A Basis for a Syllable-Based English Spelling Decoding System for Advanced Learners: A *Deletional* Model

Teruo YOKOTANI

1. Introduction

While the specific implementation may vary, it is indisputable that teaching the relationship between spelling and pronunciation is crucial in English education. Phonics is readily available in elementary education, and there are sets of rules published for intermediate learners. However, there appears to be a lack of comprehensive rule sets to impart advanced knowledge to higher-level learners, in contrast to the well-established field of English grammar. Research by the author has found no similar resources specifically focused on spelling.

Yokotani (2023a) has provided insight into a potential solution for this perceived gap. In this study, the concept of silencing is applied not only to individual letters but also to pronunciation symbols. Essentially, nearly all the phenomena that would typically be addressed by deletion rules in a conventional framework are instead treated as silencing of pronunciation symbols. This decision arose from concerns about potential complexity introduced by deletion rules. In situations where a deleted symbol influences other processes, the need to establish an order between these two kinds of processes may arise. Acquiring such extra rule-by-rule conditions can pose a burden for learners.

However, this silencing approach is not without its challenges, as partly acknowledged in the same paper. Besides encountering some rules that demanded seemingly ad hoc conditions for their application, similar operations have also been identified elsewhere. Furthermore, specifying the timing for the deletion of silenced pronunciation symbols presents its own difficulties. These drawbacks should be recognized as significant challenges.

This paper introduces a model that employs deletion rules, with minimal reliance on extrinsic ordering. The focus is primarily on one particularly challenging case, for which a practical solution is provided. Furthermore, the organization of the different rule sets assumed in the previous work has been simplified, making the entire system more learner-friendly. Moreover, the illustration of the workings of the rules employed elucidates the cognitive processes that decoders need when implementing these rules. The collective insights and methodologies presented in this paper are

anticipated to foster positive advancements in the field.

The organization of the present paper is outlined as follows: Section 2 delves into the issues associated with the silencing approach; Section 3 introduces the terms and concepts fundamental to the new system. A comprehensive overview of this system is offered in Section 4, while detailed illustrations, with a particular emphasis on one of the most challenging cases, are presented in Section 5. Concluding remarks in Section 6 wrap up the paper.

2. Problems with the Silencing Model

One of the distinctive features of the decoding model proposed by Yokotani (2023a) is its incorporation of silencing operations for not only individual letters but also pronunciation symbols. In contrast to other possible systems, this one processes nearly all events that would typically be handled by deleting pronunciation symbols via silencing of those symbols. This choice was made to avoid unnecessary complexity in the rule system that would arise from using deletion rules.

Specifically, when a deleted symbol affects other processes, it often necessitates somehow specifying the order of application for these processes. For instance, in the words *car* and *carry*, distinct vowel symbols, [ɑː] and [æ], are assigned to the <a> vowel letters. This difference can be explained by postulating that the first [r]s in the pronunciation representations are syllable-final in *car* and syllable-initial in *carry*. ([•kɑːr•] vs. [•kæ •rɪ•], where '•' stands for a syllable boundary.) Consequently, part of the decoding process flow that can explain the difference between these two words is as follows: Assignment of [r] (plus the appearance of a geminate sequence) (→ Resolution of the geminate sequence) → Assignment of [æ] (and its adjustment to [ɑː]) It is crucial that the symbol corresponding to the first <r> in *carry* does not exist in the pronunciation representation at the point where the vowel adjustment takes place. To ensure this, however, it is necessary to specify the order of rule application: deletion of the first [r] in the geminate must occur prior to the vowel adjustment. However, such specifications can be burdensome for learners.

Anticipating the prevalence of such issues, Yokotani (2023a) proposed generally replacing rules deleting pronunciation symbols with those silencing them. In this approach, even if a symbol is not pronounced, it can still persist within the decoding process, influencing other rules or preventing their application. The idea was that a provision ensuring silenced symbols do not appear in the final representation would be sufficient.

It has been found, however, that the aforementioned strategy has its drawbacks. Specifically, the following issues have been identified. First, although not explicitly mentioned as a challenge in Yokotani (2023a), it was unavoidable to specify the application conditions of a rule that adjusts

vowel symbols immediately before [r] as being immediately before a coda [r] *that is not silent*, to distinguish cases such as *car* [•kɑːr•] and *carry* [•kær•ri•] (p. 110), where the shaded [r] is a silent one. Additionally, in the case of a resyllabification rule that was posited to occur in words such as *celluloid* ([•sél**l•l**jʊ•lɔɪd•]→[•sé**ll•** jʊ•lɔɪd•]), it was necessary to introduce an undesirable condition regarding the target consonant symbols: Entire geminate strings had to be permitted to be re-syllabified as well (p. 107). Furthermore, moving a whole string of geminate to a coda position in this way created another challenge. Because silencing for degemination was assumed to take place on the element closest to the right-hand edge of a syllable, it was possible for both elements in a geminate to be silenced ([...l•l...]→[...ll•...]→[...ll•...]). To prevent this, an ad-hoc constraint was introduced to ensure that both elements of a geminate cannot be silent simultaneously (p. 109).

