==========
Language in India www.languageinindia.com ISSN 1930-2940 Vol. 25:4 April 2025
==========

Mathematical Modeling of the Optional Forms Generated by the *Krādi-niyama*

Priyanka Salekar, Ph.D. Research Scholar
Department of Humanities and Social Sciences, IIT Bombay
priyanka.salekar@iitb.ac.in

Malhar Kulkarni, Ph.D.
Department of Humanities and Social Sciences, IIT Bombay, Powai, Mumbai - 400076
malhar@hss.iitb.ac.in

Abstract

Panini, in his *Aṣṭādhyāyī*, tries to capture the usage of the then-spoken language, i.e., *bhāṣā*, in a highly concise manner. The treatise has nearly 3700 *sūtra*-s. Mathematical modeling implies providing a mathematical description of a system or a process of translating a problem into a mathematical format, i.e., with the help of mathematical symbols and mathematical language. Several works have highlighted the similarity between the P's *sūtra* and a mathematical function. Aggarwal & Kulkarni (2020) made a point by modeling the *pratyaya*-s in Sanskrit through functions with the help of Pāṇini's *sūtra*-s. The work remodeled the optional forms using the multi-valued function. We take the effort ahead by presenting the optional forms generated by the *Krādi*-niyama, specifically when added with *pratyaya thal* in the format of a multi-valued function. The present work focuses on the set of four *sūtra*-s from the A viz. A 7.2.13, 61, 62, and 63. It studies the collective understanding drawn out of the four *sūtra*-s by the commentary of *Vaiyākaraṇa-siddhānta-kaumudī*, known as *Krādi-niyama-kārikā*. We have remodeled the optional forms by studying the specific instances and developing verbal roots data sets. Currently, we have covered a group of 38 *aniṭ* verbal roots.

Mathematical modeling will help develop a more profound understanding of language and grammar. This is a sincere effort to preserve the grammatical data in newer and modern technical formats and an attempt to transfer the available data to a format accessible to learners outside the field.

Keywords: optional forms, *krādi-niyama*, multi-valued function, mathematical modeling, *aṣṭādhyāyī*, *pāṇinian* grammar

Introduction

Pāṇini (P), in his Aṣṭādhyāyī (A), tries to capture the usage of then-spoken language, i.e., *bhāṣā* in a highly concise *sūtra* format. Components of these *sūtra*-s are identical to the elements of a mathematical function. In a sense, a root word acts as an input (e.g., *dhātu*), the derivational process (e.g., substitution, sandhi, etc.) serves as the relationship to generate an output, i.e., a *pada*.[1] Thus, there seems to be a systemic similarity between the processing in P's *sūtra*-s and the mathematical functions. This similarity forces us to compare a *sūtra* with a mathematical function and further seek to remodel the *sūtra* or grammatical data in a mathematical or functional format.

Literature Review

A is a formal representation of grammar and a very brilliant one (Mishra, 2019). The introduction of A outside the Sanskrit sphere attracted curiosity towards its technical nature. Modern linguistics acknowledges it as the most complete generative grammar of any language yet written and continues to adopt technical ideas from it (Kiparsky, 1994). Efforts to discuss the similarities between P and modern computation can be traced back to 1967 when P. Z. Ingerman (1967) argued that the Backus-Naur Form could rightly be called the 'Pāṇini-Backus Form'. Further, J. Kadvany (2015), in his article 'Pāṇinian Grammar and Modern Computation' ponders upon the formalism brought out by P and its implications. Shyamsundar (2021) immaculately sums up the work done in this area. This shows that the mathematical nature of P's grammar influenced modern computation thought. On the other hand, it would not be incorrect to say that modern computation and linguistics have too

[1] The *sūtra*-s look for the preconditions in an input environment. The effects produced by *sūtra*-s become part of an ever-evolving environment that may trigger others. (Sohoni & Kulkarni, 2018)

==========
Language in India www.languageinindia.com ISSN 1930-2940 Vol. 25:4 April 2025
Priyanka Salekar, Ph.D. Research Scholar and Malhar Kulkarni, Ph.D.
Mathematical Modeling of the Optional Forms Generated by the *Krādi-niyama*

impacted the approach of Pāṇinian studies. Petersen (2004) intends to explain a grammatical phenomenon (structuring of the *śivasūtra*-s) using mathematical methods.

The development of Morphological analyzers by the University of Hyderabad paved the way for the application of P's grammar to computational systems. P's grammar has employed nuanced tools and techniques. To understand a rule and derivation one must decipher these techniques. Thus, there is a need for arranging grammatical data in newer and modern technical formats, expanding its access to experts outside the Sanskrit sphere (Kulkarni A. & Shukla, 2009). It is in this context we try to represent the grammatical data in non-verbal format i.e., using the language of mathematics[2]. This mathematical modeling aims to present the processes in P's grammar in a simplified and consistent manner.

Inspiration for the idea can be found in the work of Sohoni & Kulkarni (2018) which highlighted the structural similarity between P's rules and mathematical functions and proposed a system. Further, a precursor for the current thought is given by Aggarwal A. & Kulkarni (2020) in their article named 'Treatment of optional forms in Mathematical modeling of Pāṇini' and a thesis by Aggrawal A. (2021) focusing on writing the multi-valued functions denoted by words *vā*, *vibhāṣā* and *anyatarasyām*. We wish to take this work ahead by extending it to the multiple outputs resulting from mechanisms other than the one mentioned above, namely the mention of the sage expressing optionality.

We have focused on understanding the *Krādi-niyama* and modeling the derivation process of verbal roots when added with the suffix *thal*. The combination sometimes gives two or more than two outputs, termed as optional/variable forms. We represent the derivation using a multi-valued function.

Multi-valued Function in A

[2] A modeling language serves the need to pass data and a mathematical model description to a solver in the same way that people, especially mathematicians, describe those problems to each other (Kallrath 2013).

Language in India www.languageinindia.com ISSN 1930-2940 Vol. 25:4 April 2025
Priyanka Salekar, Ph.D. Research Scholar and Malhar Kulkarni, Ph.D.
Mathematical Modeling of the Optional Forms Generated by the *Krādi-niyama*

To understand a multi-valued function, one must first understand what is a single-valued function. The single-valued function is a function that assumes one distinct value as its output. This can be explained as - f (x) = a, where f is the function, x is the input value, and a is the output obtained. Whereas a multi-valued function gives two or more values for the same input. This can be explained as, $f(y) = \begin{cases} a1 \\ a2 \\ a3 \end{cases}$. Here, *f* is the function, y is the input value, and a1, a2, a3 are outputs.

Expression of Optionality by P

Along with use of words *vā*, *vibhāṣā* and *anyatarasyām* mention of other linguists in rule is also perceived as expressing optionality by the tradition. For instance, A 1.1.16 – *sambuddhau śākalyasyetāvanārṣe* । (The final *o* of the vocative singular, before the word *iti* is *pragṛhya* in non-vedic literature, according to sage *śākalya*). According to others it is not *pragṛhya*.

