortest’ cases above in terms of full lexical specification. They could also easily handle languages with the equivalent of a LONGEST constraint. This model is less predictive while also being more complex.

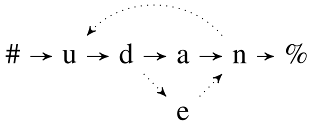

Figure 16: Javanese *udan-uden* according to Idsardi and Shorey (2007) and McClory and Raimy (2007).

But this complexity is unneeded if we instead adopt dissimilation analysis closer in spirit to Yip’s original Optimality-Theory analysis. We can say that the /a/ of the first copy is an overwritten /a/ in both *elaŋ-eliŋ* and in *udan-uden* and a phonological process causes dissimilation of the root /a/ in the presence of the added /a/. In Optimality-Theory this requires an appeal to the Obligatory Contour Principle operating between the two copies, but in Multiprecedence the dissimilation is even simpler to state because the two /a/’s are very local in the graph. We simply need a rule to the effect of raising a stem /a/ in the context of a morphologically-added /a/ that precedes the same segment as in Fig.17.

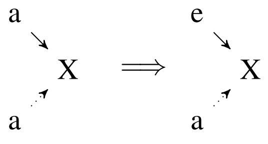

Figure 17: Dissimilation Rule

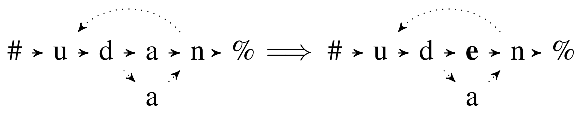

Figure 18: Derivation of *udan-uden*.

There is therefore no need to abandon Match-

Extend on the basis of Javanese.

Consider another apparent counterexample to the prediction: the Palauan root /rɛbət$^h$/ forms its distributive with CVCV reduplication and the verbal prefix *mə-* forming *mə-rəbə-rɛbət$^h$* (Zuraw, 2003). At first blush, one may be tempted to see the first schwa of the first copy as overwriting the root's /ɛ/. But the presence of this schwa actually follows from the independently-motivated phonology of Palauan in which all non-stressed vowels go to [ə]. This thus is the result of a phonological rule applying after serialization about which Match-Extend has nothing to say.

Relatedly, other apparent issues may be caused by interactions with phonology. D'souza (1991, p.294) describes how echo-formation in some Munda languages is accomplished by replacing all the vowels in the second copy with a fixed vowel, e.g. Gorum *bubuʔ* 'snake' > *bubuʔ-bibiʔ*. Fixed segmentism of each vowel individually may not be the best analysis of these forms, there may instead be a single fixed segment and a separate process of vowel harmony or something along those lines. This type of complex interaction of non-local phonology with reduplication has been investigated before in Multiprecedence, e.g. the analyses of Tuvan vowel harmony in reduplicated forms in Harrison and Raimy (2004) and Papillon (2020, §7.1), but these analyses make extra assumptions about possible Multiprecedence structures that go far beyond the basics explored here. The subject requires further exploration, but appears to be more of an issue of phonology and representation than of serialization per se.

Apparent counterexamples will have to be approached on a case-by case basis, but I have not identified many problematic examples so far that did not turn out to be errors of analysis.[1]

We have therefore seen that Match-Extend can straightforwardly account for a number of attested complex reduplicative patterns without any special stipulations. More interestingly Match-Extend makes strong novel predictions about the location of fixed segments. I have not been able to locate many examples of NEFS in the literature. For example the typology of fixed segmentism in Alderete et al. (1999) does not contain any example of NEFS. This will require further empirical research.

## 4 One limitation of Match-Extend: overly symmetrical graphs

There is a gap in the predictions of Fig. 11: Closest Attachment predicts that morphologically-added edges will attach in the order they are the closest, which relies on an asymmetry in the form such that morphologically-added links are closer in one order than the other. This leaves the problem of symmetrical forms like Fig. 19. The former of there was posited in the analysis of Semai continuative reduplication by Raimy (2000a, p.146-47) for forms like *dŋɔh* 'appearance of nodding' to *dh-dŋɔh*; the latter would be needed in various languages reduplicating CVC forms with vowel changes such as the Takelma aorist described in Sapir (1922, p.58) like *t'eu* 'play shinny' to *t'eut'au*.