Moreover, additional issues not addressed in the same paper have come to light. Consider cases such as *rheumatism*, which would start with the sequence [rhjuː...], where [h] is silent. To achieve the desired final outcome of /ruː.../, muting the [j] would be essential. However, the context for this [j]-silencing would have to be an undesirable one: "immediately after [r], *which can optionally be followed directly by silent [h]*". Additionally, in a minority of cases within a larger dataset, instances emerge where vowel letters undergo pre-[r] vowel adjustment immediately before <hr>, such as in *Nehru* /néəruː/ and *Tehran* /tèərɑ́ːn/. Given that the <e> in these words are immediately followed by a coda <h> in the orthographic representation, it is reasonable to assume that the vowel letters are initially assigned [eɪ], as in the interjection *eh*, which subsequently changes to [eə] through adjustment by [r]. In a silencing approach, accounting for this situation would necessitate one of two less-than-ideal methods. The first method would be to allow the pre-[r] vowel symbol adjustment rule to apply not only before [r] *but also before a sequence of silent [h] followed by [r]*. Alternatively, the second method involves the removal of silent [h] from the representation *prior to* vowel symbol adjustment. This latter approach would be recognized as a case of extrinsic rule ordering. While strategies from generative phonology, such as displaying silent symbols on separate tiers, are conceivable, they should not be included in the system for learners and instructors, as most of them are not experts in the theoretical field.

3. A New Model

This section introduces a model that addresses the drawbacks of the silencing approach discussed in the last section. This system abolishes the use of rules silencing pronunciation symbols and primarily relies on deletion rules. While some rules have somewhat more stipulative application conditions, efforts have been made to avoid externally specifying the order of rule sequences.

3.1 Database

The rule set presented in this paper is designed to explain the British pronunciation found in Mayor (2009, hereafter referred to as LDOCE5). LDOCE5 is a widely recognized ESL English-English dictionary, and its prominence renders further elaboration unnecessary. Word searches were conducted using the accompanying CD-ROM version and the iOS version purchased separately. The CD-ROM version allows wildcard searches using pronunciation symbols, while the iOS version enables wildcard searches based on spellings. By selecting the entries that appear on the left side of the iOS version's screen during searches, the present author identified 48,084 headwords. It should be noted, however, that unlike in most of the previous studies on related topics by the present author, simultaneous searches based on both text and pronunciation symbols were not conducted in this study.

LDOCE5 employs the following notation system, which is predominantly adopted in this paper:

(1) Symbols Employed

a. Vowels: æ, e, ɪ, ɒ, ʌ, ʊ, i, u, ə, ɚ, ɑː, iː, ɔː, ɜː, uː, juː, aɪ, eɪ, ɔɪ, aʊ, əʊ, eə, ɪə, ʊə

b. Consonants: p, b, t, d, k, g, f, v, θ, ð, s, z, ʃ, ʒ, h, t͡ʃ, d͡ʒ, m, n, ŋ, l, r, w, j

[ɚ] = optional [ə]; The ties in [t͡ʃ, d͡ʒ] are not in LDOCE5.

However, while LDOCE5 employs [ˌ] and [ˈ] for stress representation, this paper utilizes [ˊ] and [ˋ] placed over vowel symbols and vowel letters. Additionally, while LDOCE5 does not provide stress markers for monosyllabic content words, this paper indicates every primary stress with [ˊ], even for monosyllabic words. Furthermore, while the dictionary generally treats word-final <r> or <re> as silent, the present system considers them pronounced when followed by vowel-initial words, representing this situation as '/(r)/'.

3.2 Terms and Concepts

This subsection introduces the concepts and terminology that will be used in the proposed new system.

(2) Kinds of representations

a. <...>= **L-Rep** (Literal Representation)

b. [...] = **S-Rep** (Symbolic Representation)

c. /.../ = **F-Rep** (Final Representation) # Pronunciations found in LDOCE5.

(3) Units

a. Symbol Unit (SU)

(i) A single pronunciation symbol or a combination of pronunciation symbols that collectively

corresponds to one single linguistic sound is referred to as an **SU (Symbol Unit)** : [æ, aʊ, ɪə, juː], etc.

[a], [ː] are symbols, while [æ] functions as both a symbol and a unit simultaneously.

(ii) An SU representing a vowel sound is termed a VSU (Vowel SU), while one that corresponds to a consonant sound is denoted as a CSU (Consonant SU).

b. Letter Unit (LU)

(i) A single character or a combination of characters that corresponds to one single SU is referred to as an **LU (Letter Unit)** : <u, e, o͡a, s͡h, p͡h>, etc. (The same as 'grapheme'.)

When each of these characters corresponds to one single SU, it is considered an LU. The components, namely <o, a, s, p, h>, are referred to as letters.

(ii) An LU that corresponds to a VSU is referred to as a **VLU (Vowel LU)**, and one corresponding to a CSU is abbreviated as a **CLU (Consonant LU)**.

(4) Vowel Symbol Unit Strength (VSUS /víːsʌs/)

a. Weak Vowel Symbol Unit (WkVSU)

A VSU representing a weakly pronounced VSU is abbreviated as a **WkVSU (Weak VSU)** :

[ə, i, u, ɪ, ʊ, əʊ, uː] (Exhaustive)

(The underlined portion indicates the weak versions of the same pronunciation symbols.)

b. Strong Vowel Symbol Unit (StVSU)

A VSU which is not a WkVSU is denoted by a **StVSU (Strong VSU)** : [e, ɒ, ʌ, eɪ], etc.

When a stress marker is absent, word-final [juː, uː] are ambiguous in terms of stress. The same ambiguity exists for [əʊ] as well either when it is either word-final or when it immediately precedes another VSU. For the sake of clarity and convenience, all are treated as a WkSU (in this system).

[Note for Readers] This grouping of vowels is influenced by the categorization presented in Takebayashi (1996: 273-280).

(5) Vowel Letter Unit Strength (VLUS)

a. Strong Vowel Letter Unit (StVLU)

A VLU corresponding to a StVSU is a **StVLU (Strong VLU)** and is indicated by an enclosing box (□).

b. Weak Vowel Letter Unit (WkVLU)

A VLU that is not a StVLU, including a silent VLU, is referred to as a **WkVLU (Weak VLU)**.