As a result, we have two options of combining word *viṣṇo* with *iti*. One by application of term *pragṛhya*, leading to *prakṛtibhāva* (no change) due to A 6.1.125. Secondly, by not applying the term *pragṛhya*. This creates an environment for application of A 6.1.72 and generates an output – *viṣṇav iti*. Output at this stage becomes input for A 8.3.19, which again mentions option in the name of sage *śākalya*. Thus, by substituting zero (*lopa*) at the place of final *v*, we get *viṣṇa iti*. Optionally, by non-substitution of zero (without *lopa*) we have final v retained - *viṣṇav iti* → *viṣṇaviti*. So, we have three outputs for one input (Fig. 1):

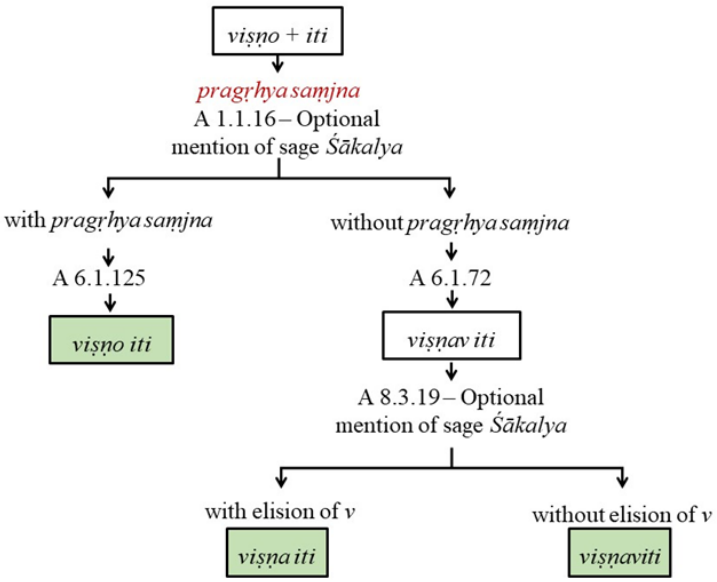

Fig. 1 Expression of optionality by P

P has mentioned 10 linguists by their names specifically – *Āpiśāli*, *Kāśyapa*, *Gārgya*, *Gālava*, *Cākravarmaṇa*, *Bhāradvāja*, *Śākaṭāyana*, *Śākalya*, *Senaka* and *Sphoṭāyana*. There are three references to other grammatical traditions or schools by words - *ācāryāṇām*, *udicām*, *prācām*. One reference is made, which remains unidentified as *eke*.

Traditional scholars have questioned the purpose behind mentioning these grammarians. It is argued that P refers to older authorities whenever he wants to suggest an option for a particular grammatical phenomenon (Bhate, 1970) – *vikalpārtham* (literally, for optional application). However, sometimes it is to show reverence to the predecessors – *pūjārtham* (literally, for reverence).

The total number of references made to linguists is 36. Out of this 19 times, the name is referred to as optionality and 17 times as respect to the predecessor. A 7.2.63 is one such instance where the name of the linguist viz. *Bhāradvāja* is used to express optional application of two previous rules A 7.2.61 and 62. Hence, we have two verbal forms for a single verbal root. Multi-valued function is a way to treat such optional forms.

The *Krādi-niyama*

Niyama is a restrictive rule or a restatement. The restriction or restatement is concerning the previous rule. *Krādi-niyama* comprises four rules from the A – 7.2.13, 61, 62, and 63. A 7.2.10[3] is a negation, which states that verbal roots having only one vowel and *prayogasamavāyī* vowel marked with *anudātta* accent in the initial enunciation are *aniṭ*[4] (without augment *iṭ*), when an *ārdhadhātuka* suffix, beginning with a consonant except for y (viz. *thal, va, ma, se, dhve, vahe, mahe*) follows. So, an *aniṭ* verbal root is never augmented with *iṭ*. A 7.2.13[5] reinstates that only *kṛ, sṛ, bhṛ, vṛ, stu, dru, sru,* and *śru* are not augmented with *iṭ*, in the *Liṭlakāra*. This implies that all other *aniṭ* verbal roots are augmented with *iṭ* in the *Liṭ-lakāra*. A 7.2.61[6] and 62[7] again restrict A 7.2.13 for a small portion – 1. *aniṭ* verbal roots ending in a vowel and 2. *aniṭ* verbal roots having *a* and ending in a consonant, specifically in the case of suffix *thal*. Thus, all other categories of *aniṭ* verbal roots are necessarily augmented with *iṭ* in the *Liṭ-lakāra*. Moreover, the above two types of *aniṭ* verbal roots also have augmentation of *iṭ* for suffixes other than *thal* (viz. *va, ma, se, dhve, vahe, mahe*). Further, A 7.2.63[8] mentions that *aniṭ* verbal roots ending in short *ṛ* are not augmented with *iṭ* when *thal* follows, according to *Bhāradvāja*. Mention of a linguist expresses an optional application of A 7.2.61 and 63. Hence, we have optional augmentation of *iṭ* when thal follows for *aniṭ* verbal roots ending in a vowel and having *a*. Both P and *Bhāradvāja* observe that *aniṭ* verbal roots ending in short *ṛ* are not augmented with *iṭ*.

This is one of the unique instances where negation of negated is implied resulting in a positive statement ultimately. The technique used by P creates complexity. Thus, VSK rearranges the rules[9] and notes collective meaning in the form of a verse known as *Krādi-niyama-kārikā* –

ajanto'kāravānvā yastāsyaniṭ thali veḍayam।

---

[3] *ekāca upadeśe'nudāttāt* । – A 7.2.10

[4] On the other hand, verbal roots having two or more ac (vowels) and a *prayogasamavāyī* vowel marked with *udātta* accent in the initial enunciation (*dhātupāṭha*) are *seṭ* (with augment *iṭ*).

[5] *kṛ-sṛ-bhṛ-vṛ-stu-dru-sru-śruvo liṭi* । - The personal endings of the *liṭ* do not get the augment *iṭ* after *kṛ, sṛ, bhṛ, vṛ, stu, dru, sru* and *śru*.

[6] *acastāsvatthalyaniṭo nityam* । - Suffix thal does not also get the augment *iṭ*, after a verbal root which ends in a vowel and which is always devoid of the *iṭ* augment after the Periphrastic Future affix *tāsi*.

[7] *upadeśe'tvataḥ* । - Suffix *thal* does not get the augment *iṭ*, after a verbal root which possesses a short a as its root-vowel in the *dhātupāṭha* and after which the Periphrastic Future *tās* is always devoid of the augment *iṭ*.

[8] *ṛto bhāradvājasya* । - In the opinion of *Bhāradvāja*, suffix *thal* does not get the augment *iṭ* only after a root which ends in short *ṛ* and after which the Periphrastic Future *tās* is always devoid of the augment *iṭ*.

[9] See 2293, 94, 95 and 96 in VSK

ṛdanta īdṛṅ nityāniṭ krādyanyo liṭi seḍ bhavet ॥ - VSK on 2296

Meaning - That verbal root which ends in *ac* (vowel) or which has *a* in the *dhātupāṭha* and which is *aniṭ* when added with suffix *tās*, is optionally *seṭ*, before suffix *thal*. But if such root ends in short *ṛ*, it is invariably *aniṭ* before the suffix *thal*. Roots other than that are *seṭ* in *Liṭ-lakāra*.

