Extending Finite-state Models of Reduplication to Tone in Thai

Casey D. Miller
Dept. of Linguistics
University of Utah
u1337847@utah.edu

Aniello De Santo
Dept. of Linguistics
University of Utah
aniello.desanto@utah.edu

Abstract

Languages exhibiting both tonal and reduplication processes pose a challenge for finite-state technologies. In this sense, Markowska et al. (2021) propose a combination of 2-way FSTs and multi-tape FSTs in order to simultaneously deal with total reduplication on the segmental level and independent tonal processes on the autosegmental level. Here, we evaluate this model for reduplication processes in Thai, which shows total reduplication both for tones and segments, and we suggest that the expressivity of 2-way FSTs is needed at both levels.

1 Introduction

Reduplication, the systematic copying/repetition of linguistic content to function with some new grammatical purpose, is a well-attested phenomenon cross-linguistically (Hurch and Mattes, 2005; Rubino, 2005; Raimy, 2012). For instance, Rubino (2005) surveys 368 languages and shows that about 85% exhibit some form of productive reduplication. While the typology of reduplication types is rich, two broader classes of processes have been usually distinguished (Inkelas and Downing, 2015; Urbanczyk, 2007):

- partial reduplication, in which a bounded number of segment are repeated (e.g. the last syllable of a word);
- total reduplication, which repeats unboundedly many segments to form some new morphological constituent.

It has been observed that reduplication presents an interesting challenge to finite-state computational approaches to morpho-phonology (Dolatian and Heinz, 2019b; Rawski et al., 2023). From a computational perspective, by its bounded nature partial reduplication can be modelled with (subsequential) 1-way finite-state transducers (FSTs), although with a significant explosion in the number of required states (Roark and Sproat, 2007). On the other hand, because the number of copied elements has hypothetically no upper bound, total reduplication cannot be modelled with these machines at all — leading some practitioners to adopt memorized lists of words as a way to deal with it in practical applications (Roark and Sproat, 2007; Dolatian and Heinz, 2019a). As total reduplication seems to be one of the few (if not the only) morpho-phonological processes not easily dealt with via 1-way FSTs, it is of particular interest both for practical and theoretical research on finite-state computational models (Dolatian and Heinz, 2019b). In this sense, Dolatian and Heinz (2020) demonstrate how it is possible to use Deterministic 2-way FSTs — essentially, FSTs able to move back and forth on the input tape — to succinctly model both partial and full *segmental* reduplication. Expanding on this intuition, Markowska et al. (2021) observe that a complete finite-state treatment of reduplication cross-linguistically is further complicated by the fact that many languages exhibiting total reduplication are also tonal, and models need to simultaneously capture the somewhat distinct processes affecting the segmental and the autosegmental levels. Importantly, by showing that tones may act independently from their tone-bearing units, classical work in autosegmental phonology has argued for the representational separation of tones from segments (Leben, 1973; Goldsmith, 1976, a.o.). Following work by Dolatian and Rawski (2020), Markowska et al. (2021) argue that modelling the morpho-phonology of languages with both reduplication and tone requires the synthesis of 1-way, 2-way FSTs, and multi-tape FSTs (Filiot and Reynier, 2016; Furia, 2012; Rawski and Dolatian, 2020) — finite state machines with multiple input/output tapes that can be used to mimic autosegmental representations (i.e., splitting the segmental and tonal levels; Wiebe, 1992; Rawski and Dolatian, 2020, a.o.).

Proceedings of the Society for Computation in Linguistics (SCiL) 2023, pages 85-94.
Amherst, Massachusetts, June 15-17, 2023

Importantly, the model in Markowska et al. (2021) is motivated and validated on languages like Shupamem, which exhibit a clear separation between tonal and segmental processes, and that seem to exhibit reduplication only on the segmental level. However, broadening our typological observations is crucial in getting insights into the generalizability of our computational approaches.

Here, we adopt Markowska et al. (2021)'s synthesis approach to reduplication in Thai, building on the observation Thai's total reduplication affects both levels of representation. In other words, Thai exhibits total reduplication both at the segmental and tonal levels, each level then undergoing additional separate transformations (e.g. vowel change in the reduplicant). We then suggest that the approach in Markowska et al. (2021) can be easily extended to languages like Thai by adopting 2-way FSTs for reduplication on both levels, supporting the overall generalizability of the synthetic approach.

2 Reduplication and Tone in Thai

Thai is a member of the Tai-Kadai language family and is the official language of Thailand (Chakshuraksha, 1994). It features five tones (Lee, 2011), which we represent orthographically with diacritics on vowels, following similar literature on the topic: Mid (M; represented by an unmarked V), Low (L; diacritic V̀), High (H; diacritic V́), Rising (R; diacritic V̌), Falling (F; diacritic V̂). Note that for simplicity, we chose to not represent rising and falling tones as a sequence of LH and HL tones, respectively, but this is a choice that does not particularly affect our analysis.[1] Before moving on to a discussion of the variety of reduplication processes available in Thai, we briefly touch on its strict relation between tone preassociation and syllable structure.

