On the Construction of *Śivasūtra*-Alphabets

Wiebke Petersen

Institute of Language and Information
University of Düsseldorf, Germany
petersew@uni-duesseldorf.de

IIIT Hyderabad, 20th January 2009

अइउण्। ऋऌक्। एओङ्। ऐऔच्। हयवरट्।
लण्। ञमङणनम्। झभञ्। घढधष्। जबगडदश्।
खफछठथचटतव्। कपय्। शषसर्। हल्।

Phonological Rules

modern notation

A is replaced by B if preceded by C and succeeded by D.

$$A \rightarrow B/_{C\_\_D}$$

example: final devoicing

$$\begin{bmatrix} + & \text{consonantal} \\ - & \text{nasal} \\ + & \text{voiced} \end{bmatrix} \rightarrow \begin{bmatrix} + & \text{consonantal} \\ - & \text{nasal} \\ - & \text{voiced} \end{bmatrix} /\_\_\sharp$$

Phonological Rules

modern notation

A is replaced by B if preceded by C and succeeded by D.

$$A \rightarrow B/_{C\_\_D}$$

example: final devoicing

$$\begin{bmatrix} + & \text{consonantal} \\ - & \text{nasal} \\ + & \text{voiced} \end{bmatrix} \rightarrow \begin{bmatrix} + & \text{consonantal} \\ - & \text{nasal} \\ - & \text{voiced} \end{bmatrix} /\_\_\sharp$$

Phonological Rules

modern notation

A is replaced by B if preceded by C and succeeded by D.

$$A \rightarrow B/_{C\_D}$$

Pāṇini's linear Coding

$$A + \text{genitive}, B + \text{nominative}, C + \text{ablative}, D + \text{locative}.$$

example

- *sūtra* 6.1.77: *iko yaṇaci* (इको यणचि)
- analysis: $[\text{ik}]_{\text{gen}}[\text{yaṇ}]_{\text{nom}}[\text{ac}]_{\text{loc}}$
- modern notation: [iK] → [yṆ]/__[aC]

Phonological Rules

modern notation

A is replaced by B if preceded by C and succeeded by D.

$$A \rightarrow B/_{C\_\_D}$$

Pāṇini's linear Coding

$$A + \text{genitive}, B + \text{nominative}, C + \text{ablative}, D + \text{locative}.$$

example

- *sūtra* 6.1.77: *iko yaṇaci* (इको यणचि)
- analysis: $[\text{ik}]_{\text{gen}}[\text{yaṇ}]_{\text{nom}}[\text{ac}]_{\text{loc}}$
- modern notation: [iK] → [yṆ]/__ [aC]

Pāṇini faced the problem of giving a linear representation of the nonlinear system of sound classes.

A similar problem occurs in . . .

Libraries

Warehouses and stores

Pāṇini's solution: *Śivasūtras*

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ
6.					l	Ṇ
7.	ñ	m	ṅ	ṇ	n	M
8.	jh	bh				Ñ
9.			gh	ḍh	dh	Ṣ
10.	j	b	g	ḍ	d	Ś
11.	kh	ph	ch	ṭh	th	
			c	ṭ	t	V
12.	k	p				Y
13.		ś	ṣ	s		R
14.	h					L

अइउण्। ऋऌक्।
a·i·uṇ| ṛ·ḷk|

एओङ्। ऐऔच्।
e·oṅ| ai·auc|

हयवरट्। लण्।
hayavaraṭ| laṇ|

ञमङणनम्। झभञ्।
ñamaṅaṇanam| jhabhañ|

घढधष्। जबगडदश्।
ghaḍhadhaṣ| jabagaḍadaś|

खफछठथचटतव्।
khaphachaṭhathacaṭatav|

कपय्। शषसर्। हल्।
kapay| śaṣasar| hal|

Pāṇini's solution: *Śivasūtras*

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ
6.					l	Ṇ
7.	ñ	m	ṅ	ṇ	n	M
8.	jh	bh				Ñ
9.			gh	ḍh	dh	Ṣ
10.	j	b	g	ḍ	d	Ś
11.	kh	ph	ch	ṭh	th	
			c	ṭ	t	V
12.	k	p				Y
13.		ś	ṣ	s		R
14.	h					L