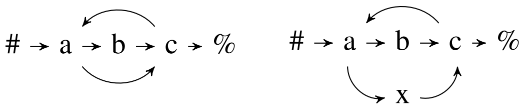

Figure 19: Two structures overly symmetrical for Match-Extend.

These are the forms which, in the course of Match-Extend, will come to a point where Match is indeterminate because two strings could match equally well in either direction. For example the WorkSpace of the first of these structures will start with `ac` and `ca`, which can match either as `a`**`c`**`a` or `c`**`a**`c`. The former would extend into `#acabc%` and the latter into `#abcac%`. Match-Extend as stated so far is therefore indeterminate with regard to these symmetrical forms.

This is not an insurmountable problem for Match-Extend. To the contrary this is a problem

[1] One such apparent counter-example is worth briefly commenting on here due to its being mentioned in well-known surveys of reduplication. This alleged reduplication is from in Macdonald and Darjowidjojo (1967, p.54) and repeated in Rubino (2005, p.16): Indonesian *belat* 'screen' to *belat-belit* 'underhanded'. If correct this example would be a counterexample to Match-Extend, as a fixed /i/ must surface in the second copy. However this pair seems to be misidentified. The English-Indonesian bilingual dictionary by (Stevens and Schmidgall-Tellings, 2004) lists a word *belit* meaning 'crooked, cunning, deceitful, dishonest, underhanded', which semantically seems like a more plausible source for the reduplicated form *belat-belit* and fits the predictions of Match-Extend. The same dictionary's entry under *belat* lists some screen-related entries and then *belat-belit* as meaning 'crooked, devious, artful, cunning, insincere' cross-referencing to *belit* as the base. I conclude that this example was misidentified by previous authors and is unproblematic for Match-Extend.

of having too many solutions without a way to decide between them, none of which require adding parametrization to Match-Extend. Maybe symmetrical forms crash the derivation and all apparent instances in the literature must contain some hidden asymmetry. It is worth noting that the pattern in Fig. 19 attested in Semai has a close cognate in Temiar, but in this language the symmetrical structure is only obtained for simple onsets, *kɔw* ‘call’ to *kw-kɔw*, but *slɔg* ‘sleep with’ to *s-g-lɔg* (Raimy, 2000a, p.146). This asymmetry resolves the Match-Extend derivation. It may simply be the case that the forms that look symmetrical have a hidden asymmetry in the form of silent segments. For example if the root has an X at the start as in Fig. 21. This is obviously very ad hoc and powerful so minimally we should seek language-internal evidence for such a segment before jumping to conclusions.

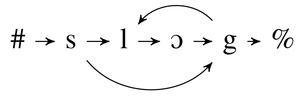

Figure 20: Temiar *sglog*.

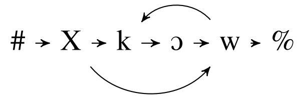

Figure 21: Semai *kw-kɔw* with hidden asymmetry in the form of a segment X without a phonetic correlate, which breaks the symmetry.

Alternatively it could be that symmetrical forms lead to both options being constructed and this optionality is resolved in extra-grammatical ways. I will leave this hole in the theory open, as a problem to be resolved through further research.

## 5 Conclusion

This article presents an invariant serialization algorithm for all morphological patterns in Multiprecedence.

The Multiprecedence research program has been fruitful in bringing various non-concatenative phenomena other than reduplication within the scope of a derivational item-and-arrangement model of morphology, including e.g. subtractive morphology (Gagnon and Piché, 2007), Semitic templatic morphology (Raimy, 2007), and vowel harmony, word tone, and allomorphy (Papillon, 2020). A serialization algorithm capable of handling these structures is crucial for the completeness of the theory.

As pointed out by a reviewer, it is crucial to develop a a typology of the possible attested graphical input structures to the algorithm so as to properly characterize and formalize the algorithm needed. In every form discussed here the roots is implicitly assumed to be underlyingly linear and affixes alone add some topological variety to the graphs, as is mostly the case in all the forms from (Raimy, 1999; Raimy, 2000a). Elsewhere I have challenged this idea by positing parallel structures both underlyingly and in the output of phonology (Papillon, 2020). If these structures are allowed in Multiprecedence Phonology then Match-Extend will need to be amended or enhanced to handle more varied structures.