2.1 Constraints on Syllable Structure

Thai has a relatively restricted syllable structure: an initial consonant followed by an optional liquid/glide consonant forms the onset, followed by a vocalic nucleus with a tone, and an optional stop/nasal coda (Gandour, 1974; Chakshuraksha, 1994; Hudak, 2007). The general syllable structure, adapted from Cooke (1963), is shown in 1 and 2, the interpretation for which is given in 3, accounting for the phoneme inventory of the language.

1. $C(C_1)\overset{T}{V}(C_2)$

2. $C(C_1)\overset{T}{V\text{ː}}(C_2)$

3. C = any consonant
   $C_1 = \{w, l, r\}$
   $C_2 = \{m, n, ŋ, j, w, p, t, k, ʔ\}$
   V = any vowel
   Vː = any long vowel or the diphthongs /ia/, /ua/, /ɯa/
   T = any tone

In what follows we will ignore the fact that some coda obstruents (C_2) are realized as unreleased {pˀ, tˀ, kˀ}, since this is a transformation not relevant to the process of interest. Note also that vowel length and aspiration are contrastive in Thai, and we use the ː symbol to indicate vowel length.

Thai's tonal phonotactics distinguishes *live* and *dead* syllables. Live syllables are defined as those that end in a sonorant, e.g. [maː] 'to come' or [jàj] 'big'. These are unrestricted and can feature all five tones. Dead syllables are defined as those that end in a stop, e.g. [jàːk] 'to want' or [rót] 'car'. These are restricted: dead syllables with a *short* vowel can feature only low and high tones, while dead syllables with a *long* vowel can feature only low and falling tones. Note that the terms *live* and *dead* are replaced elsewhere in the literature by the terms *unchecked* and *checked*, or *unclosed* and *closed* (Gandour, 1974; Lee, 2011; Cooke, 1963). These constraints on tone showcase the importance of preassociation between segmental and autosegmental levels, and how this might feed into other downstream processes. Thus, attention must be paid when formulating models that posit a strict separation between the two levels of representation (Lee, 2011; Gandour, 1974; Moren and Zsiga, 2001; Rawski and Dolatian, 2020).

2.2 Thai Reduplication

Reduplication in Thai is a productive process that is able to target every grammatical word category (Chakshuraksha, 1994; Sookgasem, 1997). Crucially, total reduplication targets both the segmental and the autosegmental level. We distinguish four types of total reduplication processes, based on their grammatical/semantic function and morpho-phonological changes they induce. This

[1] We follow past work in using an alphabet enriched with diacritics to represent associations between tones and segments, but it is important to keep in mind that enriched alphabets reveal the need for more expressive representations (e.g., graphs) to capture tone beyond orthographic conventions (Yli-Jyrä, 2013; Jardine, 2019).

paper adopts the naming conventions defined in Sookgasem (1997) for the various reduplication patterns: *Simple*, *Complex Type 1*, *Complex Type 2*, and *Complex Type 3*. Complex Type 3 is also called "emphatic reduplication" elsewhere in the literature (Lee, 2011; Haas, 1946; Chakshuraksha, 1994, a.o.). Henceforth, we use the ∼ symbol to separate the base from the reduplicant and represent the reduplication boundary, consistently with Markowska et al. (2021).

2.2.1 Simple Reduplication

Simple Reduplication exhibits no change to the base or reduplicant, neither on the segmental level nor on the tonal level (Sookgasem, 1997; Chakshuraksha, 1994; Haas, 1946). In this type of reduplication the base is copied once and the meaning is changed depending on the word class, as in (i) and (ii).

(i) dɛ̀k→dɛ̀k∼dɛ̀k 'child'→'children'

(ii) nâŋ→nâŋ∼nâŋ
'to sit'→'to sit continuously'

2.2.2 Complex Reduplication Type 1

In Complex Reduplication Type 1 the final vowel of the reduplicant is changed to either /ə/ or /æ/ (iii), both vowels being used interchangeably and usage depends only on speaker preference (Chakshuraksha, 1994; Sookgasem, 1997).

(iii) faŋ→faŋ∼f**æ**ŋ 'to listen'→'to listen'

The autosegmental level is once again fully reduplicated without any changes (in (iii), a mid-tone V is copied as a mid-tone V). This reduplication pattern indicates a level of negativity or disinterest towards something or someone.

2.2.3 Complex Reduplication Type 2

Complex Reduplication Type 2 follows a reduplicant∼base template, with the reduplicant as the first copy, and it is similar in meaning to Complex Reduplication Type 1 (Sookgasem, 1997).

(iv) còt.mă:j→còt.m**ŏ:ŋ**∼còt.m**ă:j**
'a letter'→'a letter'

(v) sì̀t→s**òk**∼s**ì̀t** 'a right'→'a right'

(vi) kà?.t^{h}í?→kà?.t^{h}**ó?**∼kà?.t^{h}**í?**
'coconut milk'→'(something like) coconut milk'

At the segmental level, if the base word ends in /oŋ/, /ok/, or /o?/, then that word cannot undergo this type of reduplication (Sookgasem, 1997). In the reduplicated form, the final syllable of the reduplicant is changed to /oŋ/, /ok/, or /o?/, with the vowel length of the final syllable of the base being maintained. The ending /oŋ/ is used when the final syllable of the base ends in /m/, /n/, /j/, /w/, or in a long vowel — i.e. live syllables (iv). The ending /ok/ is used when the final syllable of the base ends in /p/ or /t/ (v). The ending /o?/ is used when the final syllable of the base is a short vowel followed by a glottal stop (vi). Again, the tonal level is fully reduplicated with no changes.

