Formal meta-linguistic devices in Pāṇini

Gérard Huet

Emeritus, Inria Paris

International Webinar on Indian Knowledge Systems

IIT Kharagpur, November 7th 2020

Preamble

It is often difficult to assign inventions to precise historical authors. Newtonian mechanics did not come out of a vacuum. Newton stood on the shoulders of his giant predecessors, and he exchanged a lot with contemporaries such as Leibnitz, Pascal and Descartes. This is specially true of Ancient India, where śāstra was progressively developed by parampara chains, and composed and transmitted as explanations (ṭīkā), glosses (vākya), critical remarks (vārttika), formulaic condensation (sūtra). Vyākaraṇa is no exception, of course. So explaining the contribution of India to linguistics does not stop at Pāṇini or even at Trimuni. And Pāṇini did not appear on the scene unprecedented, he cites many previous grammarians, and the prātiśākhya authors had prepared the ground with phonetics (śikṣā). This said, let us examine some fundamental contributions.

Discretization

Language is a communication medium. A locutor encodes a message with speech, so that it may be decoded by an audience, and used as knowledge, instruction, etc. The discrete character of the message is not obvious. We are talking of language before writing systems were established, where the discrete nature of the signal became explicit from its representation as sequences over a finite set of glyphs. We are talking about speech, which as air vibration is a continuous entity, varying in pitch, intensity, etc. But the point is that it is *articulated* speech, and the various points of articulation discretize the signal as a stream of *phonemes*. The first invention needed to accommodate speech as a formal communication device is discretization. This is accomplished in the varṇamālā.

Varṇamālā

The varṇamālā organizes the *phonemic space* of Sanskrit into 5 vowels in 2 lengths and 3 levels of vocalic deployment (k, guṇa, vṛddhi), 5 points of articulation for consonants (varga), with 2 binary distinctions for mute/voiced and aspirated/unaspirated, 5 nasals, 3 sibilants, a total of 48 varṇas organized in a rigorous algebraic way.

Putting to shame the English alphabet

a ā i ī u ū ṛ ṝ ḷ e ai o au

ṃ ḥ

k kh g gh ṅ

c ch j jh ñ

ṭ ṭh ḍ ḍh ṇ

t th d dh n

p ph b bh m

y r l v ś ṣ s h

Side remark on speech encoding

- There is more to speech than sequences of varṇas, accent must be represented in order to decide ambiguities, like in compounding
- Pāṇini gave precise rules for accentuation
- Unfortunately devanāgarī does not carry this information
- Romanization is wrong, except in the Nancy system of Burnouf and Leupol
- Thus vāg-hariḥ is not representable in IAST, which is not a prefix code
- The only exact transliteration schemes are WX and SLP
- The truth of the matter is in Scharf & Hyman LIES.

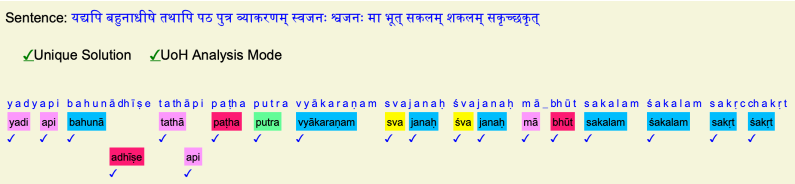
Varṇas discriminate

These discrete atoms of speech are NOT some kind of universal grid on the continuous speech space. Each language has its own. Some parts of the space of humanly producible sounds may be missing, like f and z for Sanskrit. English has 20 vowels, American English only 15. The important point is that phonemes are distinguishable, so that decoding is non-ambiguous even if pronunciation may vary from locutor to locutor. The corresponding notion in Western linguistics is called phoneme. It is attributed to the Prague school of linguistics around 1930. But it is already there fully equipped for Sanskrit in Aṣṭādhyāyī, and it even predates Pāṇini.