अइउण्। ऋऌक्।
a·i·uṇ | *ṛ·ḷk* |
एओङ्। ऐऔच्।
e·oṅ | *ai·auc* |
हयवरट्। लण्।
hayavaraṭ | *laṇ* |
ञमङणनम्। झभञ्।
ñamaṅaṇanam | *jhabhañ* |
घढधष्। जबगडदश्।
ghaḍhadhaṣ | *jabagaḍadaś* |
खफछठथचटतव्।
khaphachaṭhathacaṭatav |
कपय्। शषसर्। हल्।
kapay | *śaṣasar* | *hal* |

Pāṇini's solution: *Śivasūtras*

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ
6.					l	Ṇ
7.	ñ	m	ṅ	ṇ	n	M
8.	jh	bh				Ñ
9.			gh	ḍh	dh	Ṣ
10.	j	b	g	ḍ	d	Ś
11.	kh	ph	ch	ṭh	th	
			c	ṭ	t	V
12.	k	p				Y
13.		ś	ṣ	s		R
14.	h					L

anubandha

अइउण्। ऋऌक्।
a·i·uṇ| *ṛ·ḷk*|
एओङ्। ऐऔच्।
e·oṅ| *ai·auc*|
हयवरट्। लण्।
hayavaraṭ| *laṇ*|
ञमङणनम्। झभञ्।
ñamaṅaṇanam| *jhabhañ*|
घढधष्। जबगडदश्।
ghaḍhadhaṣ| *jabagaḍadaś*|
खफछठथचटतव्।
khaphachaṭhathacaṭatav|
कपय्। शषसर्। हल्।
kapay| *śaṣasar*| *hal*|

Pratyāhāras

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ

Pratyāhāras

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ

iK

Pratyāhāras

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ

iK= ⟨i, u, ṛ, ḷ⟩

Analysis of iko yaṇaci: [iK] → [yṆ]/__[aC]

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ
6.					l	Ṇ

- [iK] → [yṆ]/__[aC]
- ⟨i, u, ṛ, ḷ⟩ → ⟨y, v, r, l⟩/__⟨a, i, u, ṛ, ḷ, e, o, ai, au⟩

Analysis of iko yaṇaci: [iK] → [yṆ]/_ [aC]

1.	a	i	u			Ṇ
2.				ṛ	ḷ	K
3.		e	o			Ṅ
4.		ai	au			C
5.	h	y	v	r		Ṭ
6.					l	Ṇ

- [iK] → [yṆ]/_ [aC]
- ⟨i, u, ṛ, ḷ⟩ → ⟨y, v, r, l⟩/_ ⟨a, i, u, ṛ, ḷ, e, o, ai, au⟩

General problem of S-sortability

Given a set of classes, order the elements of the classes (without duplications) in a linear order (in a list) such that each single class forms a continuous interval with respect to that order.

- The target orders are called S-orders
- A set of classes is S-sortable if it has an S-order

General problem of *Śivasūtra*-alphabets (S-alphabets)

Given a set of classes, find an S-order of the elements of the classes. Interrupt this list by markers (*anubandhas*) such that each single class can be denoted by a sound-marker-pair (*pratyāhāra*).

Note that every S-order becomes a *Śivasūtra*-alphabet (S-alphabet) by adding a marker (*anubandha*) behind each element.

Given the set of classes $\{\{a, b\}, \{a, b, c\}, \{a, b, c, d\}\}$, the order $a\ b\ c\ d$ is one of its S-orders and $a\ M_1\ b\ M_2\ c\ M_3\ d\ M_4$ is one of its S-alphabets.

General problem of *Śivasūtra*-alphabets (S-alphabets)

Given a set of classes, find an S-order of the elements of the classes. Interrupt this list by markers (*anubandhas*) such that each single class can be denoted by a sound-marker-pair (*pratyāhāra*).

Note that every S-order becomes a *Śivasūtra*-alphabet (S-alphabet) by adding a marker (*anubandha*) behind each element.