2.2.4 Complex Reduplication Type 3

Complex Reduplication Type 3 is similar to Simple Reduplication, except that the first copy is made to exhibit a high tone on its final syllable (Sookgasem, 1997; Lee, 2011; Chakshuraksha, 1994; Haas, 1946).

(vii) sŭaj→sú̋aj∼sŭaj
'pretty'→'really pretty'

(viii) nâ:.rák→nâ:.rǎ̋k∼nâ:.rák
'cute'→'really cute'

When the final syllable of the base word already exhibits a high tone, then an *extra* high tone is used (represented with the diacritic V̋). The extra high tone, also called the *emphatic* high tone, is not considered among the basic five tones in Thai because it is not contrasting. Phonetically speaking, the emphatic high tone differs from the basic high tone in that it is higher in pitch and usually lengthened (Lee, 2011). Complex Reduplication Type 3 is, by implication, emphatic or intensifying in meaning.

3 Finite-state Models of Total Reduplication in Tonal Languages

With an understanding of Thai tonal and reduplicative processes in place, in this section we provide a brief, intuitive overview to the classes of finite-state machines combined by Markowska et al. (2021) in their model of total reduplication. We will then explore how this model can be adapted to Thai in the next section.

3.1 Total Reduplication with 2-way FSTs

As mentioned, reduplication in general has been the focus of many studies in the computational linguistics' literature, as it seems to be (one of) the

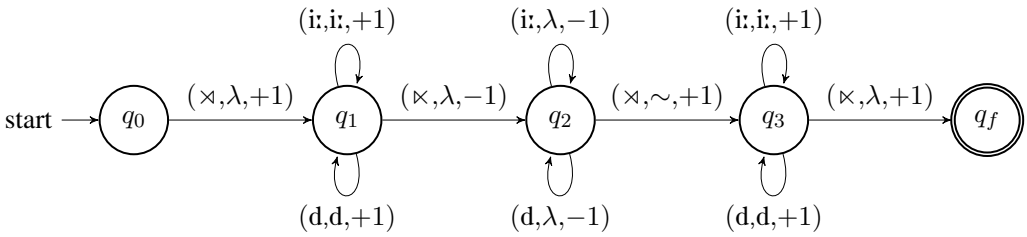

Figure 1: 2-way FST for full reduplication of diː 'good' → diː∼diː 'very good'

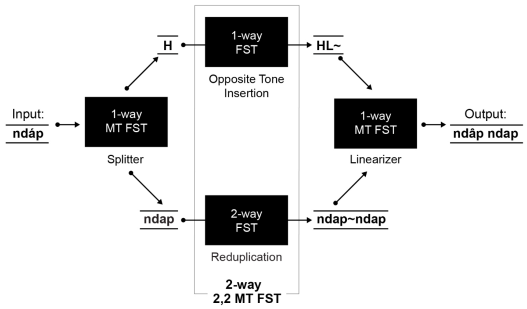

Figure 2: Shupamem reduplication model adapted from (Markowska et al., 2021)

only process(es) in morpho-phonology that cannot be modelled with a 1-way FST (i.e., the output of this process is not a regular language; Roark and Sproat, 2007). In the case of partial reduplication (where only a bounded set of elements needs to be copied) the issue lies in an explosion in the number of states. However, total reduplication affects elements (e.g. full words or phrases) with no a-priory bounds. Dolatian and Heinz (2020) address this problem by adopting 2-way FSTs. Essentially, a 2-way FST increases the expressivity of 1-way FSTs by being able to move back and forth on the input tape, allowing it to read its input more than once (Rabin and Scott, 1959). In designing the machine, state transitions are enriched with a direction parameter ($\{-1,0,+1\}$) that indicates if the FST should move back to the previous symbol, stay on the current symbol, or advance to the next symbol. Dolatian and Heinz (2020) show that this class of transducers not only is able to capture both partial and total reduplication, but it does so in a way that is more transparent with respect to the generalizations argued for in the linguistic literature (see also Dolatian and Heinz, 2019a).

Modelling total reduplication with a 2-way FST involves three steps: (1) reading the input tape left-to-right and outputting the first copy, (2) reading the input tape right-to-left and stopping once the left word boundary ⋊ is read, (3) reading the input tape from left-to-right and outputting the second copy. Figure 1 is an example of a 2-way FST that fully reduplicates the Thai word *diː* 'good' to produce *diː∼diː* 'very good'. In the graphical representation, the input-output pair is grouped with the direction parameter, with each element being separated by a comma. Following Dolatian and Heinz (2019a), we make it so that when reading left-to-right (forward) the input tape is copied on the output tape faithfully. When moving backward (right-to-left), the machine outputs an empty symbol, so that the input string can then be copied again in an additional forward pass. We refer the reader to Dolatian and Heinz (2020) for a full formal treatment of these machines.

3.2 Tone, Reduplication, (2-way) MT FSTs

While the 2-way FST approach of Dolatian and Heinz (2019a) is successful in modeling reduplication at the segmental level, Markowska et al. (2021) point out that many of the world languages exhibiting productive reduplication processes are *tonal*. This presents an additional challenge for finite-state models, as there is the need to handle processes that affect the segmental and autosegmental representations separately. Autosegmental processes have also been argued to exhibit different computational properties than their segmental counterparts (Yli-Jyrä, 2013; Jardine, 2015, 2019, a.o.).