WkVLUs may be enclosed with a circle or explicitly marked as "└wk" (wk = weak), but most often, no specific markings are applied.

c. Examples: <pa[nd]ⓐ>[pǽndə], <a[ct]ⓘvⓔ>[æktɪv]
<be[st]>[bést], <b[ir]d>[bɜ́ːrd], <c[it]ⓨ>[síti]
<[a]>, <[e]>, <[i]> = StVLUs; ⓐ, ⓔ, ⓘ, ⓨ = WkVLUs.

(6) Regulation (Reg): The word *regulation* is used as a cover term encompassing various rules and decisions.

(7) Meaning of Regulations (MoR): 'X → Y' signifies the following:

a. "Change X to Y" or b. "Impart the characteristics of Y to X."

3.3 Overview of the System

This subsection presents an overview of the system proposed in this paper, which is influenced by such generative phonological analyses of English phonology as presented by Kiparsky (1982), Halle and Mohanan (1985), Borowsky (1990), and Jensen (2022). The present system houses five rule sets, each corresponding to different types of regulations. Rules are treated as directives to be followed. Thus, instead of using the term *application*, which is typically associated with rule-based generative phonology, the word *execution* is employed.

(8) Rule Group (RG)

A Rule Group (RG) refers to a rule set to be considered when seeking rules to be executed. There are five such distinct sets.

(9) Overview of the Rule Groups

a. RG-1: Preparation rules
• Primary operation: Modification or alteration of L-Reps (e.g., <x>→<ks> for facilitating subsequent rule execution).

b. RG-2: Rules without syllable structure information
• Primary operation: Assignment and modification of CSUs.

c. RG-3: Rules for constructing syllable structures
• Primary operation: Construction of syllable structures

d. RG-4: Other Rules
• Primary operation: Assignment and adjustment of VSUs

e. RG-5:"Fair-copying" rules (in a limited number)
• Primary operation: "Finalization" of S-Reps.

During the decoding process, the RGs are consulted in the order given here when selecting rules for execution.

As mentioned in Section 3.1, the goal of the proposed system is to replicate the pronunciation representations found in LDOCE5. This study shows the potential mental operations required to

reproduce those pronunciation representations, given knowledge of which vowel letters—or sequences thereof—within words are pronounced as full vowels. Decoding processes may introduce deviations from structures produced in typical English phonological systems. For example, some rules sometimes create structures that are phonologically ill-formed, such as [j] being the nucleus of a syllable. However, no corrective measures are implemented unless such irregularities significantly affect the entire decoding process.

3.4 Details of Rule Execution

What follows is a detailed description of the steps involved in rule execution. As mentioned earlier, the instructions are articulated as directives for the decoder. The rules are written under the assumption that learners have knowledge of what decoding means.

(10) Rule Selection and Execution (RSE)

a. Rep Set (RS) and RS Finished (RSfin)

Representations arising during the decoding in this system are mainly comprised of an L-Rep and an S-Rep. This pairing is termed a **Rep Set (RS)**. Once an RS has undergone all executable rules, it is referred to as an **RS Finished** or **RSfin** for short.

At the beginning of the decoding process, each of the target representations solely consists of the word's spelling, enclosed in a pair of angle brackets. For the sake of convenience, these initial states are treated as if they were RSfins.

b. Candidate Search

(i) Rules whose execution conditions are satisfied by RSfins are referred to as **candidates**.

(ii) Once an RSfin is achieved, begin searching for candidates within the available RG. This process is termed **candidate search**. The RG available at the start of the decoding process is RG-1. If no more candidates are found within a given RG, consider the next immediately following RG. Continue this pattern until RG-5 is reached. Do not return to a previous RG or skip one.

c. Approval

For identified candidate rules to be executed, they need **approval**. If a rule is not in rivalry with another, it is automatically approved. Refer to the Rule Strength Condition in (11) below for the definition of rivalry and the approval process for rules in rivalry.

d. Obligatoriness of Execution

Execute all rules that have been approved for the RS under operation. This includes cases where undesired structures, such as [kk], may arise during execution.

e. “Approved-Only” Execution

For a given RSfin, execute only the rules that have been approved for that particular RSfin. Even if the execution of approved rules creates structures upon which other rules can be executed, do not execute these additional rules. Only after the RS under operation becomes an RSfin and a new candidate search is completed, should you consider executing such rules alongside other candidates.

For example, within RG-2, the first rules apprroved for the word *mecca* are rules assigning CSUs. At this stage, execute only these rules. Although their execution produces a sequence of identical CSUs, [kk], do not immediately execute any rule to resolve this. The resolution rule can be executed only after it becomes a candidate and is approved, which occurs following the completion of the CSU assignment operations and the initiation of a new candidate search.

(11) Rule Strength Condition (RSC)

When the following criterion applies to any pair of candidate rules, they are considered to be in **rivalry:**

If one is executed, the RS will no longer satisfy the execution conditions of the other.

When such a pair of candidates arises, retain the **stronger** candidate and discard the weaker one. (By “discard,” it is meant that the candidate is no longer considered for selection.) In any given pair, the stronger candidate is the one with **stricter** execution conditions. Conditions are deemed stricter when they encompass more specific word constituents than others or require more particular structures. If other pairs of candidates are in rivalry, repeat this comparison process as many times as the number of those pairs. Only the candidates that survive this process are approved.

[Note for Readers] This condition is greatly influenced by the Elsewhere Condition by Kiparsky (1982: 8).

4. System Illustration

This section offers step-by-step illustrations of the decoding process as envisaged in the proposed system using specific words. Before delving into the details, it is appropriate to introduce the following provisions, which are integral to the system and apply consistently throughout decoding processes:

(12) General Principle on Structure Assignment (GePSA /dʒépsə/)

Unless otherwise specified, structures (SUs, syllable structure, etc.) assigned by rules are

attached to units that are not marked as silent.