Given the set of classes $\{\{a, b\}, \{a, b, c\}, \{a, b, c, d\}\}$, the order $a\ b\ c\ d$ is one of its S-orders and $a\ M_1\ b\ M_2\ c\ M_3\ d\ M_4$ is one of its S-alphabets.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\, b\, c\, g\, h\, f\, i\, d\, e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\, b\, c\, g\, h\, f\, i\, d\, e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\ b\ c\ g\ h\ f\ i\ d\ e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\ b\ c\ g\ h\ f\ i\ d\ e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d,e\},\{a,b\},\{b,c,d,f,g,h,i\},\{f,i\},\{c,d,e,f,g,h,i\},\{g,h\}\}$ is S-sortable;
one of its S-orders is
$a\,b\,c\,g\,h\,f\,i\,d\,e$

non-S-sortable example

The set of classes:
$\{\{a,b\},\{b,c\},\{a,c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d,e\},\{a,b\},\{b,c,d\},\{b,c,d,f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\ b\ c\ g\ h\ f\ i\ d\ e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.

Some more Examples

S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$ is S-sortable;
one of its S-orders is
$a\ b\ c\ g\ h\ f\ i\ d\ e$

non-S-sortable example

The set of classes:
$\{\{a, b\}, \{b, c\}, \{a, c\}\}$ is not S-sortable.

non-S-sortable example

The set of classes:
$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$ is not S-sortable.
$a\ b\ c\ d\ e$ or $e\ d\ c\ b\ a$

Some more Examples

S-sortable example

The set of classes:
$\{\{d,e\},\{a,b\},\{b,c,d,f,g,h,i\},\{f,i\},\{c,d,e,f,g,h,i\},\{g,h\}\}$ is S-sortable;
one of its S-orders is
$a\,b\,c\,g\,h\,f\,i\,d\,e$

non-S-sortable example

The set of classes:
$\{\{a,b\},\{b,c\},\{a,c\}\}$ is not S-sortable.

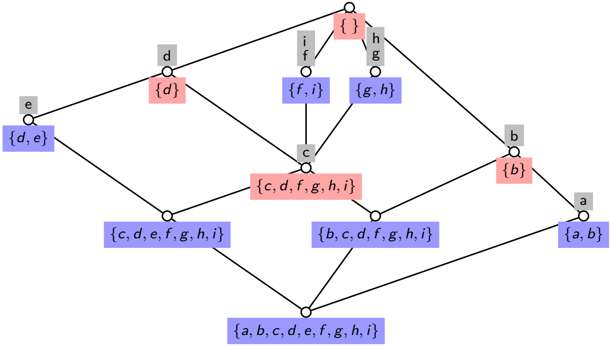
non-S-sortable example

The set of classes:
$\{\{d,e\},\{a,b\},\{b,c,d\},\{b,c,d,f\}\}$ is not S-sortable.
$a\,b\,c\,d\,e$ or $e\,d\,c\,b\,a$

Visualize relations

set of classes $(\mathcal{A}, \Phi)$: $\mathcal{A} = \{a, b, c, d, e, f, g, h, i\}$

$$\Phi = \{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$$



concept lattice of $(\mathcal{A}, \Phi)$

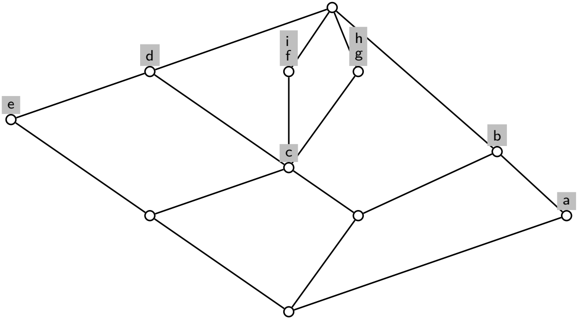
	a	b	c	d	e	f	g	h	i
$\{d, e\}$				×	×				
$\{b, c, d, f, g, h, i\}$		×	×	×	×	×	×	×	×
$\{a, b\}$	×	×							
$\{f, i\}$						×			×
$\{c, d, e, f, g, h, i\}$			×	×	×	×	×	×	×
$\{g, h\}$							×	×	

formal context of $(\mathcal{A}, \Phi)$

Visualize relations

set of classes $(\mathcal{A}, \Phi)$: $\mathcal{A} = \{a, b, c, d, e, f, g, h, i\}$

$$\Phi = \{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$$



concept lattice of $(\mathcal{A}, \Phi)$

	a	b	c	d	e	f	g	h	i
$\{d, e\}$				×	×				
$\{b, c, d, f, g, h, i\}$		×	×	×	×	×	×	×	×
$\{a, b\}$	×	×							
$\{f, i\}$						×			×
$\{c, d, e, f, g, h, i\}$			×	×	×	×	×	×	×
$\{g, h\}$							×	×	