Thus, there are three categories of *aniṭ* verbal roots when it comes to suffix *thal* –

(i) Optional augmentation of *iṭ* – Verbal roots ending in *ac* (vowel) or having *a* in the *dhātupāṭha*

(ii) Without augmentation of *iṭ* – Verbal roots ending in short *ṛ*

(iii) Always with augmentation *iṭ* – All other verbal roots

Meaning of suffix *thal*

The verbal endings (*tiṅ*) in Sanskrit are divided into two categories - *parasmaipada* and *ātmanepada*. These *tiṅ* suffixes are added after a verbal root[10]. *Tiṅ* suffix or function takes *dhātu* (verbal root) as its input and gives *kriyāpada* (verb) as its output. In technical terms this output is called a *pada*[11]. In layman's term *pada* is a verbal utterance which can be used in a sentence.

In *parasmaipada*, the suffix *sip* is added to the verbal root expressing an agent (i.e., Active Voice). Suffix *sip* is used in the sense of IInd Person, and Singular is substituted by *thal* in the *Liṭ-lakāra* (Past Perfect or Pluperfect tense). Just as any other *tiṅ* suffix, *thal* also depicts four meanings – 1. *vācya* (voice) - *kartṛvācya* (active), 2. *kāla* (tense) - *bhūtakāla* (Past Perfect Tense), 3. *puruṣa* (person) - *madhyama* (IInd) and 4. *vacana* (number) - *ekavacana* (Singular). One must remember that these senses are in addition to the action expressed by a verbal root to which the suffix is added. For example, *tvaṃ cikretha* or *cikrayitha* । - You had bought.

Methodology & Notation

[10] *dhātoḥ* । - A 3.1.91

[11] *suptiṅantaṃ padam* । - A 1.4.14

We try to remodel the data produced by *krādi-niyama-kārikā* for which we referred to the list of *aniṭ* verbal roots (*aniṭkārikās*) by the VSK. Verbal roots (ending in a consonant) having vowel *a* and ending in a vowel other than the ones mentioned in *aniṭkārikās*[12] were noted, and forms were studied with the help VSK (Sathe, 1968), *bṛhaddhātukusumākara* by Harekant Mishra (1999), *Iḍāgamaḥ* by Pushpa Dikshit (Dikshit, 2010) and Verb-form generator available at sanskrit.uohyd.ac.in (Kulkarni A., 2002-22). Out of 167 *aniṭ* verbal roots ending in a vowel, 109 have optional augmentation of *iṭ*. One must note that the verbal root *vī* substituted in the place of the verbal root *aj*[13] also ends in a vowel. So, they are a total of 110 in number. We have selected 7 sample verbal roots for representation. These 7 verbal roots form Set-1. Set-2 consists of 31 verbal roots ending in a consonant and having vowel *a*. We noted down the derivative process of these 38 verbal roots dictated by P's rules in detail and tried to reproduce the stages in mathematical format (See Appendix). Verbal roots having the exact same derivation are grouped together under one set. In mathematical representation, we note the verbal roots or suffixes without *anubandhas*[14]. *Anubandhas* do play an important role in P's grammatical system but are not mentioned here, considering the ease of writing an equation. E.g., *i* for *iṭ*, *tha* for *thal*.

The mathematical representation method adopted here was proposed by Aggarwal & Kulkarni (2020). The work served as the terminus a quo for us. The following are the notations –

Symbol	**Represents**
X	Input (a verbal root here in this case)
€	Is an element of - x € A
F	Function (suffix to be added i.e., *thal*)
{}	Set with element – {a, b, c}
→	Converts to
[]	Conversion or step in the equation
c'	Consonant
v'	Vowel

[12] VSK on rule A 7.1.5
[13] *ajervyaghañapoḥ* । - A 2.4.56
14 A 1.3.2 - 9

Ø	Zero
≠	Not equals to
W	String or the unit
Any number, e.g., 1	Position of the operation from left to right
Number with horizontal bar, e.g., $\bar{2}$	Position of the operation from right to left
'+' operator	*sandhi* and then *saṃhitā*

Table 1: Notations

While noting the conversion, each syllable in the verbal root is counted separately. Suppose x= *śak*, the first syllable is ś, the second is "*a*", and the third is "*k"*.

Pre-requisites

Saṃhitā (proximity between the words) is presumed here. Two elements viz. *prakṛti* (herein *dhātu*) and *pratyaya* (suffix) are always in close proximity with each other. At the stage of *saṃhitā*, a vowel which follows a consonant is invariably combined with the it and both are written together. For example, *k + i = ki, n + i = ni, v + i = vi, gl + i = gli*, etc. Same is the case with conjunct-consonants. Consonants coming after one another (without any obstacle of vowel) are to be combined with each other. For example, *k + tha = ktha, d + dha = ddha, ṅ +k + tha = ṅktha, n + tha = ntha, p + tha = ptha*, etc.

Saṃhitā is a pre-condition for *sandhi* [15] (euphonic combination). We have here focused on modeling of the *krādi-niyama-kārikā* only; thus, we will not be dealing with modeling of the *sandhi niyama-s*. Although, we cognize that *sandhi* is an integral part of the derivation process, without which the elements kept together will not be united in true sense and will not be called *pada*. Hence, we are giving a list of *sandhi* cases we have observed through the examples –

e + i = ayi *o + i = avi*

m + tha = ntha *c / j + tha = ktha* *d + tha = ttha*

dh + tha = ddha *bh + tha = bdha* *ñj / sj / snj + tha = ṅktha*

15 *saṃhitāyām* । - A 6.1.72

ś / ṣ / cch / sj + tha = ṣṭha

The cases for specific verbal roots -

vah - uvah + tha = uvoḍha

nah - nanah + tha = nanaddha

dah - dadah + tha = dadagdha

This is not an exhaustive list of instances where the '+' operator can be used in Sanskrit; newer instances can be added as we study more data. It is to be noted that the '+' operator is used for denoting both – *saṃhitā* (close proximity[16]) and *sandhi* (close proximity resulting in a euphonic combination).

Multi-valued Function for Suffix *thal*

We here attempt to present the multi-valued function for the first category of *aniṭ* verbal roots when followed by the suffix *thal*. Each condition is considered a separate set. Hence, we will be dealing with two sets of *aniṭ* verbal roots –

(i) *aniṭ* verbal roots ending in a vowel - *krī, glai (glā), ci, dā, nī, vī, hu*

(ii) *aniṭ* verbal roots ending in a consonant and having vowel *a* - *ad, gam, tap, tyaj, dah, daṃś, nam, nah, pac, pracch, bhaj, bhañj, bhrasj (bhrajj), masj (majj), yaj, yabh, yam, rañj, vac, vap, vas, vah, vyadh, śak, śad, śap, sad, sanṃj, skand, svap, han*

Now, let us look at the mathematical modeling of the same using the multi-valued function. Let A be a set of *aniṭ* verbal roots ending in a vowel except for *ṛ*, A = (*ci, nī, vī, krī, hu, dā, glai*[17]). Here, the first branch of the equation describes representation without augmentation of *iṭ (i).* The second branch of the equation denotes representation with augment *iṭ*.