(13) Meaning of "bare"

The term *bare* refers to the state of not having been given a higher-level structure. (This means when either LUs have not been assigned SUs or SUs do not belong to any syllable.)

4.1. Illustration 1

Proceeding to the first actual illustration, it is pertinent to mention that the words below have been selected as examples. Their choice is influenced by space limitations and the inherent moderate complexity they present in explanation.

(14) a. <pártial> /pάːʃəl/ b. <pàrtiálity> /pὰːʃiǽləti/

Note that in these examples, the <ti> portions correspond to /ʃ/ in (a) and to /ʃi/ in (b).

4.1.1 RG-1 (Preparation)

The initial representations for the first candidate searches are as follows:

(15) a. <pártial> b. <pàrtiálity>

Within RG-1, the first rules indentified as candidates for these are the rules in (16) and (17) below:

(16) Letter Unit Clarification (LUC)

a. Boxing (Box): Enclose StVLUs in □.

Examples: <agó> → <ag□o>, <mánner> → <m□anner>, <róad> → <r□oa□d>

b. Tying (Tie):

When consecutive characters form an LU, connect them with '⁀'. However, this is not necessary when they are enclosed in □.

Examples: <phone> → <ph⁀one>, <shine> → <sh⁀ine>, <thread> → <th⁀read>

(17) S-Rep Creation (SRepC /ɛ̀srepsíː/): Provide an empty S-Rep around the L-Rep.

Not in rivalry, theses rules are immediately approved and executed. The process is illustrated in (18):

(18) a. <pártial> ⇒ 1 —LUC, SRepC→ <p□artial> [ ]

b. <pàrtiálity> ⇒ 1 —LUC, SRepC→ <p□artiálity> [ ]

Here, '⇒ 1' denotes the "start of the search of RG-1," and "—LUC, SRepC→" indicates that the rules LUC and SRepC are executed on the representation to the left, leading to the subsequent RS. This presesntational principle also applies to instances that follow. Both sets of representations at the end are RSfins, which means that all approved rules have been executed on them. As a result, new candidate searches commence. No candidates are identified within RG-1. Consequently, candidate searches based on RG-2 begin.

4.1.2 RG-2

In RG-2, the following rules are identified as candidates.

(19) Consonant SU Assignment Default (CDft /siːdéft/)

Assign CSUs to the CLUs situated to the left of an arrow. Treat those processes as separate rules: <c> → [k], <g> → [g], <j> → [d͡ʒ], <q> → [k], <y> → [j]

For the remaining <C>s, excluding <x>, the assignment is the same as the character itself: <b> → [b], <m> → [m], <v> → [v], etc.

When it is necessary to indicate specific rules, a rule that assigns [k] to <q>, for example, is referred to as CAss(q). It should be noted that <x> is excluded here because in this system, that letter is decomposed into <ks> by an RG-1 rule, which is not detailed here. Additionally, <s> very often corresponds to [z] in practice, but this paper primarily uses <s> → [s]. Concerning the two RSs under consideration, CAss(p), CAss(r), CAss(t), and CAss(l) are candidates, and since there is no rivalry among them, all of them are approved and executed. The following changes occur:

(20) a. <p□rtial> [　] ⇒ 2 —CDft→ <p□rtial> [p rt l]

b. <p□rti□lity> [　] ⇒ 2 —CDft→ <p□rti□lity> [p rt l t]

Since the RSs on the right are RSfins, new candidate searches are initiated within RG-2. However, no further candidates are identified, prompting subsequent searches based on RG-3.

4.1.3 RG-3

The rule set directly below is the main part of RG-3. It is concerned with constructing syllable structures. As such, these rules are commonly executed. Each presented rule will be followed by a display of the resulting RS produced upon its execution.

(21) Syllable Structure Construction Rules (SSC)

Consider each of the following rules individually for candidacy, and execute those that are approved. Conduct the processes of consideration, approval, and execution separately for each rule. The order of the rules to be considered is as follows:

PHAss >> Nuc >> Ons >> Cod >> LRepSyll

Continue this procedure until there are no more rules that can be approved (and executed).

[Note for Readers] Repeating the procedure is essential for processing words such as *Khm□r* [•kə̲•méə(r)•]. The initial syllable results from a [ə]-insertion rule, not detailed in this paper, that takes effect only after constructing the second syllable. This necessitates two iterations of the procedure immediately above.

a. Placeholder Assignment (PHAss /fǽs/)

Assign [□]s to StVLUs and [○]s to WkVLUs.

Note that dummy <V>s and silent VLUs are also WkVLUs.

< V~ > → [□], < V~ > → [○]
┌ wk, **can be silent or dummy <V>**

▷ <partial> [p rt l] ⇒ 3 —PHAss→ <partial> [p□rt○○l]

▷ <partiality> [p rt l t] ⇒ 3 —PHAss→ <partiality> [p□rt○□l○t○]

[Note for Readers] Think of placeholders as symbols where all attributes are underspecified, except for syllabicity, vocality, and strength. While it is possible to use [V]s to represent such objects, employing distinct symbols or placeholders can help avoid confusion with the [V] meaning "some kind of VSU" that might appear in execution conditions. A dummy <V> has been incorporated into the execution context to address cases such as <tabVl>, which is derived from <táble> with an RG-1 rule. Although not discussed in this paper, word-final <Cle> and <Cre> are typically replaced respectively with <CVl> and <CVr> by an RG-1 rule for ease of subsequent operation. Furthermore, part of the condition on the WkVLU indicates that this rule is an exception to the general provision GePSA (12), which generally prohibits structure assignment on silent items.

b. Nucleus Formation (Nuc)

Assign the status of a nucleus to all placeholders, including those associated with silent VLUs or dummy <V>s: [□] → ■ / [○] → ●,

where the blackened placeholders denote those with a nucleus status.