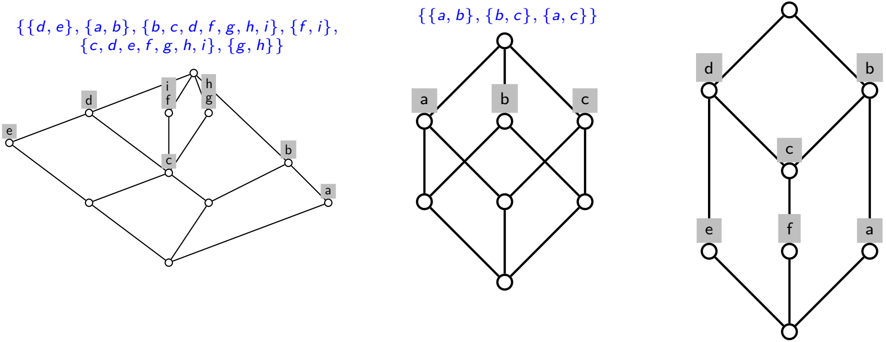
formal context of $(\mathcal{A}, \Phi)$

Visualize relations

$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\}, \{c, d, e, f, g, h, i\}, \{g, h\}\}$

$\{\{a, b\}, \{b, c\}, \{a, c\}\}$

$\{\{d, e\}, \{a, b\}, \{b, c, d\}, \{b, c, d, f\}\}$

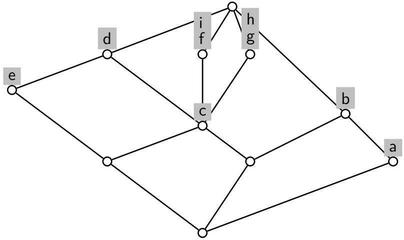


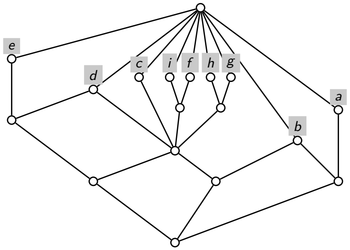
Main theorem of S-sortability

A set of classes is S-sortable without duplications if one of the following equivalent statements is true:

1. Its concept lattice is Hasse-planar and for any element *a* there is a node labeled *a* in the S-graph.
2. The concept lattice of the enlarged set of classes is Hasse-planar.
3. The Ferrers-graph of the enlarged set of classes is bipartite.

Example: S-sortable



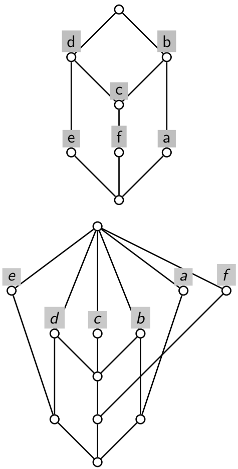


Main theorem of S-sortability

A set of classes is S-sortable without duplications if one of the following equivalent statements is true:

1. Its concept lattice is Hasse-planar and for any element *a* there is a node labeled *a* in the S-graph.
2. The concept lattice of the enlarged set of classes is Hasse-planar.
3. The Ferrers-graph of the enlarged set of classes is bipartite.

Example: not S-sortable



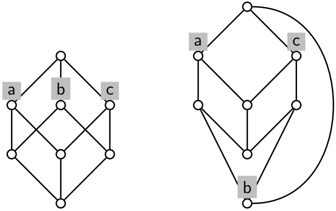
2nd condition: terminology

2nd condition

A set of classes $(\mathcal{A}, \Phi)$ is S-sortable without duplications if and only if the concept lattice of the enlarged set of classes $(\mathcal{A}, \tilde{\Phi})$ is Hasse-planar.