(a) Aniṭ verbal roots ending in a vowel

(i) Case I – ending in *i*

If x € { *ci* }, then

16 This can be compared with concatenation – a technical term in programming languages used for combining a string, text, or other data in a series, without any gaps.

17 Here after noted as *glā* – A 6.1.45

$$thal\ (\mathrm{x}) = \begin{cases} ci + ci\left[i \xrightarrow{\bar{1}} e\right] + tha \\ ci + ci\left[i \xrightarrow{\bar{1}} e\right] + i + tha \\ ci + ci\left[c \to k\right]\left[i \xrightarrow{\bar{1}} e\right] + tha \\ ci + ci\left[c \to k\right]\left[i \xrightarrow{\bar{1}} e\right] + i + tha \end{cases}$$

ci	*ci*	$ci\left[i \xrightarrow{\bar{1}} e\right]$	+	*thal* (x)
	ci	*ce*	*ci+ce+tha*	*cicetha*
	ci	*ce*	*ci+ce+i+tha*	*cicayitha*
	ci	$ci\left[c \to k\right]\left[i \xrightarrow{\bar{1}} e\right]$	+	*thal* (x)
	ci	*ke*	*ci+ke+tha*	*ciketha*
	ci	*ke*	*ci+ke+i+tha*	*cikayitha*

Table 2: Verbal root *ci*

(ii) Case II – ending in *ī*

If x ∈ {*nī*}, then

$$thal\ (\mathrm{x}) = \begin{cases} n\bar{\imath}\left[\bar{\imath} \to i\right] + n\bar{\imath}\left[i \xrightarrow{\bar{1}} e\right] + tha \\ n\bar{\imath}\left[\bar{\imath} \to i\right] + n\bar{\imath}\left[i \xrightarrow{\bar{1}} e\right] + i + tha \end{cases}$$

nī	$n\bar{\imath}\left[\bar{\imath} \to i\right]$	$n\bar{\imath}\left[i \xrightarrow{\bar{1}} e\right]$	+	*thal* (x)
	ni	*ne*	*ni+ne+tha*	*ninetha*
	ni	*ne*	*ni+ne+i+tha*	*ninayitha*

Table 3: Verbal roots *nī, vī*

Similarly, *thal* (*vi*) = *vivetha, vivayitha*

(iii) Case III - ending in *ī* and having consonant cluster

If x ∈ {*krī*}, then

$$thal\ (\mathrm{x}) = \begin{cases} kr\bar{\imath}\left[\bar{\imath} \to i\right]\left[c' \xrightarrow{\neq 1} \emptyset\right]\left[k \to c\right] + kr\bar{\imath}\left[i \xrightarrow{\bar{1}} e\right] + tha \\ kr\bar{\imath}\left[\bar{\imath} \to i\right]\left[c' \xrightarrow{\neq 1} \emptyset\right]\left[k \to c\right] + kr\bar{\imath}\left[i \xrightarrow{\bar{1}} e\right] + i + tha \end{cases}$$

krī	$krī\left[ī \to i\right]\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[k \to c\right]$	$krī\left[i \overset{1}{\to} e\right]$	+	*thal* (x)
	ci	*kre*	*ci+kre+tha*	*cikretha*
	ci	*kre*	*ci+kre+i+tha*	*cikrayitha*

Table 4: Verbal root *krī*

(iv) Case IV - ending in *u*

If x ∈ $\{hu\}$, then

$$thal\ (\mathrm{x}) = \begin{cases} hu\left[h \to jh\right]\left[jh \to j\right] + hu\left[u \overset{1}{\to} o\right] + tha \\ hu\left[h \to jh\right]\left[jh \to j\right] + hu\left[u \overset{1}{\to} o\right] + i + tha \end{cases}$$

hu	$hu\left[h \to jh\right]\left[jh \to j\right]$	$hu\left[u \overset{1}{\to} o\right]$	+	*thal* (x)
	ju	*ho*	*ju+ho+tha*	*juhotha*
	ju	*ho*	*ju+ho+i+tha*	*juhavitha*

Table 5: verbal root *hu*

(v) Case V - ending in *ā*

If x ∈ $\{dā\}$, then

$$thal\ (\mathrm{x}) = \begin{cases} dā\left[ā \to a\right] + dā + tha \\ dā\left[ā \to a\right] + dā\left[ā \to \emptyset\right] + i + tha \end{cases}$$

dā	$dā\left[ā \to a\right]$	*dā*	+	*thal* (x)
	da	*dā*	*da+dā+tha*	*dadātha*
	$dā\left[ā \to a\right]$	$dā\left[ā \to \emptyset\right]$	+	*thal* (x)
	da	*d*	*da+d+i+tha*	*daditha*

Table 6: Verbal root *dā*

(vi) Case VI -

If x ∈ {*glā*}, then

$$thal\ (x) = \begin{cases} gl\bar{a}\left[\bar{a} \to a\right]\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right] + gl\bar{a} + tha \\ gl\bar{a}\left[\bar{a} \to \bar{a}\right]\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right] + gl\bar{a}\left[\bar{a} \to \emptyset\right] + i + tha \end{cases}$$

glā	$gl\bar{a}\left[\bar{a} \to a\right]\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right]$	*glā*	+	*thal* (x)
	ja	*glā*	*ja+glā+tha*	*jaglātha*
	$gl\bar{a}\left[\bar{a} \to a\right]\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right]$	$gl\bar{a}\left[\bar{a} \to \emptyset\right]$	+	*thal* (x)
	ja	*gl*	*ja+gl+i+tha*	*jaglitha*

Table 7: Verbal root *glā*

(b) Aniṭ verbal roots ending in a consonant

(i) Case I –

If x ∈ {*gam*}, then

$$thal\ (\mathrm{x}) = \begin{cases} gam\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right] + gam + tha \\ gam\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right] + gam + i + tha \end{cases}$$

gam	$gam\left[\mathrm{C} \overset{\neq 1}{\to} \emptyset\right]\left[g \to j\right]$	*gam*	+	*thal* (x)
	ja	*gam*	*ja+gam+tha*	*jagantha*
	ja	*gam*	*ja+gam+i+tha*	*jagamitha*

Table 8: Verbal root *gam*

(ii) Case II -

If x ∈ {*nam*}, then

$$thal\ (\mathrm{x}) = \begin{cases} nam\left[c' \overset{\neq 1}{\to} \emptyset\right] + nam + tha \\ nam\left[c' \overset{\neq 1}{\to} \emptyset\right]\left[\mathrm{W} \overset{1}{\to} \emptyset\right] + nam\left[a \overset{5}{\to} e\right] + i + tha \end{cases}$$

nam	$nam\left[c' \overset{\neq 1}{\to} \emptyset\right]$	*nam*	+	*thal* (x)

	na	*nam*	*na+nam+tha*	*nanantha*
	$nam\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[\mathrm{W} \rightarrow \emptyset\right]$	$nam\left[a \rightarrow e\right]$	+	*thal* (x)
	-	*nem*	*nem+i+tha*	*nemitha*