▷ <partial> [p□rt○○l] —PHAss→ <partial> [p■rt●●l]

▷ <partiality> [p□rt○□l○t○] —PHAss→ <partiality> [p■rt●■l●t●]

[Note for Readers] While it is possible for PHAss (a) to directly assign blackened placeholders, thereby eliminating this rule from the paper, this approach has not been taken. Ideally, syllable structures are represented in the arboreal form, as shown in (22) below, where nucleus formation is depicted through the assignment of a syllable node and its associated line. Retaining this rule better aligns with this representation.

c. Onset Formation (Ons)

If the word has a single CSU or a sequence of CSUs that typically appear at the beginning of a word, treat those CSUs as part of the nearest syllable to the right. However, when the CSUs in question are sandwiched between nuclei, exclude [sC] sequences.

▷ <partial> [p■rt●●l] —Ons→ <partial> [p■rt●●l]

▷ <partiality> [p■rt●■l●t●] —Ons→ <partiality> [p■rt●■l●t●]

The shaded portions represent onset elements.

[Note for Readers] This rule is presented here in a simplified version. Please refer to Appendix

1 of Yokotani (2023b) for the complete version.

d. Coda Formation (Cod)

Make CSUs part of the nearest syllable to the left if such a syllable is available.

▷ <p[a]rtial> [p■rt●●l] → [•p■r•t●•●l•]

▷ <p[a]rti[á]lity> [p■rt●■l●t●] → [•p■r•t●•■•l●•t●•]

The shaded portions represent coda elements.

e. L-Rep Syllable (LRepSyll /élrepsɪ̀l/)

Create a syllable structure on the L-Rep based on that in the S-Rep.

<p[a]rtial> → <•p[a]r•ti•al•>　　<p[a]rti[a]lity> → <•p[a]r•ti•[a]•li•ty•>

[•p■r•t●•●l•]　[•p■r•t●•●l•]　　[•p■r•t●•■•l●•t●•]　[•p■r•t●•■•l●•t●•]

These are all the syllable structure construction rules presented in this paper. For words such as *partial* and *partiality*, the operations of syllable structure construction conclude at this point. Even if the process is applied again from PHAss, no candidates are identified, which prompts candidate searches based on RG-4.

4.1.4 RG-4

For the examples currently under consideration, candidate searches based on RG-4 are conducted on the LRepSyll outputs, presented below in alternative forms.

(22) a.

		σ			σ	σ				
LRepC	／	|	＼	／	|	|	＼			
	<	p	[a]	r	t	i	a	l	>	
→	[	p	□	r	t	○	○	l	]	
	＼	|	／	＼	|	|	／			
		σ			σ	σ				

b.

		σ			σ	σ		σ		σ		
LRepC	／	|	＼	／	|	|	／	|	／	|	
	<	p	[a]	r	t	i	[a]	l	i	t	y	>
→	[	p	□	r	t	○	□	l	○	t	○	]
	＼	|	／	＼	|	|	＼	|	＼	|	
		σ			σ	σ		σ		σ		

For these RSs, the rule in (23) below is among those identified as candidates. It addresses the VLUs of the <ti> portions, <i>, which exhibit the unique behaviours previously mentioned. The inclusion of [ʃ] and [t͡ʃ] in the contextual CSUs accounts for words such as *fash*i̲*on*, *lunch*e̲*on*, and *trunch*e̲*on*, where <i> and <e> are considered to lack vocalic content analogously to *partial*. The bold text is used solely for emphasis and has no relation to the execution conditions.

(23) <i>-to-Yod (iYod /áɪjɵ̀d/)

a. Main conditions

[s, z, ʃ, t͡ʃ, d͡ʒ] ┐　　┌ from the YWP in (b) below

[　X　●　Y　] → **j**ɪ if [Y] corresponds to <[V~]> (or <o>, <at(e)>)

<i>/<e> ┘ → j otherwise

Prose version: Suppose there is a weak placeholder (‘●’) corresponding to either <i> or <e>.

Let [X] denote the SU immediately preceding this placeholder and [Y] the SU immediately following it. [X] can be one of [s, z, ʃ, t͡ʃ, d͡ʒ], while [Y] matches one of the LU sequences listed in (b) below. If [Y] corresponds to an StVLU, <o> or <at(e)>, then replace the weak placeholder with [jɪ]. Otherwise, replace it with [j]. However, based on the value of [Y], additional SUs might be included in [X]. Refer to (c) for details.

b. The Yod Word Parts (YWP): *-a, -al, -an, -ance/-ence, -ancy/-ency, -ant/-ent, -ary/-ery, -at(e), -o, -on, -o͡us*.

c. Further conditions

(i) [t] can also be part of [X] if [Y] comes from <-a>, <-on>, or <-ous>.

Examples: *digés<u>t</u>ion, Croá<u>t</u>ia, nát<u>i</u>on, cáu<u>t</u>ious.*

[Note for Readers] [t͡ʃ] in the first example is considered to be derived via a rule not discussed here from a sequence of [t] followed by [j]. [ʃ] in the others is arise from a rule discussed below that changes [t] to [ʃ] in certain environments.

(ii) If the weak placeholder is <i>, [X] can also be the following:

- [l, n] if [Y] corresponds to <on> Ex. *onion*, *opinion*, *million*, etc.
- [n] from <ann>, <toman>, <Son> if [Y] corresponds to <a>

Examples: *Britannia*, *kleptomania*, *Sonia* [Exhaustive examples]

There are additional execution conditions beyond (c), but due to space limitations, they are not discussed here. iYod (23) is notably intricate. However, when addressing palatalization/affrication induced by weak <i, e>, it has to be admitted that any precise rule aiming for similar effects, regardless of the approach taken, would require a complexity on a par with the one described above.