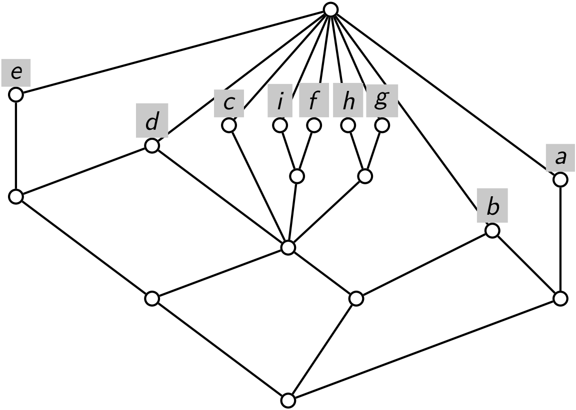
Enlarging a set of classes means adding all singleton sets:
$\tilde{\Phi} = \Phi \cup \{\{a\} \mid a \in \mathcal{A}\}$

Hasse-planarity: $\{\{a, b\}, \{a, c\}, \{b, c\}\}$



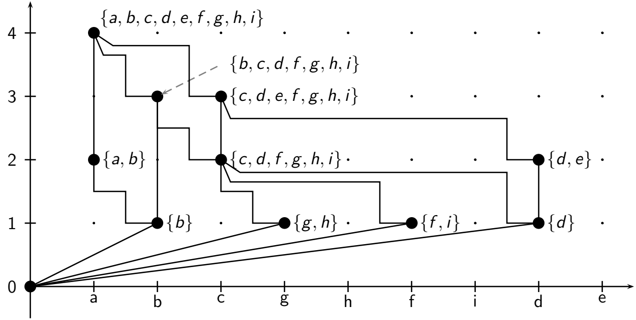
planar, but not Hasse-planar

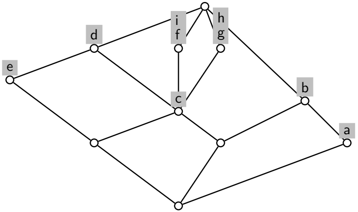
2nd condition: Hasse-planar ⇒ S-sortable

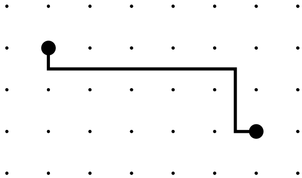


$\{\{d, e\}, \{a, b\}, \{b, c, d, f, g, h, i\}, \{f, i\},$
$\{c, d, e, f, g, h, i\}, \{g, h\}\}$

2nd condition: S-sortable ⇒ Hasse-planar







2nd condition: evaluation

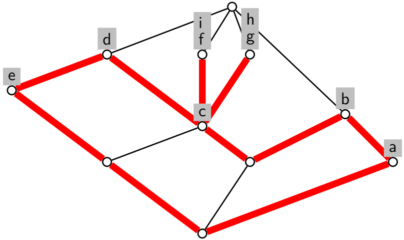
- It is of no help in the construction of S-alphabets with minimal number of markers.
- The planarity of a graph is difficult to check.

Main theorem of S-sortability

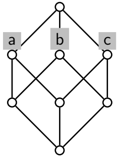
A set of classes is S-sortable without duplications if one of the following equivalent statements is true:

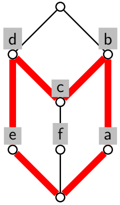
1. Its concept lattice is Hasse-planar and for any element *a* there is a node labeled *a* in the S-graph.
2. The concept lattice of the enlarged set of classes is Hasse-planar.
3. The Ferrers-graph of the enlarged set of classes is bipartite.