Even Raghunātha Śiromaṇi had to learn it

One grammarian anecdote shows the non-obviousness of the notion. In the 15th century, a genius kid called Raghunātha Śiromaṇi went from his native Bengal to Mithilā to study. There, attending the first Vyākaraṇa lesson, on varṇamālā, he protested at a redundant system which distinguishes va and ba, and has a useless distinction of 3 sibilants. This showed that he had grasped correctly the notion, but he was influenced by his native tongue, and had to learn the phonemic system of Sanskrit, that is different from that of Bengali for good reasons. These reasons are exposed in the following subhāṣita, taken from Subhāṣitaratnabhāṇḍāgāra Vaiyākaraṇapraśaṃsā.

Why śa should not be confused with sa



O son, although you studied a lot, you must now learn grammar. One should say svajana (relatives) and not śvajana (canines), sakala (entire) and not śakala (fragment), sakṛt (sudden) and not śakṛt (filth).

This looks like a partial answer to Raghunātha Śiromaṇi's problem, giving three minimal pairs for distinguishing sa and śa.

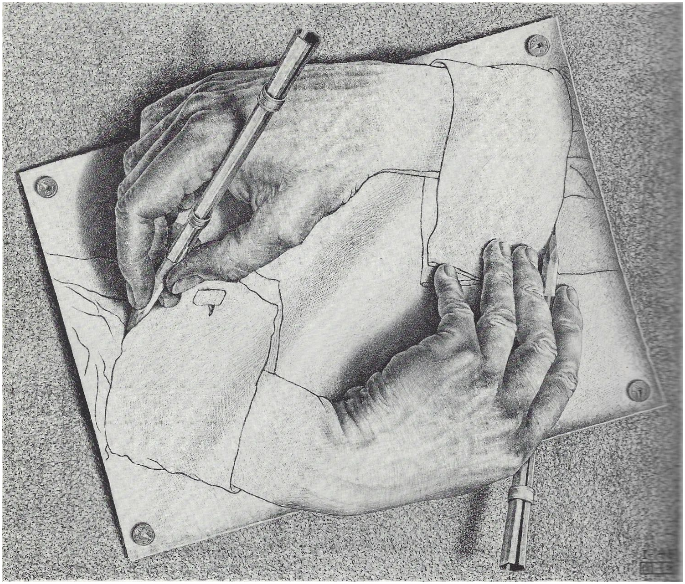
Reflexivity

Studying language means using language to describe language, instead of the ordinary use of language to elaborate thoughts that relate denotations in the real world. So some kind of reflexivity is needed here.

Reflexivity (svayam) is not a notion that should be taken lightly. With reflexivity and negation you get paradoxes. For instance, we attribute to the Cretean philosopher Epimenides the paradox ”I am lying”. Damned if he lies, damned if he does not lie !

This is more serious than it may appear. Variations of this paradox were used by Cantor to show that the reals were not denumerable, and by Gödel to show that arithmetic was incomplete.

Escher's version



Meta-variables for algebra

The problem is how to encode phonetic material (śabda) and operations on it **in its own medium** without confusing the language and the meta-language. For instance, in a context-free grammar, one has to distinguish carefully between the alphabet of the object language and the meta-variables that are instantiated over words. In modern algebra, they are distinguished by fonts, or by formulaic expressions where the position of the metavariables is fixed by the syntax. This builds on the tradition of formal logic, and uses the flexibility of typography in writing. But at the time of Pāṇini, the only way to express formulas was the language itself: sūtras had to be transmitted orally, at the risk of confusing language and meta-language.

Anubandhas and sañjñās

The ingenious solution found by Pāṇini was to use varṇas as meta-variables, but in restricted positions in such a way that there was no possible confusion between phonemes and markers that belong to the meta-language of the grammar. These markers are called anubandhas.
On the other hand, Pāṇini specified in an axiomatic sūtra that Sanskrit words which are not technical terms of the grammar denote their own form:

(I.1.68) svaṃ rūpaṃ śabdasyāśabdasañjñā ||

The sañjñās are the root notions of the meta-description level.