In order to mimic the representational difference between segmental and autosegmental levels within finite-state machines, Dolatian and Rawski (2020) adopt *multi-tape* FSTs (MT FSTs) (see also Fischer, 1965; Wiebe, 1992; Frougny and Sakarovitch, 1993; Furia, 2012; Rawski and Dolatian, 2020). We refer the reader to (Dolatian and Rawski, 2020; Rawski and Dolatian, 2020) for a complete formal treatment of these machines, and here we just cover the basic intuition behind them. Essentially, a MT FST is similar to a 1-way FST with a single tape, but

is able to operate (read from and write to) multiple tapes. This means that such machines can take as input two tapes — a tonal tape and a segmental tape — and operate over them synchronously even when they are subject to different processes.

Using as a motivating starting point Shupamem (a Bantu language), Markowska et al. (2021) observes that a combination of the properties of both 2-way FSTs and MT FSTs is in fact needed to correctly account for the patterns observed in tonal languages with reduplication. Specifically, they synthesize the work in Dolatian and Heinz (2020) and Dolatian and Rawski (2020) to propose deterministic 2-way (n,m) MT FSTs, where n,m refer respectively to the number of input and output tapes. They then present a model of reduplication that makes use of 1-way MT FSTs with a single input tape and two output tapes, in order to split a single string — where tone is orthographically represented with an enriched alphabet using diacritics — into a tonal level and an segmental level. Those are then used as inputs to a 2-way (2,2) MT FSTs composed of a 2-way FST which reduplicates the segmental level, and a 1-way FST dealing with an insertion process on the tonal level. Finally, the two output tapes in the previous step are fed into a $(2,1)$ MT FST which combines them into a reduplicated, enriched output string (Figure 2). Again, we refer the reader to Markowska et al. (2021) for a full discussion of the formal details.

4 Modeling Thai

The synthetic approach surveyed above shows how it is possible to handle both reduplication and autosegmental representations deterministically within a finite-state model. Importantly though, Shupamem (and the other tonal languages analyzed by Markowska et al., 2021) exhibits full reduplication exclusively at the segmental level, while the autosegmental level is affected by other phonological processes targeting tone. Because of this, their 2-way (2,2) MT FST is really 2-way only on one of the two tapes. However, we observed how in Thai the reduplication process on the tonal level mimics the reduplication process on the segmental level. Each of the reduplication types above illustrates full reduplication on both levels, which would by itself be challenging for the single 2-way FST adopted for Shupamem. Additionally, different reduplication types are distinguished by the need of additional dedicated transformations on either the segmental or autosegmental level. Specifically, Complex Reduplication Type 2 showcases transformations that target segmental information, while Type 3 illustrate changes targeting tone specifically.

C	any consonant
V	any vowel
T	any tone
T′	{M, L, R, F}
K	{p, t, k, ʔ}
S	{m, n, ŋ, j, w}
C′	C - S
E	extra high tone
λ	empty string

Table 1: List of shorthand symbols used in the FSTs.

Because of these facts, Thai serves as a good test case to explore the flexibility of the synthetic approach. In particular, by formalizing the reduplication types discussed above, in what follows we illustrate how Thai clearly shows the need for 2-way FSTs on both segmental and autosegmental tapes.

We assume a model like the one in Figure 2, which utilizes MT FSTs as *splitters* and *linearizers* to move from and to orthographic representations with an enriched alphabet. These MT FSTs are unchanged with respect to the ones presented by Markowska et al. (2021), and thus we refrain from including examples of them in this paper. We focus instead on the application of the 2-way $(2,2)$ MT FST (boxed section in Figure 2) to the variety of reduplication processes in Thai.

Henceforth, we define the alphabet our machines operate on using the following shorthand: C refers to any consonant, V refers to any short vowel, V: refers to any long vowel or diphthong, and a period (.) to syllable boundaries. Additionally, we use K for the set {p, t, k, ʔ}, and S for the set {m, n, ŋ, j, w}. A summary of these abbreviations (and all those used in the FSTs that follow) is shown in Table 1.

4.1 Syllable-Tone Association

If we follow Markowska et al. (2021)'s in adopting an initial alphabet with diacritics, it seems useful to incorporate an additional step before the splitter in order to guarantee the correct preassociations of tones and segments. Recall that tone restrictions are placed only on dead syllables: short dead syllables only feature low and high tones, and long dead syllables only feature low and falling tones. As these constraints are all local over the enriched alphabet, we could easily handle them

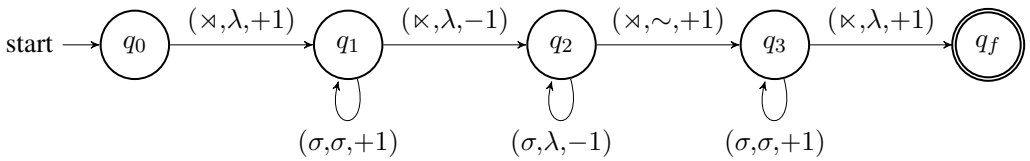

Figure 3: 2-way FST for Simple Reduplication (either segmental or tonal level).

with a 1-way FST. Dealing with tonal constraints with 1-way FSTs over enriched representations in not novel of course (see for example Yli-Jyrä, 2013, a.o.), and we could alternatively handle preassociation with MT FSTs scanning the two levels synchronously (Rawski and Dolatian, 2020). What this draws attention to though, is the need to consider tone-segment preassociation even within models which require separate levels at some point.