Example: S-sortable



Examples: not S-sortable





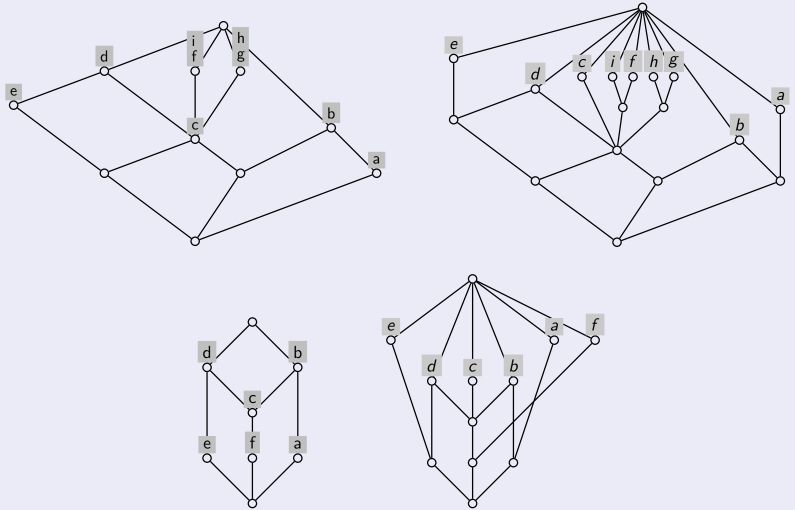
1st condition: proof

2nd condition → 1st condition

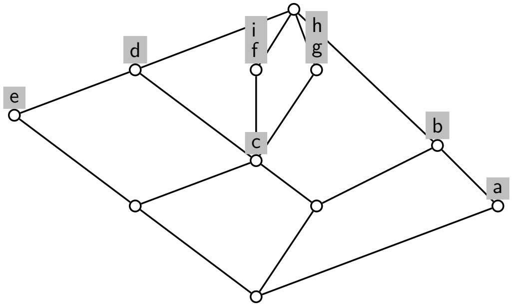
Each S-order of the enlarged set of classes $(\mathcal{A}, \tilde{\Phi})$ is trivially an S-order of the original set of classes $(\mathcal{A}, \Phi)$.

1st condition: proof

1st condition → 2nd condition



S-alphabets with a minimal number of markers

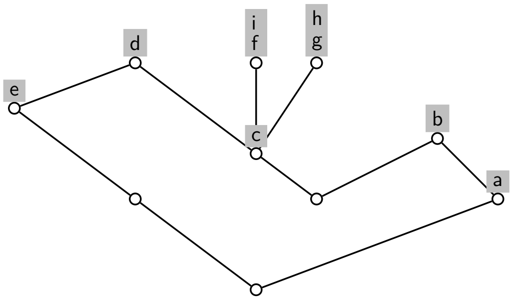


procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a labeled node is reached, add the labels in arbitrary order to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers

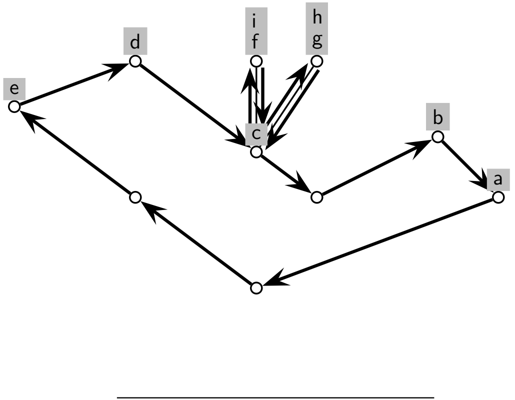


procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a labeled node is reached, add the labels in arbitrary order to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers

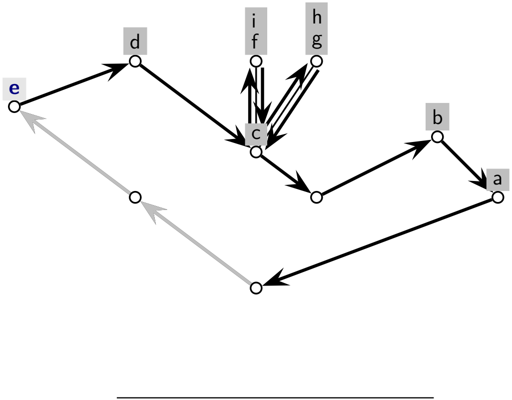


procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers

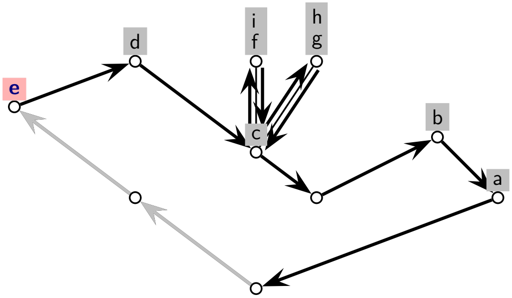


procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



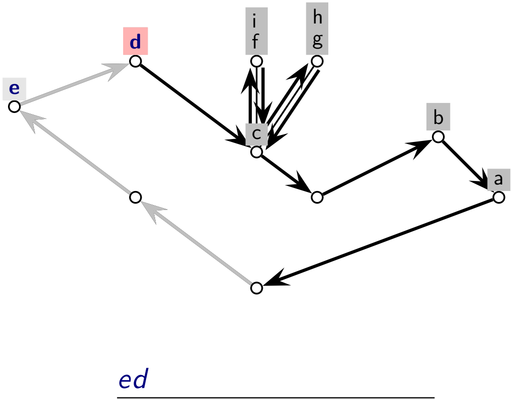
e

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers

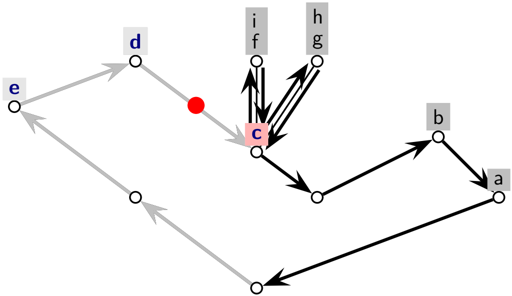


procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



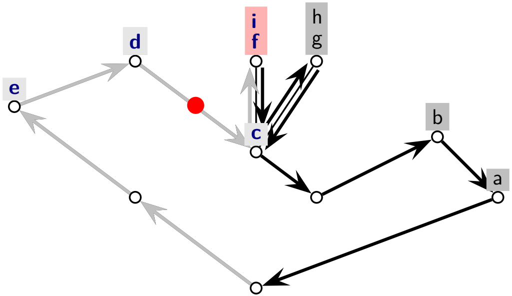
$ed\,M_1\,c$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



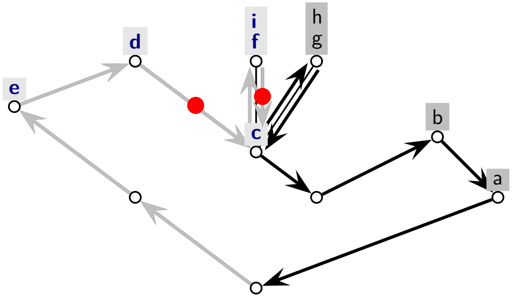
ed M_1 *cfi*

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



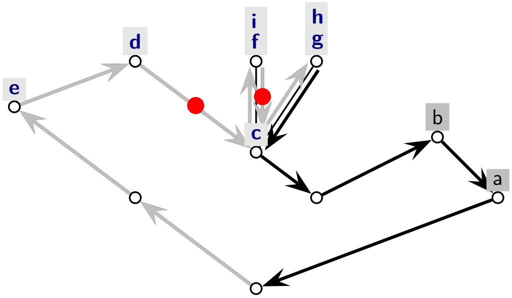
$ed\,M_1\,cfi\,M_2$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



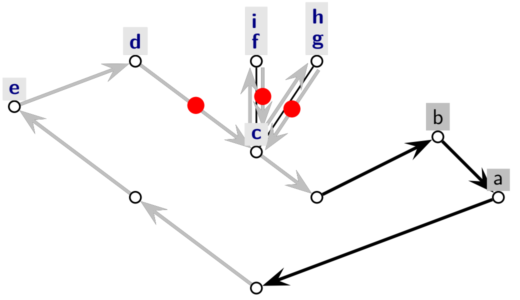
$ed\,M_1\,cfi\,M_2\,gh$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



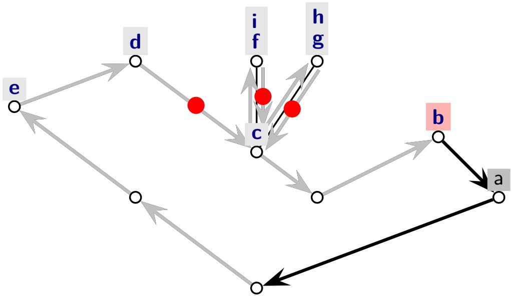
$\color{navy}{ed}\color{red}{M_1}\color{navy}{cfi}\color{red}{M_2}\color{navy}{gh}\color{red}{M_3}$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