Table 9: Verbal roots *nam*

Similarly, *thal* (*yam) = yayantha, yemitha; thal* (*tap) = tataptha, tepitha;*

thal (*śak) = śaśaktha, śekitha; thal* (*śap) = śaśaptha, śepitha;*

thal (*pac) = papactha, pecitha; thal* (*śad) = śaśattha, śeditha;*

thal (*sad) = sasattha, seditha; thal* (*yabh) = yayabdha, yebhitha;*

thal (*dah) = dadagdha, dehitha;thal* (*nah) = nanaddha, nehitha;*

(iii) Case III –

If x ∈ {*bhaj*}, then

$$thal\ (\mathrm{x}) = \begin{cases} bhaj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right] + bhaj + tha \\ bhaj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[\mathrm{W} \rightarrow \emptyset\right] + bhaj\left[a \rightarrow e\right] + i + tha \end{cases}$$

bhaj	$bhaj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right]$	*bhaj*	+	*thal* (x)
	ba	*bhaj*	*ba+bhaj+tha*	*babhaktha*
	$bhaj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[W \rightarrow \emptyset\right]$	$bhaj\left[a \rightarrow e\right]$		
	-	*bhej*	*bhej+i+tha*	*bhejitha*

Table 10: Verbal root *bhaj*

(iv) Case IV –

If x ∈ {*han*}, then

$$thal\ (\mathrm{x}) = \begin{cases} han\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[h \rightarrow jh\right]\left[jh \rightarrow j\right] + han\left[h \rightarrow gh\right] + tha \\ han\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[h \rightarrow jh\right]\left[jh \rightarrow j\right] + han\left[h \rightarrow gh\right] + i + tha \end{cases}$$

han	$han\left[\mathrm{C}\overset{\neq 1}{\rightarrow}\emptyset\right]\left[h\rightarrow jh\right]\left[jh\rightarrow j\right]$	$han\left[h\rightarrow gh\right]$	+	thal (x)
	Ja	*ghan*	*ja+ghan+tha*	*jaghantha*
	Ja	*ghan*	*ja+ghan+i+tha*	*jaghanitha*

Table 11: Verbal root *han*

(v) Case V –

If x ∈ {*bhañj*}, then –

$$thal\ (\mathrm{x}) = \begin{cases} bhañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[bh\rightarrow b\right]+bhañj+tha \\ bhañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[bh\rightarrow b\right]+bhañj+i+tha \end{cases}$$

bhañj	$bhañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[bh\rightarrow b\right]$	*bhañj*	+	*thal* (x)
	ba	*bhañj*	*ba+bhañj+tha*	*babhaṅktha*
	ba	*bhañj*	*ba+bhañj+i+tha*	*babhañjitha*

Table 12: Verbal root *bhañj*

(vi) Case VI –

If x ∈ {*rañj*}, then –

$$thal\ (\mathrm{x}) = \begin{cases} rañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]+rañj+tha \\ rañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]+rañj+i+tha \end{cases}$$

rañj	$rañj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]$	*rañj*	+	*thal* (x)
	ra	*rañj*	*ra+rañj+tha*	*raraṅktha*
	ra	*rañj*	*ra+rañj+i+tha*	*rarañjitha*

Table 13: Verbal roots *rañj*

Similarly, *thal* (*sañj*) = *sasaṅktha, sasañjitha*

(vii) Case VII –

If x ∈ {*yaj*}, then

$$thal\ (\mathrm{x}) = \begin{cases} yaj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[ya\rightarrow i\ a\rightarrow i\right]+yaj+tha \\ yaj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[ya\rightarrow i\ a\rightarrow i\right]+yaj+i+tha \end{cases}$$

yaj	$yaj\left[\mathrm{C}\overset{\neq 1}{\rightarrow}\emptyset\right]\left[ya\rightarrow i\ a\rightarrow i\right]$	*yaj*	+	*thal* (x)
	i	*yaj*	*i+yaj+tha*	*iyaṣṭha*
	i	*yaj*	*i+yaj+i+tha*	*iyajitha*

Table 14: Verbal root *yaj*

(viii) Case VIII –

If x ∈ {*vac*, *vap*, *vas*, *vah*}, then

$$thal\ (\mathrm{x}) = \begin{cases} vac\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[va\rightarrow u\ a\rightarrow a\right]+vac+tha \\ vac\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[va\rightarrow u\ a\rightarrow a\right]+vac+i+tha \end{cases}$$

vac	$vac\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]\left[va\rightarrow u\ a\rightarrow u\right]$	*vac*	+	*thal* (x)
	u	*vac*	*u+vac+tha*	*uvaktha*
	u	*vac*	*u+vac+i+tha*	*uvacitha*
vap	Same as above		*u+vap+tha*	*uvaptha*
			u+vap+i+tha	*uvapitha*
vas	Same as above		*u+vas+tha*	*uvastha*
			u+vas+i+tha	*uvasitha*
vah	Same as above		*u+vah+tha*	*uvoḍha*
			u+vah+i+tha	*uvahitha*

Table 15: Verbal roots *vac*, *vap*, *vas*, *vah*

(ix) Case – IX

If x ∈ {*tyaj*}, then

$$thal\ (\mathrm{x}) = \begin{cases} tyaj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]+tyaj+tha \\ tyaj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]+tyaj+i+tha \end{cases}$$

tyaj	$tyaj\left[c'\overset{\neq 1}{\rightarrow}\emptyset\right]$	*tyaj*	+	*thal (x)*

	ta	*tyaj*	*ta+tyaj+tha*	*tatyaktha*
	ta	*tyaj*	*ta+tyaj+i+tha*	*tatyajitha*

Table 16: Verbal root tyaj

(x) Case – X

If x ∈ {*skand*}, then

$$thal\ (\mathrm{x}) = \begin{cases} skand\left[c' \overset{\neq 2}{\rightarrow} \emptyset\right]\left[k \rightarrow c\right] + skand + tha \\ skand\left[c' \overset{\neq 2}{\rightarrow} \emptyset\right]\left[k \rightarrow c\right] + skand + i + tha \end{cases}$$

Skand	$skand\left[c' \overset{\neq 2}{\rightarrow} \emptyset\right]\left[k \rightarrow c\right]$	*skand*	+	*thal (x)*
	ca	*skand*	*ca+skand+tha*	*caskanttha*
	ca	*skand*	*ca+skand+i+tha*	*caskanditha*

Table 17: Verbal root *skand*

(xi) Case – XI

If x ∈ {*daṃś*}, then

$$thal\ (\mathrm{x}) = \begin{cases} daṃś\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + daṃś + tha \\ daṃś\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + daṃś + i + tha \end{cases}$$

daṃś	$daṃś\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]$	*daṃś*	+	*thal* (x)
	da	*daṃś*	*da+daṃś+tha*	*dadaṃṣṭha*
	da	*daṃś*	*da+daṃś+i+tha*	*dadaṃśitha*

Table 18: Verbal root *daṃś*

(xii) Case – XII

If x ∈ {*pracch*}, then

$$thal\ (\mathrm{x}) = \begin{cases} pracch\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + pracch + tha \\ pracch\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + pracch + i + tha \end{cases}$$