Example: Śivasūtras

The grammar starts by giving another view of the varṇamala:

a i u ṇ |
ṛ ḷ k |
e o ṅ |
ai au c |
ha ya va ra ṭ |
la ṇ |
ña ma ṅa ṇa na m |
jha bha ñ |
gha ḍha dha ṣ |
ja ba ga ḍa da ś |
kha pha cha ṭha tha ca ṭa ta v |
ka pa y |
śa ṣa sa r |
ha l ||

anubandhas are marked in red.

Condensed definitions

The Śivasūtras are used to define abbreviations for the families of phonemes sharing common treatment in the grammar, called *pratyāhāras*. Each *pratyāhāra* is of the form 'XaY', where X is a *varṇa* and Y is an *anubandha*, and it denotes the set of phonemes between X and Y (markers excluded). For instance, nasals are denoted by ñam, vowels are denoted by ac, consonants are denoted by hal.

This is a very compact representation of all subsets of *varṇas* that are needed as characteristic properties for the machine operations.

How to derive the primary nominal kāraka

Let us show quickly how to derive the stem *kāraka* in the sense of actor, i.e. 'agent of acting'. This stem is a primary derivative (*kṛdanta*) obtained by root *kṛ* (to act), with morpheme *aka* affixed to morpheme *kār*, obtained by raising root *kṛ* to its second grade by the *vṛddhi* operation. Here is the (simplified) Paninian derivation.

First, we retrieve the sign for root *kṛ*, by looking up the roots table (*dhātupāṭhaḥ*). At entry *kṛ*, we get: ḍukṛñkaraṇe. We first peel off the morphological parameters ḍu and ñ of the root, record them, and extract the sign components: *kṛ* (the *śabda* phonetic component) and acting its *artha* meaning component (since the locative *karaṇe* means "in the sense of acting"). Thus we start with sign ⟨*kṛ*,acting⟩.

A worked-out example (continued)

Next, since we intend to express the notion of agent, we go to the section of the grammar concerning agent nouns, starting with sūtra (III.1.133): *ṇvultṛcau*, i.e. "both (kṛtpratyayas) *ṇvul* and *tṛc* (are applicable to any root)". By selecting the first component *ṇvul* we are now licensed to affix *pratyaya* *ṇvul* to the current *prakṛti* '*kṛ*', yielding string *kṛṇvul*. Now the string rewriting proceeds. The first operation is denoted by anubandha *ṇ*, which is microcode for the *vṛddhi* operation, rewriting *'kṛ'* into *'kār'*. Next the marker string *vu* invokes an abbreviation mechanism, which expands into śabda *'aka'*, which is thus appended to '*kār*' to yield string *'kāraka'*. The last anubandha *l* indicates that the accent precedes the suffix, yielding accented śabda *'kā́raka'*. And since the sūtra is in the section of agent nouns, the new computed sign is

$$\langle \textit{kā́raka}\ ,\ \text{agent of acting}\rangle.$$

We may then use sup-pratyayas etc to get an inflected pada.

A complex interleaving of śabda and anubandhas

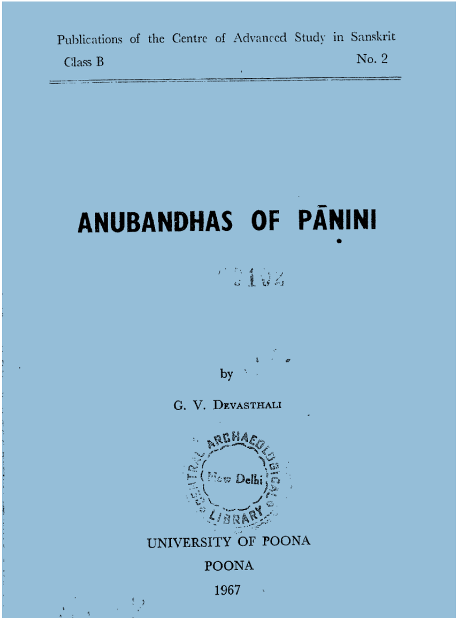
Let us consider again ḍukṛñkaraṇe. We did not discuss ḍu yet. This marker is a very special prefix to certain roots. It is used to license sūtra (III.3.88) to apply first kṛt pratyaya ktri, affixing -tri to the root without effecting guṇa, and followed immediately with sūtra (IV.4.20) to apply taddhita pratyaya map, yielding finally kṛtrima, adjective meaning 'artificial'. Thus we may get saṃskṛtrima, bombastic sanskrit. This may be thought to be a very artificial way to produce the word 'artificial'! But it recognizes correctly a mildly productive morphological scheme yielding e.g. paktrima from root pac (thoroughly cooked).