Focusing on the rule under discussion, it yields the following S-Reps:

(24) a. [•p■r•tj•●l•] b. [•p■r•tjɪ•■•l●•t●•]

In *partial*, the <a> in *-al* constitutes a WkVLU, leading to the assignment of [j] to the preceding <i> VLU. Conversely, in *partiality*, the relevant <a> portion after <i> is an StVLU, resulting in the allocation of [jɪ] to the preceding weak <i>. Note that in the former, [j] comes to occupy a nucleus position, which is ill-formed from a phonological point of view, but this is not blocked in this system as mentioned at the end of Section 3.3. The current system leans more towards *phonics* rather than traditional English phonology. As such, the formation of such structures in the workflow is not deemed problematic.

Analyses in generative phonology such as Hayes (1982: 267-269) and Jensen (2022: 299) propose that the sound corresponding to the (first) <i> in words such as *partial* and *partiality* is originally [j] in both instances, postulating a rule vocalizing the glide in a specified context ([j]→[ɪ]). In this approach, since [j] needs to disappear after causing palatalization, there must be some kind

of ordering among palatalization, [j]-vocalization, and the deletion of [j]. On the other hand, this paper introduces [jɪ] into the environment from the outset. This eliminates the need for the vocalization operation, thereby reducing the necessity to specify the rule application order.

When considering parts other than <ti>, the following rules, presented in simplified forms, become candidates and, upon approval, are executed. The '◆'s denote conditions on the targets. (See Yokotani (2023b: 22) for the full forms of the rules assigning default StSUs and WkSUs. Rule (25) is based on a rule by Carney (1994: 299, Rule A.31).

(25) Strong <a> Default (S<a>Dft): [■] → æ (◆ <a>)

(26) Weak <i/y> (Wk<i/y>): [●] → ɪ (◆ <i/y>) (or → ə in certain cases)

(27) Weak <a> (Wk<a>): [●] → ə (◆ <a>)

In relation to these, the two regulations below take effect. The notation '1m' in the first denotes '1 mora', and in the second, influenced by rules from Halle and Mohanan (1985: 67, 81),'NIF-[C]'is an abbreviation for 'not immediately followed by a CSU', meaning 'either prevocalic or word-final'.

(28) Vowel Adjustment Pre-[r] 1m (VAR1m)

[1m V] r co	→	ɑː if [V] = [æ]	Ex. <f a r> → [fɑːr]
	→	ɔː if [V] = [ɒ]	Ex. <p o rt> → [pɔːrt]
	→	ɜː otherwise	Ex. <f u r>, <f i r>, <v e rb> → [ɜː]

Prose version: When placing a 1m VSU immediately before coda [r], modify the VSU as follows: change [æ] to [ɑː], [ɒ] to [ɔː], and any other 1m VSU to [ɜː].

(29) Vowel Adjustment High Vowel (VAhv /vάːv/)

a. [ɪ] → i (◆ NIF-[C])　b. [...ʊ] → ...u (◆ NIF-[C])

Prose version: When placing [ɪ] or […ʊ] in NIF-[C] position, modify the VSU or part of it as follows: change [ɪ] to [i] and […ʊ] to […u].

When determining the approved rules for the RSs in (22) above based on RSC (11), there is a rivalry between iYod (23) and Weak <i/y> (26) concerning the vowel portion of <ti>. This is because both of them involve the manipulation of weak <i>. The former yields [j] or [jɪ] depending on the environment, while the latter typically produces [ɪ] consistently. When one rule is executed, the resulting RS no longer meets the execution conditions of the other. This is because once the target placeholder is replaced with an SU or a sequence of SUs, there is no placeholder to change any longer. RSC (11) decides that iYod is the stronger rule. While the execution conditions of Weak <i/y> are not tied to specific word-parts, iYod incorporates several such conditions, particularly regarding the [Y] portion. Thus, in both *partial* and *partiality*, Weak <i/y> is superseded by iYod for the second-syllable <i>.

Concurrently, the rule of Weak <a> assigns [ə] to the weak <a> in the adjective; and the rule of Weak <i/y> (along with its associated VAhv) gives [i] to <y> in the noun, and [ə] to the <i> in the penultimate syllable of the latter. When these rules are executed along with iYod, the RSs in (22) result in those in (30) below. These are the results of these combined operations, including iYod:

(30) a. <•pa r•ti•al•> [•pɑːr•tj•əl•] b. <•pa r•ti•á•li•ty•> [•pɑːr•tji•æ•lə•ti•]

New candidate searches are initiated by the novel RSfins. For both *partial* and *partiality*, the following rule in (31)——a revised version of the one presented in Yokotani (2018: 64)——is identified as a candidate. The transformation in (32) arises from the approval and execution of this rule.

(31) <t>-to-Esh (tEsh /téʃ/)

Not [s] ¬ ⌐ given by iYod (23)
[X t j (ɪ)] → ʃ

Prose version: Change [t] to [ʃ] if the former is not immediately preceded by [s] and is immediately followed by either [j] or [jɪ] that has been given by iYod.

(32) a. <pa rtial> [•pɑːr•tj•əl•] —tEsh→ <pa rtial> [•pɑːr•ʃj•əl•]

b. <pa rti á lity> [•pɑːr•tji•æ•lə•ti•] —tEsh→ <pa rti á lity> [•pɑːr•ʃji•æ•lə•ti•]

Since no other candidates are found, these RSs are RSfins.

Candidate searches are performed again, with only one detected for the RSs immediately above.