4.2 Simple Reduplication Model

We can now start looking at Thai's reduplication processes. Recall that in the case of simple reduplication, both the segmental and autosegmental levels undergo total reduplication, with both copies being rendered faithfully with respect to the input:

sàʔ.ʔàːt→sàʔ.ʔàːt∼sàʔ.ʔàːt
'clean'→'very clean'

Although the synthetic model for Shupamem assumes a 1-way FST for tone, the most general, formal definition of 2-way (2, 2) MT FST in Markowska et al. (2021) seems to allow for 2-way FSTs on both tapes. This is exactly the approach that we take. Figure 3 is an example of 2-way FST that models simple reduplication in Thai. This is essentially identical to the FST shown in Figure 1. The symbol σ represents any symbol in an alphabet, that is $\sigma \in \Sigma$, so that (instances of) this FST can work for both the segmental level and the tonal level. A (2,2) MT FST of simple reduplication would then apply an instantiation of the FST in Figure 3 on both tapes.

4.3 Complex Reduplication Type 1

Consider now Complex Reduplication of Type 1:

faŋ→faŋ∼fæŋ 'to listen'→'to listen'

Recall that a vowel without a diacritic is not toneless, but bears a Mid tone. This reduplication type shows full reduplication of both tones and segments, but at the segmental level the final vowel of the reduplicant is changed to either /ə/ or /æ/ (we will use /æ/ for simplicity, since this assignment is speaker-specific).

A 2-way FST that reduplicates the segmental level is shown in Figure 4, a derivation for which is shown in Table 2. The first time the word is copied, it is copied faithfully. The second time it is copied, we want the final vowel of the word to change. For this reason, we output the syllable and loop back to q_3 until a word boundary symbol is read. Once the word boundary symbol is read, the final syllable is outputted accordingly, including the vowel change. For total reduplication on the tonal level, the FST in Figure 3 suffices since there is no tone change.

State	Input-Tape		Output-Tape
q_0	⋊*sa*ʔ.ʔa:*t*⋉	+1	λ
q_1	⋊*s*a*ʔ*.ʔa:*t*⋉	+1	s
q_1	⋊*sa*ʔ.ʔa:*t*⋉	+1	sa
q_1	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ
q_1	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.
q_1	⋊*sa*ʔ.*ʔ*a:*t*⋉	+1	saʔ.ʔ
q_1	⋊*sa*ʔ.ʔ*a*:*t*⋉	+1	saʔ.ʔa:
q_1	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t
q_1	⋊*sa*ʔ.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔ*a*:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.*ʔ*a:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*s*a*ʔ*.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔa:*t*⋉	−1	saʔ.ʔa:t
q_2	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t∼
q_3	⋊*s*a*ʔ*.ʔa:*t*⋉	+1	saʔ.ʔa:t∼s
q_4	⋊*s*a*ʔ*.ʔa:*t*⋉	+1	saʔ.ʔa:t∼s
q_6	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t∼s
q_8	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t∼saʔ.
q_3	⋊*sa*ʔ.*ʔ*a:*t*⋉	+1	saʔ.ʔa:t∼saʔ.ʔ
q_4	⋊*sa*ʔ.ʔ*a*:*t*⋉	+1	saʔ.ʔa:t∼saʔ.ʔ
q_5	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t∼saʔ.ʔ
q_7	⋊*sa*ʔ.ʔa:*t*⋉	+1	saʔ.ʔa:t∼saʔ.ʔæ:t

Table 2: Complex Type 1 derivation for the segmental level (Figure 4) of *sàʔ.ʔàːt* 'clean' → *sàʔ.ʔàːt∼sàʔ.ʔæ̀ːt* 'too clean'.

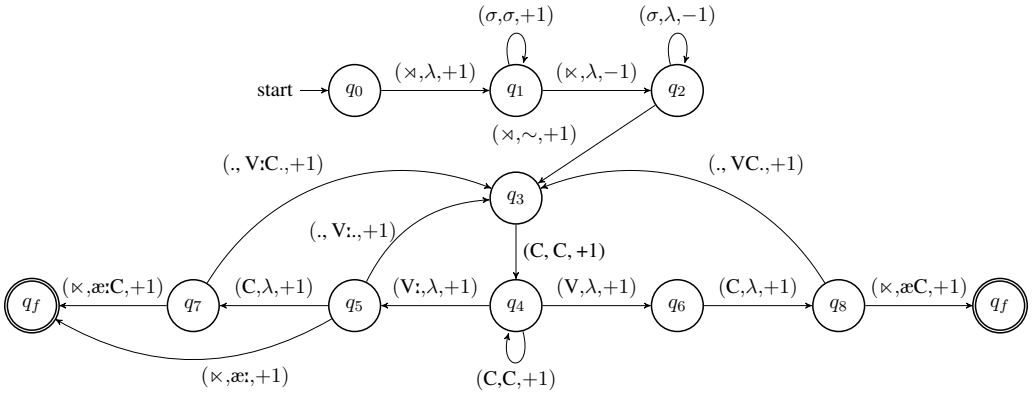

Figure 4: 2-way FST for Complex Reduplication Type 1 at the segmental level.