In this paper I proposed a model that departs from the previous ones in being framed as patching a path from the morphology-added links towards # and % from the inside-out, as opposed to the existing models seeking to give a set of instructions to correctly traverse the graph from # to % from beginning to the end.

## References

John Alderete, Jill Beckman, Laura Benua, Amalia Gnanadesikan, John McCarthy, and Suzanne Urbanczyk. 1999. Reduplication with fixed segmentism. *Linguistic inquiry*, 30(3):327–364.

Ellen I Broselow. 1983. Salish double reduplications: subjacency in morphology. *Natural Language & Linguistic Theory*, 1(3):317–346.

Charles B Chang. 2007. Accounting for the phonology of reduplication. Presentation at LASSO XXXVI. Available on `https://cbchang.com/curriculum-vitae/presentations/`.

Jean D’souza. 1991. Echos in a sociolinguistic area. *Language sciences*, 13(2):289–299.

Justin Fitzpatrick and Andrew Nevins. 2002. Phonological occurrences: Relations and copying. In *Second North American Phonology Conference, Montreal*.

Justin Fitzpatrick and Andrew Nevins. 2004. Linearization of nested and overlapping precedence in multiple reduplication.

Michaël Gagnon and Maxime Piché. 2007. Principles of linearization & subtractive morphology. In *CUNY Phonology Forum Conference on Precedence Relations*.

K. David Harrison and Eric Raimy. 2004. Reduplication in Tuvan: Exponence, readjustment and phonology. In *Proceedings of Workshop in Altaic Formal Linguistics*, volume 1. Citeseer.

William Idsardi and Rachel Shorey. 2007. Unwinding morphology. In *CUNY Phonology Forum Workshop on Precedence Relations*.

SD Khan. 2006. *Similarity Avoidance in Bengali Fixed-Segment Reduplication*. Ph.D. thesis, University of California.

Roderick Ross Macdonald and Soenjono Darjowidjojo. 1967. *A student's reference grammar of modern formal Indonesian*. Georgetown University Press.

Daniel McClory and Eric Raimy. 2007. Enhanced edges: morphological influence on linearization. In *Poster presented at The 81st Annual Meeting of the Linguistics Society of America. Anaheim, CA*.

Maxime Papillon. 2020. *Precedence and the Lack Thereof: Precedence-Relation-Oriented Phonology*. Ph.D. thesis. `https://drum.lib.umd.edu/handle/1903/26391`.

Eric Raimy. 1999. *Representing reduplication*. Ph.D. thesis, University of Delaware.

Eric Raimy. 2000a. *The phonology and morphology of reduplication*, volume 52. Walter de Gruyter.

Eric Raimy. 2000b. Remarks on backcopying. *Linguistic Inquiry*, 31(3):541–552.

Eric Raimy. 2007. Precedence theory, root and template morphology, priming effects and the structure of the lexicon. CUNY Phonology Symposium Precedence Conference, January.

Carl Rubino. 2005. Reduplication: Form, function and distribution. *Studies on reduplication*, pages 11–29.

Bridget Samuels. 2009. *The structure of phonological theory*. Harvard University.

Edward Sapir. 1922. Takelma. In Franz Boas, editor, *Handbook of American Indian Languages,*. Smithsonian Institution. Bureau of American Ethnology.

Alan M. Stevens and A. Ed. Schmidgall-Tellings. 2004. *A comprehensive Indonesian-English dictionary*. PT Mizan Publika.

Moira Yip. 1995. Repetition and its avoidance: The case in Javanese.

Moira Yip. 1998. Identity avoidance in phonology and morphology. In *Morphology and its Relation to Phonology and Syntax*, pages 216–246. CSLI Publications.

Kie Zuraw. 2003. Vowel reduction in Palauan reduplicants. In *Proceedings of the 8th Annual Meeting of the Austronesian Formal Linguistics Association [AFLA 8]*, pages 385–398.