(33) Yod-Deletion (YodDel /jʊddèl/)

[C j] → Ø But **Not** [V (r) l j V] → Ø
└ [ʃ, t͡ʃ, d͡ʒ, ʒ, l, r, w, j] stronger ┘ └ weaker

Prose version: Delete [j] when it directly follows any of the SUs [ʃ, t͡ʃ, d͡ʒ, ʒ, l, r, w, j], unless all three of the following conditions are met: (i) C is [l], which is immediately preceded by [V(r)]; (ii) the target [j] is directly followed by a VSU; (iii) the VSUs on either side of the [j] are in a strong-weak relationship.

[Notes for Readers] The CSUs for [C] may seem too numerous for learners to remember. However, based on the author's experience, this challenge can be addressed relatively easily with mnemonic techniques.

It is worth noting that this is a revised version of the one given in Yokotani (2023ab), which mandates [l] to be in the onset based on the observations first made by Borowsky (1990) and utilized by Davis and Hammond (1995). In the previous models, instances where [j] remains in the final output were addressed by a re-syllabification rule mentioned in Section 2. This rule moves [l] to a coda position when it appears in the 'But Not' environment in (33). It

was necessary to ensure that the resyllabifcation applies earlier than the old [j]-deletion/silencing rule. The end was achieved through RSC (11) or an analgous provision.

The revision was prompted by the observation that no other cases truly necessitated the re-syllabification rule. While re-syllabification may provide a more comprehensive account of the phonetic value of [l] in the coda, such detailed phonetic considerations are not pursued in the present system. Allophonic details of this nature can be derived by constructing syllable structures in the F-Rep. Additionally, eliminating the re-syllabification rule simplifies the entire system. The rule can compete with YodDel, which is currently under discussion. Removing the former makes the decoding processes more straightforward, leading to easier learning and teaching.

With no competing candidates, YodDel is immediately approved and executed, resulting in the removal of [j] following [ʃ] in each RS.

(34) a. <•par•ti•al•> [•pɑːr•ʃ •əl•]　b. <•par•ti•á•li•ty•> [•pɑːr•ʃ i•æ•lə•ti•]

These new RSfins trigger new candidate searches.

Of the two RSs presented immediately above, no RG-4 candidates are found for (b), whereas one is identified for (a). This rule, which is still under development, modifies [ə] into a form that might be pronounced or not. It is provided here to illustrate its general working.

(35) Schwa Optionalization (SchwOp /ʃwɒ́p/)　[Incomplete]

┌ strong　　┌ [l, n, m], in coda
[V　C~　ə̲　Y　]　→　ɘ　# [C~] = one or more [C]s

The RS that reflects this change is shown in (a) below, while (b) simply repeats a previous one.

(36) a. <•par•ti•al•> [•pɑːr•ʃ •əl•]　b. <•par•ti•á•li•ty•> [•pɑːr•ʃi•æ•lə•ti•]

A new candidate search is initiated for (a), but no candidates within RG-4 are found, just as was already the case with (b). Consequently, the two RSs in (36) undergo candidate searches based on RG-5, the final Rule Group.

4.1.5 RG-5

Within RG-5, two rules are identified as candidates for both RSs. One rule pertains to the deletion of [r] in the coda, while the other is a“fair-copying” rule specified to be executed at the conclusion of the process. The presentation will begin with the former.

(37) [r]-Deletion (rDel /ɑ́ːdèl/) : [r] → Ø　(◆ in coda, not word-final)

This becomes a candidate for both words, and its execution after approval yields the RSfins below:

(38) a. <•par•ti•al•> [•pɑːr•ʃ•əl•] ⇒ 5 —rDel→ <•par•ti•al•> [•pɑː •ʃ•əl•]

b. <•p[a]r•ti•[á]•li•ty•> [•pɑ:r•ʃi•æ•lə•ti•] ⇒ 5 —rDel→ <•p[a]r•ti•[á]•li•ty•> [•pɑ: •ʃi•æ•lə•ti•]

The final rule in the entire system, provided in (39), marks the conclusion of the decoding processes. The ensuing representations are the final ones.

(39) Fair Copying (FairCop /féəkɒ̀p/) : Perform the following:

a. Create an F-Rep (/.../) adjacent to the S-Rep. Copy the non-stray SUs and their accompanying stress information onto the F-Rep in the order they appear.

b. /**r**/ → (r) (◆ word-final)

(40) a. <•p[a]r•ti•al•> [•pɑ: •ʃ •ə l•] —FairCop→ /pά:ʃəl/

b. <•p[a]r•ti•[á]•li•ty•> [•pɑ: •ʃi•æ•lə•ti•] —FairCop→ /pὰ:ʃiǽləti/

"/(r)/" in (39b) denotes that the final /r/ is pronounced when followed by a word starting with a vowel as mentioned towards the end of Section 3.1. This rule is not relevant to the examples under consideration.

4.2 Illustration 2: *Car* and *Carry*

The cases of *car* and *carry* presented in Section 2 are addressed as follows. The only addition required is a rule placed in RG-2 that deletes the first element of a geminate. This rule not only removes the unnecessary CSU but also silences the associated CLU, preventing it from being reassigned a CSU by CDft (19). It is worth noting that this rule belongs to RG-2, the RG with rules that are executed on an RS whose S-Rep contains nothing but CSUs.

(41) No Geminate (NoGem) [RG-2] < C > → ◈ (= silent C)

[C C] → Ø (◆ CC = geminate)

The following illustrates how this rule works in the processes involved.