4.3.1 Complex Reduplication Type 2

Complex Reduplication of type 2 involves a reduplicant-base pattern, with a change to the final sylllable of the reduplicant (the first copy):

> càt.măːj→càt.m**ŏːŋ**∼càt.m**ăːj**
> 'a letter'→'a letter'

An FST that handles reduplication for the segmental level for Complex Reduplication Type 2 is shown in Figure 5. For the sake of readability, only one of the three endings (/oŋ/) is considered here. We use S as a shorthand for the set {m, n, j, w}. The shorthand C represents the set of all consonants in Thai, as previously used in this paper. The shorthand C′ represents the set of all consonants in Thai excluding the set S, such that the operation C−S = C′ holds true.

For this process, the first time a word is copied we want the rhyme of the final syllable to change. Thus, we loop back to q_1 until a word boundary symbol is read. The FST only allows words to end in consonants in the set S = {m, n, j, w}. Once the first copy is outputted with the rhyme change, then the second copy is faithfully read and outputted.

We mentioned that Complex Reduplication Type 2 is not possible for words that end in /oŋ/, /ok/, or /oʔ/ (Sookgasem, 1997). We could of course include this restriction in the FST in Figure 5, for example by handling the /o/ and /oː/ vowels separately from all other vowels, and excluding a transition where the ⋉ symbol is read after a syllable containing /o/ or /oː/. Alternatively, another FST could be added to the pipeline to filter what kind of inputs are appropriate for each reduplication type. Once again, we can use the FST in Figure 3 for the tonal level reduplication here since it involves total reduplication with no tone change.

4.3.2 Complex Reduplication Type 3

In Complex Reduplication of type 3, the segmental level is reduplicated faithfully (which can be accomplished with the FST in Figure 3). At the autosegmental level, the final syllable of the first copy is made to bear a high tone, while the original tone appears faithfully in the second copy:

> nâː.rák→nâː.rǎk∼nâː.rák
> 'cute'→'really cute'

This process is modelled by the 2-way FST in Figure 6. We use T as a stand in for any tone ({M, L, H, R, F}) except for the extra high tone with we represent as E, and T′ to stand in for non-high tones ({M, L, R, F}). For the first copy, as the only tone that needs to be changed is associated to its last syllable, after reading a tone from the input tape the FST "waits" to check whether the immediate next element is a boundary symbol (⋉) before outputting it. If the tone was a non-high tone and the next element is ⋉, a high tone is outputted. If the tone was a high tone and the next element is ⋉, then an extra-high tone is outputted. If not at the end of the string, tones are outputted faithfully. The second copy is fully faithful.

5 Conclusion

This paper builds on previous work in adopting a deterministic finite-state approach to model the interaction of total reduplication and tonal processes in Thai. Markowska et al. (2021) synthesized an approach to autosegmental processes via MT FSTs (Dolatian and Rawski, 2020; Rawski and Dolatian, 2020) and 2-way FSTs to deal with total reduplication (Dolatian and Heinz, 2019a, 2020) in order to account for what observed in Shupamem. They show how this combination allows them to deal with

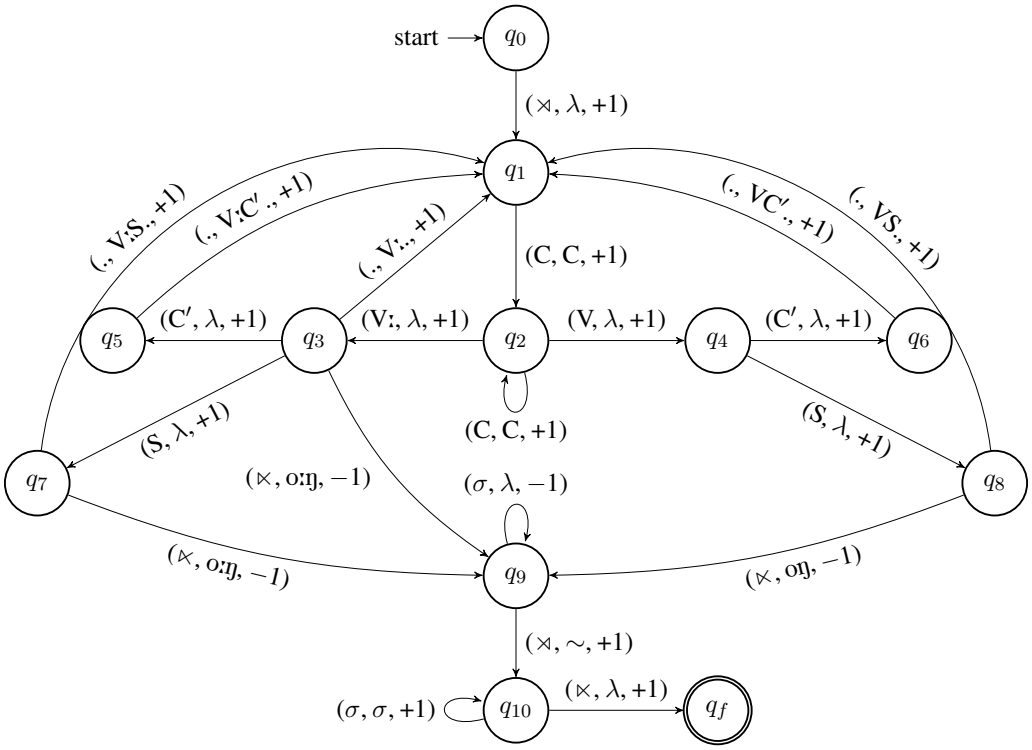

Figure 5: 2-way FST for the segmental level of Complex Reduplication Type 2.