Pracch	$pracch\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]$	*pracch*	+	*thal* (x)
	pa	*pracch*	*pa+pracch+tha*	*paprașṭha*

	pa	*pracch*	*pa+pracch+i+tha*	*papracchitha*

Table 19: Verbal root *pracch*

(xiii) Case – XIII

If x ∈ {*bhrasj*}, then

$$thal\ (\mathrm{x}) = \begin{cases} bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right] + bhrasj + tha \\ bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right] + bhrasj + i + tha \\ bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right] + bhrasj\left[\mathrm{W} \rightarrow bharj\right] + tha \\ bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right] + bhrasj\left[\mathrm{W} \rightarrow bharj\right] + i + tha \end{cases}$$

bhrasj	$bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right]$	*bhrasj*	+	*thal* (x)
	Ba	*bhrasj*	*ba+bhrasj+tha*	*babhraṣṭha*
	ba	*bhrasj*	*ba+bhrasj+i+tha*	*babhrajjitha*
	$bhrasj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[bh \rightarrow b\right]$	$bhrasj\left[W \rightarrow bharj\right]$	+	*thal (x)*
	Ba	*bharj*	*ba+bharj+tha*	*babharṣṭha*
	Ba	*bharj*	*ba+bharj+i+tha*	*babharjitha*

Table 20: Verbal root *bhrasj*

(xiv) Case – XIV

If x ∈ {*masj*}, then –

$$thal\ (\mathrm{x}) = \begin{cases} masj\left[\emptyset \overset{\bar{2}}{\rightarrow} n\right]\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + masj\left[\emptyset \overset{\bar{2}}{\rightarrow} n\right] + tha \\ masj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right] + masj + i + tha \end{cases}$$

masj	$masj\left[\emptyset \overset{\bar{2}}{\rightarrow} n\right]\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]$	$masj\left[\emptyset \overset{\bar{2}}{\rightarrow} n\right]$	+	*thal* (x)
	Ma	*masnj*	*ma+masnj+tha*	*mamaṅktha*

	$masj\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]$	*masj*	+	*thal* (x)
	Ma	*masj*	*ma+masj+i+tha*	*mamajjitha*

Table 21: Verbal root *masj*

(xv) Case – XV

If x ∈ {*vyadh*}, then –

$$thal\ (\text{x}) = \begin{cases} vyadh\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[ya \rightarrow i\ a \rightarrow i\right] + vyadh + tha \\ vyadh\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[ya \rightarrow i\ a \rightarrow i\right] + vyadh + i + tha \end{cases}$$

vyadh	$vyadh\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[ya \rightarrow i\ a \rightarrow i\right]$	*vyadh*	+	*thal* (x)
	Vi	*vyadh*	*vi+vyadh+tha*	*vivyaddha*
	Vi	*vyadh*	*vi+vyadh+i+tha*	*vivyadhitha*

Table 22: Verbal root *vyadh*

(xvi) Case – XVI

If x ∈ {*svap*}, then –

$$thal\ (\text{x}) = \begin{cases} svap\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[va \rightarrow u\ a \rightarrow u\right] + svap + tha \\ svap\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[va \rightarrow u\ a \rightarrow u\right] + svap + i + tha \end{cases}$$

svap	$svap\left[c' \overset{\neq 1}{\rightarrow} \emptyset\right]\left[va \rightarrow u\ a \rightarrow u\right]$	*svap*	*'+' operator*	*thal* (x)
	Su	*svap*	*su+svap+tha*	*suṣvaptha*
	Su	*svap*	*su+svap+i+tha*	*suṣvapitha*

Table 23: Verbal root *svap*

Observations

From this, it is evident that –

(i) P's grammar has ingrained a concept called 'Mathematics of language'.

(ii) P attempted to model the Sanskrit language using the meta-language Sanskrit in the best possible way. Here, we attempted to remodel the grammatical data using the language of mathematics – mathematical symbols and equation method.

(iii) We treated optional forms using the functional approach, with the help of multi-valued functions. Doing so gave us new insights into the mathematical nature of the P's grammar.

(iv) This further brought into light the necessity of concepts such as *dvitva* (duplication), *abhyāsa* (the duplicated section), and notion of *aṅga* as well as *antya* (the ultimate syllable) being a by-default location of the conversion[18] denoted by the left-word arrow and number on top ($\overset{1}{\leftarrow}$).

(v) Understanding the rules and meta-rules (paribhāṣā) helps combine different cases and form a more general case. For example, Case II – {*nam, yam, tap, śak, śap, pac, śad, sad, nah, dah, yabh*}.

(vi) Sandhi plays a vital role. Modeling becomes a less tedious task if we consider sandhi-s as a separate function. This helps in combining the cases as well. For example, Case VII - {*vac, vap, vas, vah*}.

(vii) The mathematical modeling, in turn, re-emphasized the reason behind the employment of such techniques by P, i.e., to ensure brevity and form a more generalized system covering similar instances.

(viii) Mathematical modeling in this way helps identify some general patterns dependent upon the occurrence of specific syllables at certain places. The cases in the functions are grouped separately and defined as subsets, following the patterns noticed.

Concluding Remarks

We have noted that P's *sūtra*-s share similarity with the mathematical function, where there is - 1. Input in the form of a root word (e.g., *dhātu, prātipadika*), 2. Outputs (one or more than one) in the form of a *pada* or *vākya* , 3. The relationship in the form of the derivational process explained by the *sūtra* (e.g., substitution, *sandhi*, etc.) which generates the output, 4. The *sūtra* is triggered only when the conditions are fulfilled successfully, 5. The output thus produced

18 *alo'ntyasya* । - A 1.1.52

may become an input at next stage and trigger *sūtra*-s further and the process continues until there remains no rule which is fit to be operational.

Implications

This mathematical modeling positively impacts our understanding of the language and grammar. The advantage of mathematical models is that they can be analyzed precisely using mathematical theory and algorithms (Schichl, 2013). Mathematical modeling is a step towards achieving an acute understanding of the workings of P's grammar and preservation of the grammatical data in formats accessible to learners outside the field. The mathematical model can also form a base for further processing of the grammatical rules for natural language processing of the language with the help of well-defined input and output sets (Aggrawal & Kulkarni, 2018). Efforts can be made in this direction. There are 10 tenses in Sanskrit – *laṭ, liṭ, luṭ, lṛṭ, loṭ, laṅ, vidhiliṅ, āśīrliṅ, luṅ* and *lṛṅ*. This work attempted mathematical modeling of a small section, namely an exceptional case in *liṭ-lakāra*, i.e., *krādiniyama*. We look forward to studying and remodeling all the *lakāra* (tenses) in Sanskrit following the grammatical derivation dictated by P's rules.

==========

Acknowledgment

Our deepest gratitude to Prof. Swapneel Mahajan, Department of Mathematics, IIT Bombay. Discussions with him helped us understand the mathematical concepts better, shaped the representation, and gave new insights into the subject.

==========

References

Aggarwal A. & M. Kulkarni., 2020. "Treatment of optional forms in Mathematical modelling of Pāṇini". In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pp. 15–21, Indian Institute of Technology Patna, Patna, India. NLP Association of India (NLPAI).

Vasu, S. C. 1905. *The Siddhanta Kaumudi of Bhattoji Dikshita*. The Panini office, Allahabad.