(42) a. <cár> ⇒ 1 —LUC, SRepC→ <c[a]r> [ ]

b. ⇒ 2 —CDft→ <c[a]r> [k r]

c. ⇒ 3 —SSC→ <•c[a]r•> [•k■r•]

d. ⇒ 4 —S<a>Dft (25)+VAR1m (28) → <•c[a]r•> [•kά:r•]

e. ⇒ 5 —FairCop→ /kά:(r)/

(43) a. <cárry> ⇒ 1 —LUC, SRepC→ <c[a]rry> [ ]

b. ⇒ 2 —CDft→ <c[a]rry> [k rr] —NoGem→ <c[a]◈ry> [k Ør]

c. ⇒ 3 —SSC→ <•c[a]◈•ry•> [•k■ •r●•]

d. ⇒ 4 —S<a>Dft (25), Wk<i,y> (26)+VAh (29) → <•c[a]◈•ry•> [•kæ •ri•]

e. ⇒ 5 —FairCop→ /kǽri/

It is worth noting that it is not a geminate per se that blocks vowel symbol adjustment, as demonstrated in the case below:

(44) a. <Stárr> (a surname) ⇒ 1 —LUC, SRepC→ <Sta͡rr> [ ]

b. ⇒ 2 —CDft→ <Sta͡rr> [st rr] —NoGem→ <Sta͡r̂r> [st Ør]

c. ⇒ 3 —SCC→ <•Sta͡rr•> [•st■ r•]

d. ⇒ 4 —S<a>Dft (25)+VAR1m (28) → <•Sta͡rr•> [•stɑ́ː r•]

e. ⇒ 5 —FairCop→ /stɑ́ː(r)/

5. Final Remarks

The foregoing discussions have highlighted the following novel aspects of this paper:

(i) The concept of Rep Set Finished (RSfin) explicitly defines the scope and timing of executable rules.

(ii) The Rule Strength Condition in (11) determines the executability of competing rules, thereby minimizing the need for rule ordering. The provision is undeniably much easier for learners to understand compared to conditions similar to Kiparsky's Elsewhere Condition. It must be acknowledged that some rules have stipulative executions, as seen in 'given by iYod' in tEsh (31). However, this level of complexity can be justified when considering the simplification it brings to the overall system.

(iii) The process flows depicted in (43) and (44) demonstrate that geminate sequences are resolved using representations consisting solely of consonant symbols. This obviates the necessity for a sequential ordering of rules between the relevant VSU assignment (and any associated adjustments) and the degemination rule. This observation can be considered a novel finding.

Assigning rules to distinct groups may be seen as a kind of extrinsic ordering. It should be noted, however, that at least in most cases, the assignments are natural ones. For example, RG-2 is placed before the group responsible for syllable structure construction and only houses rules that make no reference to syllable structure. Similarly, RG-4 contains rules for vowel symbol assignment. This arrangement is inevitable because, although it may not be apparent in the present paper, many of these rules are based on the syllable structures of the words under decoding. The preparation and finalization stages of RG-1 and RG-5 are not central to the overall system. Thus, what learners need to remember about the rule grouping is basically "consonant → syllable → vowel" only.

Although parts of it have been alterned in the presnt paper, Yokotani (2023b) pesents a more comprehensive basis of the system with more illustrations. The author hopes those intrested in the topic find hints for their research.

Acknowledgement

This paper is adapted from an unread section of Yokotani (2023b), which was presented at the Tokyo Circle of Phonologists meeting held at International Christian University, Tokyo, on September 23, 2023. Although the relevant section was not presented at the meeting, the author extends gratitude to the participants for their insightful comments and remarks on the general architecture of the present system. Their feedback significantly enhanced the final version of this paper.

References

Borowsky, T. (1990) *Topics in the Lexical Phonology of English*, New York: Garland Pub. (Published version of the same author's 1986 Ph. D. disseration.)

Carney, E. (1994) *A Survey of English Spelling*, London and New York: Routledge.

Davis, S. and M. Hammond (1995) "On the Status of Onglides in American English," *Phonology* 12, 159-182.

Halle, M. and K.P. Mohanan (1985) "Segmental Phonology of Modern English," *Linguistic Inquiry* 16, 57-116.

Hayes, B. (1982) "Extrametricality and English Stress," *Linguistic Inquiry* 13, 227-276.

Jensen, J. T. (2022) *The Lexical and Metrical Phonology of English*, Cambridge: Cambridge University Press.

Kiparsky, P. (1982) "Lexical Morphology and Phonology," In I. S. Yang (ed.) *Linguistics in the Morning Calm*, Seoul: Hanshin, 3-91.

Mayor, M. (2009) *Longman Dictionary of Contemporary English* (5th), Essex: Pearson Education Limited.

Takebayashi, S. (1996) *Eigoonseigaku*, Tokyo: Kenkyusha (『英語音声学』東京：研究社)

Yokotani, T. (2018) "Pronunciation Rules for <e, i>-triggered Palatoalveolarization," *Journal of College of Literature* 59, College of Literature, Aoyama Gakuin University, 51-71.(「<e, i> が引き起こす硬口蓋歯茎化のための発音規則」『紀要』(59), 青山学院大学文学部 , 51-71.)

----(2023a) "English Spelling Decoding Rules for Advanced Learners: A Syllable-Based Model," *Journal of College of Literature* 64, College of Literature, Aoyama Gakuin University, 95-116.(「上級者向け英語スペリング読み替え規則集―音節に基づいたモデル―.『紀要』(64), 青山学院大学文学部 , 95-116.)

----(2023b) "A Basis for English Spelling Decoding Rules for Advanced Learners: A *Deletional* Approach," A talk at Tokyo Circle of Phonologists (TCP), at International Christian University, Tokyo on September 23, 2023. As of March 2024, its handout is retrievable as a PDF titled "Yokotani 2023 TCP" from https://drive.google.com/drive/folders/1S2vaYFN5Qcv0rFOnHQJT_4esPLRTRDSI?usp=drive_link