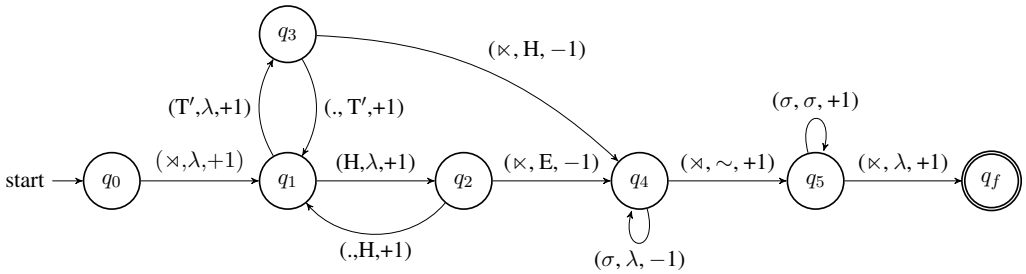

Figure 6: 2-way FST for the tonal level of Complex Reduplication Type 3.

the double challenge of handling unbounded copies (as required by total reduplication), and separate segmental and autosegmental processes while remaining faithful to linguistic analyses of these patterns.

Crucially, Shupamem exhibits total reduplication exclusively on the segmental level, thus allowing the model to fully treat tone and segments separately. Here, we used Thai as an example of a language where tones also undergo reduplication. We suggested then to take full advantage of the expressivity of the 2-way (2,2) MF FST model, by making sure that both the segmental and the autosegmental tapes are used as inputs to 2-way FSTs. In doing this, we showed how carefully exploring the typological diversity of tonal languages with reduplication will enrich our understanding of the expressivity required by finite-state models.

Looking back at our analyses of Thai, it is reasonable to wonder whether we could have handled the reduplication pattern as a whole with a single 2-way FST, without need for the MT FST split. While this is doable adopting an enriched alphabet, the MT FST approach allows us to remain as close as possible to linguistic analyses when modeling the independent changes the segmental and autosegmental levels go through in the Complex Reduplication types. However, the concatenation of 2-way and multi-tape FSTs potentially pushes the expressivity of these machines quite high (Fischer, 1965; Furia, 2012), stressing how crucial it is going to be for an insightful computational theory of morpho-phonology to conduct an extensive formal

evaluation of the expressive power of alternative combinations/restrictions of these devices.

In sum, these results add support to the deterministic finite-state approach to total reduplication advanced in previous literature, while highlighting the fundamental role of broader typological evaluation.

Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable feedback. This work was supported by funding from the Undergraduate Research Opportunities Program at the University of Utah awarded to Casey Miller.

References

Nuttathida Chakshuraksha. 1994. Prosodic structure and reduplication in thai. *Working Papers of the Linguistics Circle*, 12:27–38.

Joseph R. Cooke. 1963. The vowels and tones of standard thai: Acoustical measurements and experiments. arthur s. abramson. *American Anthropologist*, 65:1406–1407.

Hossep Dolatian and Jeffrey Heinz. 2019a. Learning reduplication with 2-way finite-state transducers. In *International Conference on Grammatical Inference*, pages 67–80. PMLR.

Hossep Dolatian and Jeffrey Heinz. 2019b. Redtyp: A database of reduplication with computational models. *Proceedings of the Society for Computation in Linguistics*, 2(1):8–18.

Hossep Dolatian and Jeffrey Heinz. 2020. Computing and classifying reduplication with 2-way finite-state transducers. *Journal of Language Modelling*, 8(1):179–250.

Hossep Dolatian and Jonathan Rawski. 2020. Multi-input strictly local functions for templatic morphology. *Proceedings of the Society for Computation in Linguistics*, 3(1):282–296.

Emmanuel Filiot and Pierre-Alain Reynier. 2016. Transducers, logic and algebra for functions of finite words. *ACM SIGLOG News*, 3(3):4–19.

Patrick C Fischer. 1965. Multi-tape and infinite-state automata—a survey. *Communications of the ACM*, 8(12):799–805.

Christiane Frougny and Jacques Sakarovitch. 1993. Synchronized rational relations of finite and infinite words. *Theoretical Computer Science*, 108(1):45–82.

Carlo A Furia. 2012. A survey of multi-tape automata. *arXiv preprint arXiv:1205.0178*.

Jack Gandour. 1974. On the representation of tone in siamese. *UCLA Working Papers in Phonetics*, 27:118–146.

John Anton Goldsmith. 1976. *Autosegmental phonology*. Ph.D. thesis, Massachusetts Institute of Technology.