Mishra A. 2019. Modeling the Pāṇinian System of Sanskrit Grammar. Heidelberg University Publishing. [doi: https://doi.org/10.17885/heiup.294.395]

Kiparsky, P. 1994. Pāṇinian Linguistics, in R.E. Asher, ed., *Encyclopedia of Language and*

==========

Linguistics, pp. 2918-2923. Oxford, New York: Pergamon Press.

Kadvany, J. 2015. Pāṇinian Grammar and Modern Computation. *History and Philosophy of Logic*. pp. 325-346. Taylor & Francis.

Shyamsunder, R. K. 2021. "Computing Processes and Pāṇini's Aṣṭādhyāyī". *Bhāvanā -The mathematics magazine*. Vol. 5, Issue 3.

Petersen, Wiebke. "A Mathematical Analysis of Pāṇini's Śivasūtras." *Journal of Logic, Language, and Information*, vol. 13, no. 4, 2004, pp. 471–89. *JSTOR*. [URL: http://www.jstor.org/stable/40180377]

Kulkarni, A. & Shukl, D. 2009. "Sanskrit Morphological Analyser: Some Issues". Special issue (in the Festscrift volume of Bh. Krishnamoorty). *Indian Linguistics*. pp. 169–177.

Sohoni, S., & Kulkarni, M. 2018. "A Functional Core for the Computational Aṣṭādhyāyī". *The 17th World Sanskrit Conference. Computational, Sanskrit and Digital Humanities.* Huet, Gérard; Kulkarni, Amba (Ed.), pp. 1-30. University of British Columbia Vancouver, Canada. [doi: https://dx.doi.org/10.14288/1.0391834]

Kallrath, J. 2013. Modeling Languages in Mathematical Optimization. *Springer Science & Business Media*. Springer New York, NY. [doi: https://doi.org/10.1007/978-1-4613-0215-5]

Ingerman, P.F. 1967. *Pāṇini Backus Form Suggested*, Communications of ACM, Vol. 10, 3, pp. 137.

Aggrawal, A., 2021. Mathematical functions in Pāṇinian Grammar. Thesis submitted to Department of Humanities and Social Sciences, IIT Bombay.

Bhate, S. 1970. Pre-Pāṇinian Grammatical elements in Pāṇini's Aṣṭādhyāyī, University of Poona, Poona, pp. 490-508.

Sathe, M. D. 1968. *Vaiyākaraṇa-Siddhānta-Kaumudī of Bhattoji Dixit.* Sanskrit Vidya Parisanstha, Poona 2.

Mishra, H. 1999. *Bṛhaddhātukusumakara*. Chaukhamba Sanskrit Pratisthana, Delhi 07.

Dikshita, P. 2010. *Idāgamaḥ*. Samskrita Bharati, New Delhi 55.

Kulkarni, A. 2002-22. Tool for Verb-form generator. University of Hyderabad. [doi: https://sanskrit.uohyd.ac.in/scl/]

Pandey, G. 2017. *Aṣṭādhyāyī of Pāṇini.* Chaukhamba Surabharati Prakashana. Varanasi 01.

Saxena, S. & Agrawal, R. 2013. Sanskrit as a Programming Language and Natural Language

Processing. *Global Journal of Management and Business Studies* 3 (10). pp. 1135–1142.

==========

Appendix: Example of *thal* function by Panini's *sūtra*-s

As a part of our methodology, we prepared a list of aniṭ verbal roots based on their endings, noting the details of class (*gaṇa*) and *pada*. Following this, we wrote a detailed derivation of every verbal root and then the mathematical representation of it. The mathematical equation represents stages in the derivation starting from the augmentation or non-augmentation of *iṭ*. We note down the derivation process of three verbal roots viz. *ci* (ending in a vowel), *gam* (ending in a consonant), and *bhañj* (having consonant cluster) hereafter –

(a) *Aniṭ* verbal root ending in vowel *i* –

(i) *ci - ciñ cayane* । (*svādi* – 1251)

Steps	**Applicable rules**
ciñ	*ciñ cayane* । [19] – to collect
ci	*halantyam* । (1.3.3) In the initial enunciation, the final *hal* (consonant) is termed *it*. *tasya lopaḥ* । (1.3.9) The *it* marker is deleted.
ci + *liṭ*	*dhātoḥ* । (3.1.91) Suffixes enlisted onwards this rule, till the end of the third chapter, are added after a verbal root and have a *udātta* accent at the beginning. *parokṣe liṭ* । (3.2.115)

[19] This is not a rule but verbal root with its meaning as noted in the *dhātupātha*.

	The affix *liṭ* comes after the verbal root in the sense of the past, before the commencement of the current day and unperceived by the speaker. *laḥ karmaṇi ca bhāve cākarmakebhyaḥ* । (3.4.69) The tense affixes called *la* after *sakarmaka* (transitive) verbal roots are in the sense of *kartṛ* (agent) and *karman* (object); after *akarmaka* (intransitive) verbal roots are in the sense of *kartṛ* (agent) and *bhāva* (action).
ci + l	*upadeśe'janunāsika it* । (1.3.2) In the *upadeśa*, nasalized *ac* (vowel) is termed *it*. *halantyam* । (1.3.3), *tasya lopaḥ* । (1.3.9)
ci + sip	*lasya* । (3.4.77) The suffixes we shall announce hereafter are substituted in the place of *l*. *yuṣmadyupapade samānādhikaraṇe sthāninyapi madhyamaḥ* । (1.4.105) Verbal terminations called *madhyama* (2nd person) are used when *yuṣmad* (you) is understood, and the same is expressed by the verb; Word *yuṣmad* may be used or may not be used in the sentence. *dvyekayordvivacanaikavacane* । (1.4.22) The dual and singular case-suffixes are employed severally in the sense of *dvivacana* (duality) and *ekavacana* (unity).
ci + thal	*parasmaipadānāṃ ṇalatususthalathusaṇalvamāḥ* । (3.4.82) In *liṭlakāra*, *parasmaipada* suffixes are substituted by *ṇal, atus, us, thal, athus, a, ṇal, va* and *ma* respectively.

ci + tha	*halantyam* । (1.3.3), *tasya lopaḥ* । (1.3.9)
ci + tha *ci + iṭ tha*	*acastāsvatthalyaniṭo nityam* । (7.2.61) Suffix *thal* does not also get the augment *iṭ,* after a verbal root, which ends in a vowel and which is always devoid of the *iṭ* augment after the Periphrastic Future affix *tāsi.* *ṛto bhāradvājasya* । (7.2.63) In the opinion of *Bhāradvāja*, the suffix *thal* does not get the augment *iṭ* only after a root which ends in short *ṛ* and after which the Periphrastic Future *tās* is always devoid of the augment *iṭ*.