Now this marker ḍu should not be confused with anubandha ḍ, which followed by some śabda, affixes it after truncation of the trailing consonants of a string, up to and including its last vowel. This sort of linguistic analogue of the K combinator of combinatory logic is used as some kind of magic wand yielding the cow (go) from root gam, by uṇādi suffix ḍo.

Anubandhas have scope

We see that the topic of anubandhas is complex, since their meaning depends of their position in the grammar: these markers have scope, they do not have a global value. Similarly, anubandha ñ following a root in the Dhātupāṭha licences ātmanepada conjugation in the present system, as we saw in ḍukṛñkaraṇe. But in the kṛdanta section of the grammar it denotes an accentuation operation, plus vṛddhi of the final vowel or penultimate 'a' of the root, whereas in the taddhita section it denotes vṛddhi of the first vowel. Not to speak of its role as a pratyāhāra terminal marker.
How can one understand this complexity?

An important resource



Publications of the Centre of Advanced Study in Sanskrit

Class B No. 2

ANUBANDHAS OF PĀṆINI

by

G. V. DEVASTHALI

CENTRAL ARCHAEOLOGICAL LIBRARY
New Delhi

UNIVERSITY OF POONA

POONA

1967

Microcode of an abstract machine

Anubandhas may be considered names for microcode operations of an abstract machine. This “machine” has complex control structures, such as anuvṛtti, recurrence of conditions in rules defined in a certain scope. This is analogous to the invocation of functions by pattern-matching of their argument that you find in functional programming languages. Except that Pāṇini states the general case before listing the exceptions, which bleed out prior more general cases, whereas in programming we state the exceptions before the general case with no need for bleeding. Many scholars remarked on the analogy between Paninian concepts and programming concepts, and on the prescience of information theory that is implicit from many of his encodings, which permitted such a concise formulation.

Post production systems

The basic computing paradigm used in the vidhi style of instructions is rewriting phonemic strings. This algebraic style of computation emerged only in the 20th century Western mathematics with the work of Emil Post (1943) on string rewriting systems. The Paninian vṛtti computations of stems and suffixes may thus be modeled as string rewriting and substitution, with specialized subroutines for doing operations such as vṛddhi, driven by the anubandhas. And the abreviation mechanisms such as pratyahāras permit a succinct expression of families of rewrites. This is especially striking in the formulation of the sandhi rules. Let us look at an example.

(VI.1.77) iko yaṇaci

In order to make sense of this *hocus pocus* formulation, we have first to look up the Śivasūtras for the definition of ik as {i, u, ṛ, ḷ}, of ac as {a, i, u, ṛ, ḷ, e, o, ai, au}, and of yaṇ as {ya, va, ra, la}. Then we have to understand that anubandhas may actually be used as stems in the metalanguage in order to derive padas in certain genders, by reflecting the metalanguage in the object language. Thus we may construct genitive ikaḥ, nominative yaṇ, and locative aci. We may check that the sandhied padas yield indeed the vākya iko yaṇaci. Then we have to understand that Pāṇini encodes a rewrite rule that we would write in modern morpho-phonemic notation $A \rightarrow B/C - D$ as a sūtra $A_6 B_1 C_5 D_7$ where the subscript denotes the case, meaning *From A derive B within left context C and right context D*.