Mary R Haas. 1946. Techniques of intensifying in thai. *Word*, 2(2):127–130.

Thomas John Hudak. 2007. *William J. Gedney's comparative Tai source book*. University of Hawaii Press.

Bernhard Hurch and Veronika Mattes. 2005. *Studies on reduplication*. Mouton de Gruyter Berlin.

Sharon Inkelas and Laura J Downing. 2015. What is reduplication? typology and analysis part 1/2: The typology of reduplication. *Language and linguistics compass*, 9(12):502–515.

Adam Jardine. 2015. Computationally, tone is different. *Phonology*.

Adam Jardine. 2019. The expressivity of autosegmental grammars. *Journal of Logic, Language and Information*, 28:9–54.

William Ronald Leben. 1973. *Suprasegmental phonology.* Ph.D. thesis, Massachusetts Institute of Technology.

Leslie Lee. 2011. Fixed autosegmentism in thai emphatic reduplication. *Journal of the Southeast Asain Linguistics Society*, 4:41–63.

Magdalena Markowska, Jeffrey Heinz, and Owen Rambow. 2021. Finite-state model of shupamem reduplication. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 212–221.

Bruce Moren and Elizabeth Zsiga. 2001. lexical tone and markedness in standard thai. In *Annual Meeting of the Berkeley Linguistics Society*, volume 27, pages 181–191.

Michael O Rabin and Dana Scott. 1959. Finite automata and their decision problems. *IBM journal of research and development*, 3(2):114–125.

Eric Raimy. 2012. *The phonology and morphology of reduplication*, volume 52. Walter de Gruyter.

Jonathan Rawski and Hossep Dolatian. 2020. Multi-input strict local functions for tonal phonology. *Proceedings of the Society for Computation in Linguistics*, 3(1):245–260.

Jonathan Rawski, Hossep Dolatian, Jeffrey Heinz, and Eric Raimy. 2023. Regular and polyregular theories of reduplication. *Glossa: a journal of general linguistics*, 8(1).

Brian Roark and Richard Sproat. 2007. *Computational approaches to morphology and syntax*, volume 4. OUP Oxford.

Carl Rubino. 2005. Reduplication: Form, function and distribution. *Studies on reduplication*, 28:11–29.

Prapa Sookgasem. 1997. A complicating distortion of syntactic categories: The case of reduplication in thai. *Southeast Asian linguistics studies in honor of Vichin Panupong*, pages 253–272.

Suzanne Urbanczyk. 2007. Themes in phonology. *The Cambridge Handbook of Phonology, edited by Paul de Lacy*, pages 473–493.

Bruce Wiebe. 1992. Modelling autosegmental phonology with multi-tape finite state transducers.

Anssi Mikael Yli-Jyrä. 2013. On finite-state tonology with autosegmental representations. In *Proceedings of the 11th international conference on finite state methods and natural language processing*. The Association for Computational Linguistics.

A Appendix

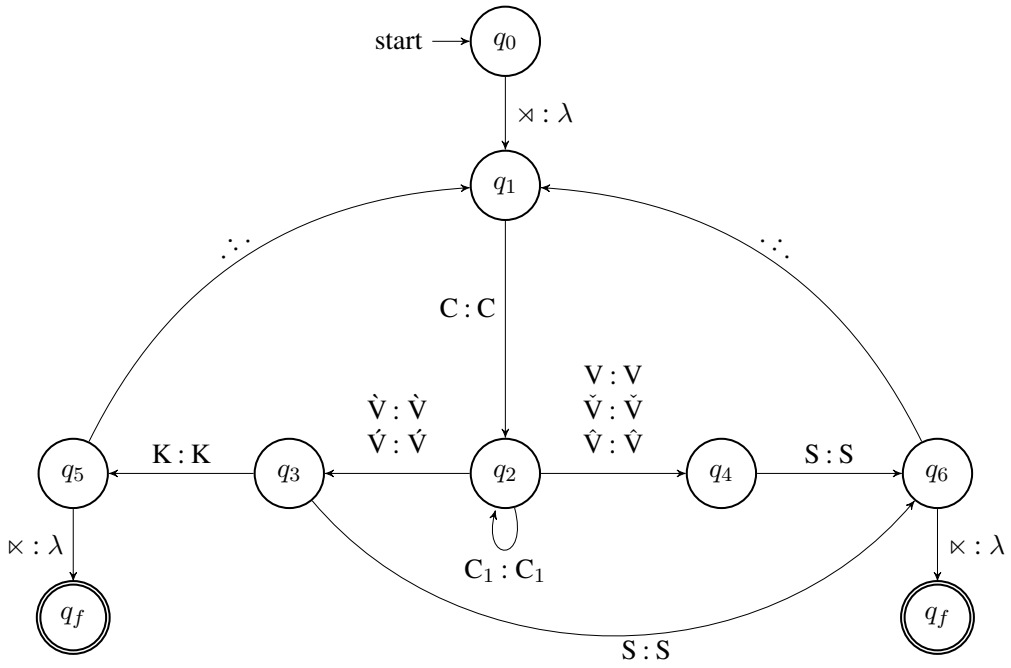

Figure 7: 1-way FST to model the phonotactics of short dead syllables in Thai. $C_1 = \{w, l, r\}$.