Without augment *iṭ*	
ci + tha	
ci ci + tha	*liṭi dhātoranabhyāsasya*। (6.1.8) When followed by a suffix of *liṭlakāra*, verbal root which is not already reduplicated, is reduplicated. *pūrvo'bhyāsaḥ*। (6.1.4) When reduplication is done, the first of the two is called *abhyāsa* (reduplicate).
ci ci + tha *ci ki + tha*	*vibhāṣā ceḥ*। (7.3.58) *c* of verbal root *ci* appearing after *abhyāsa* (reduplicate) is optionally substituted by *k*, when followed by *san* (Desiderative) and *liṭ* suffixes.
ci ce + tha *ci ke + tha*	*sārvadhātukārdhadhātukayoḥ*। (7.3.84) When followed by an *sārvadhātuka* or *ārdhadhātuka* suffix, the final *ik* (*i*, *u*, *ṛ*) vowel of a *aṅga* (stem) is substituted by a *guṇa* (*a, e, o*) vowel.
cicetha / ciketha	

With augment *iṭ*	
ci + iṭ tha	*halantyam* ।(1.3.3) *ādyantau ṭakitau* ।(1.1.46) Augment is added at the beginning if it has *ṭ* as *it* and at the end if it has *k* as *it*.
ci + i tha	*tasya lopaḥ*।(1.3.9)
ci ci + i tha	*liṭi dhātoranabhyāsasya* ।(6.1.8)
ci ci + tha *ci ki + tha*	*vibhāṣā ceḥ* ।(7.3.58)
ci ce + i tha *ci ke + i tha*	*sārvadhātukārdhadhātukayoḥ* ।(7.3.84)
ci cay + i tha *ci kay + i tha*	*eco'yavāyāvaḥ* ।(6.1.78) *e, ai, o,* and *au* are respectively substituted by *ay, āy, av* and *āv* when followed by a vowel.
cicayitha / cikayitha	

(b) Aniṭ verbal roots ending in a consonant

(i) *gam – gamlṛ gatau* ।(*bhvādi* – 982)

Steps	**Applicable rules**
gamlṛ	*gamlṛ gatau* ।– to go
gam	*upadeśe'janunāsika It* ।(1.3.2), *tasya lopaḥ* ।(1.3.9)
gam + liṭ	*dhātoḥ* ।(3.1.91), *parokṣe liṭ* ।(3.2.115), *laḥ karmaṇi ca bhāve cākarmakebhyaḥ* ।(3.4.69)
gam + l	*upadeśe'janunāsika it* ।(1.3.2), *halantyam* ।(1.3.3), *tasya lopaḥ* ।(1.3.9)
gam + sip	*lasya* ।(3.4.77), *yuṣmadyupapade samānādhikaraṇe sthāninyapi madhyamaḥ* ।(1.4.105), *dvyekayordvivacanaikavacane* ।(1.4.22)

gam + thal	*parasmaipadānāṃ ṇalatususthalathusaṇalvamāḥ* । (3.4.82)
gam + tha	*halantyam* । (1.3.3), *tasya lopaḥ* । (1.3.9)
gam + tha *gam + iṭ tha*	*upadeśe'tvataḥ* । (7.2.62) Suffix *thal* does not get the augment *iṭ,* after a verbal root which possesses a short *a* as its root-vowel in the *dhātupāṭha* and after which the Periphrastic Future *tās* is always devoid of the augment *iṭ.* *ṛto bhāradvājasya* । (7.2.63)

Without augment *iṭ*	
gam + tha	
gam gam + tha	*liṭi dhātoranabhyāsasya* । (6.1.8), *pūrvo'bhyāsaḥ* । (6.1.4)
ga gam + tha	*halādiḥ śeṣaḥ* । (7.4.60)
ja gam + tha	*kuhoścuḥ* । (7.4.62)
ja gaṃ + tha	*naścāpadāntasya jhali* । (8.3.24) *n* and *m* which do not occur at the end of a *pada* are substituted by *anusvāra* when followed by *jhal*. *jhal* = all consonants, except *h, y, v, r, l,* and nasals.
ja gan + tha	*anusvārasya yayi parasavarṇaḥ* । (8.4.58) *anusvārā* is substituted by a syllable homogenous with the latter one when followed by *yay*. *yay* = all consonants, except *ś, ṣ, s, h*
jagantha	

With augment *iṭ*	
gam + iṭ tha	*halantyam* । (1.3.3), *ādyantau ṭakitau* । (1.1.46)
gam + i tha	*tasya lopaḥ* । (1.3.9)
gam gam + i tha	*liṭi dhātoranabhyāsasya* । (6.1.8), *pūrvo'bhyāsaḥ* । (6.1.4)
ga gam + i tha	*halādiḥ śeṣaḥ* । (7.4.60)
ja gam + i tha	*kuhoścuḥ* । (7.4.62)
jagamitha	

(ii) *bhañj – bhañjo āmardane* । (*rudhādi* - 1453)

Steps	**Applicable rules**
bhañjo	*bhañjo āmardane* । – to reduce to ashes
bhañj	*upadeśe'janunāsika it* । (1.3.2), *tasya lopaḥ*। (1.3.9)
bhañj + liṭ	*dhātoḥ*। (3.1.91), *parokṣe liṭ*। (3.2.115), *laḥ karmaṇi ca bhāve cākarmakebhyaḥ*। (3.4.69)
bhañj + l	*upadeśe'janunāsika it*। (1.3.2), *halantyam*। (1.3.3), *tasya lopaḥ*। (1.3.9)
bhañj + sip	*lasya*। (3.4.77), *yuṣmadyupapade samānādhikaraṇe sthāninyapi madhyamaḥ*। (1.4.105), *dvyekayordvivacanaikavacane*। (1.4.22)
bhañj + thal	*parasmaipadānāṃ ṇalatususthalathusaṇalvamāḥ*। (3.4.82)
bhañj + tha	*Halantyam* । (1.3.3), *tasya lopaḥ* । (1.3.9)
bhañj + tha *bhañj + iṭ tha*	*upadeśe'tvataḥ* । (7.2.62) Suffix *thal* does not get the augment *iṭ,* after a verbal root which possesses a short *a* as its root-vowel in the *dhātupāṭha* and after which the Periphrastic Future *tās* is always devoid of the augment *iṭ.*

	ṛto bhāradvājasya । (7.2.63)

Without augment *iṭ*	
bhañj + tha	
bhañj bhañj + tha	*liṭi dhātoranabhyāsasya* । (6.1.8), *pūrvo'bhyāsaḥ* । (6.1.4)
bha bhañj + tha	*halādiḥ śeṣaḥ* । (7.4.60)
ba bhañj + tha	*abhyāse carca* । (8.4.54)
ba bhañg + tha	*coḥ kuḥ* । (8.2.30)
ba bhañk + tha	*khari ca* । (8.4.55)
ba bhaṃk + tha	*naścāpadāntasya jhali* । (8.3.24)[20]
ba bhaṅk + tha	*anusvārasya yayi parasavarṇaḥ* । (8.4.58)
babhaṅktha	

With augment *iṭ*	
bhañj + iṭ tha	*halantyam*। (1.3.3), *ādyantau ṭakitau* । (1.1.46)
bhañj + i tha	*tasya lopaḥ* । (1.3.9)
bhañj bhañj + i tha	*liṭi dhātoranabhyāsasya*। (6.1.8), *pūrvo'bhyāsaḥ* । (6.1.4)
bha bhañj + i tha	*halādiḥ śeṣaḥ* । (7.4.60)
ba bhañj + i tha	*abhyāse carca* । (8.4.54)
babhañjitha	

==========

[20] *nakārajāvanusvārapañcamau jhali dhātuṣu*। - *anusvāra* and *anunāsikas* are to be considered as modification of *nakāra*, when *jhal* follows.