(VI.1.77) continued

Now we have to understand the quantification. A and B correspond pairwise, and we quantify over these pairings, while quantifying over all of D. Thus the rule expresses that {i, u, ṛ, ḷ} rewrite into **respectively** {y, v, r, l} when followed by any vowel. This expands into 4*9=36 rules ! Note that cases are used to build the record or 4-tuple $\langle A, B, C, D\rangle$. This is a striking exemple of a very slick mixture of language and metalanguage.

Is Aṣṭādhyāyī written in Sanskrit?

This is an interesting question. On one hand, even a very fluent Sanskrit speaker will not understand a statement such as iko yaṇaci, unless he has been taught Aṣṭādhyāyī thoroughly. On the other hand, we saw that sūtras are actually combining Sanskrit at two levels. One level is Sanskrit used for technical terms (saṃjñā). These words have their technical meaning in Sanskrit, they mean what they say. At another level, we have words and morphemes of Sanskrit as the object language that are self-referencial, in the sense that they denote their own śabda digitalized as a string of *varṇa*s and *anubandhas*. At this level we also have the *pratyāharas*, that denote sets of *varṇa*s. These symbolic objects act as extra root nominal stems, that take vibhakti in order to satisfy the kārakas of the meta-level. Thus it appears that the answer is **yes**, and that Pāṇini's most striking contribution is that he managed to bootstrap Sanskrit with his machinery.

A fundamental, unique scientific achievement

This bootstrapping of Sanskrit in Pāṇini's grammar is a really striking scientific achievement, at the crossroad of linguistics and informatics. No other human language has been subjected to such a complete self-descriptive formalization, to this date. It is not just very early, it is unique !

Automating sandhi and sandhi viccheda

It is possible to gather together all the vidhi rules concerning external sandhi (at the junction of padas). These rules define what mathematicians call a rational relation over strings of varṇas, implementable by finite transducers. Its inverse is also a rational relation, and thus it is possible to use morpho-phonetics toolkits of computational linguistics to compile Paninian rules into a segmenter that builds the padapāṭha from a continuous enunciation. This is the basis of the Sanskrit Heritage Segmenter, using the Zen toolkit.

Paninian sandhi as Ocaml data

```
; (("p","n","mn"), [ (8,4,45) ]) (*  tri.s.tup+nayati -> tri.s.tumnayati  *)
; (("p","n","bn"), [ (8,2,39) ]) (*  tri.s.tup+nayati -> tri.s.tubnayati *)
; (("p","m","mm"), [ (8,4,45) ])
; (("p","m","bm"), [ (8,2,39) ])
; (("k","h","ggh"), [ (8,4,62) ]) (*  vaak+hari -> vaagghari *)
; ((".t","h",".d.dh"), [ (8,4,62) ]) (*  madhuli.t+hasati -> madhuli.d.dhasati *)
; (("t","h","ddh"), [ (8,4,62) ]) (*  tat+hita -> taddhita *)
; (("p","h","bbh"), [ (8,4,62) ]) (*  tri.s.tup+hasati -> tri.s.tubbhasati  *)
; (("t","z","cch"), [ (8,4,40); (8,4,63) ]) (*  tat+ziva.h -> tacchiva.h *)
; (("t","z","cz"), [ (8,4,40) ]) (*  tat+ziva.h -> tacziva.h *)
; (("t","c","cc"), [ (8,4,40) ]) (*  sat+carita.h -> saccarita.h *)
; (("t","ch","cch"), [ (8,4,40) ])
; (("t","j","jj"), [ (8,4,40); (8,4,53) ]) (*  sat+jana.h -> sajjana.h *)
; (("t","jh","jjh"), [ (8,4,40); (8,4,53) ])
; (("n","j","~nj"), [ (8,4,40) ]) (*  sat+jana.h -> sajjana.h *)
; (("n","jh","~njh"), [ (8,4,40) ])
; ((".t","z",".tch"), [ (8,4,63) ]) (*  svali.t+zete -> svali.tchete *)
```

Pawan, author of the translation

Thank you for your attention