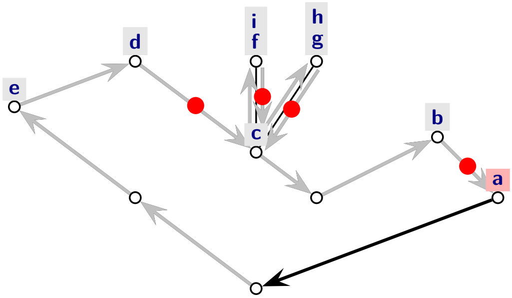
$ed\,M_1\,c\,fi\,M_2\,gh\,M_3\,b$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



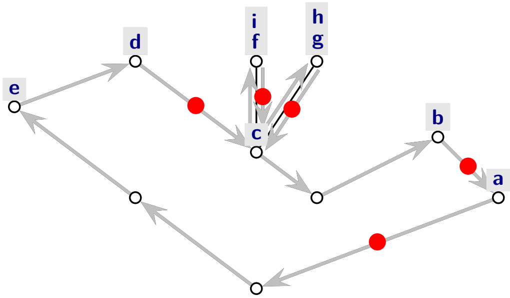
$ed\,M_1\,cfi\,M_2\,gh\,M_3\,b\,M_4\,a$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

S-alphabets with a minimal number of markers



$ed\,M_1\,cfi\,M_2\,gh\,M_3\,b\,M_4\,a\,M_5$

procedure

Start with the empty sequence and choose a walk through the S-graph:

- While moving upwards do nothing.
- While moving downwards along an edge add a new marker to the sequence unless its last element is already a marker.
- If a sound is reached, add the sound to the sequence, unless it has been added before.

1st condition: evaluation

+ Allows the construction of S-alphabets with minimal number of markers.
− The planarity of a graph is difficult to check.

Main theorem of S-sortability

A set of classes is S-sortable without duplications if one of the following equivalent statements is true:

1. Its concept lattice is Hasse-planar and for any element *a* there is a node labeled *a* in the S-graph.
2. The concept lattice of the enlarged set of classes is Hasse-planar.
3. The Ferrers-graph of the enlarged set of classes is bipartite.

- The Ferrers-graph can be computed directly from the formal context.
- Its bipartity can be checked algorithmically.

3rd condition: terminology & proof

Theorem (Zschalig 2007)

The concept lattice of a formal context is Hasse-planar if and only if its Ferrers-graph is bipartite.

	a	b	c	d	e	f
0				×	×	
1		×	×	×		
2	[×]	×				
3		×	[×]			×

3rd condition: terminology & proof

Theorem (Zschalig 2007)

The concept lattice of a formal context is Hasse-planar if and only if its Ferrers-graph is bipartite.

	a	b	c	d	e	f
0	●	●	●	×	×	●
1	●	×	×	×	●	●
2	[×]	×	●	●	●	●
3	●	×	[×]	●	●	×

3rd condition: terminology & proof

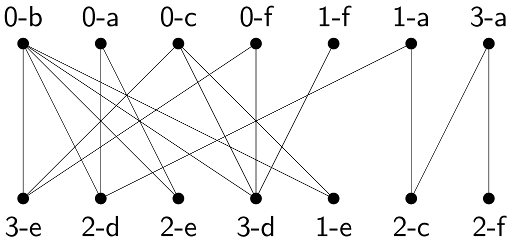
Theorem (Zschalig 2007)

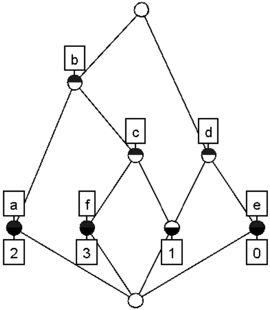
The concept lattice of a formal context is Hasse-planar if and only if its Ferrers-graph is bipartite.

	a	b	c	d	e	f
0	●	●	●	×	×	●
1	●	×	×	×	●	●
2	×	×	●	●	●	●
3	●	×	×	●	●	×

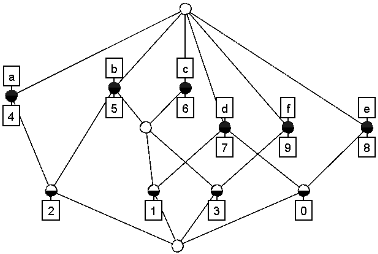
3rd condition: example

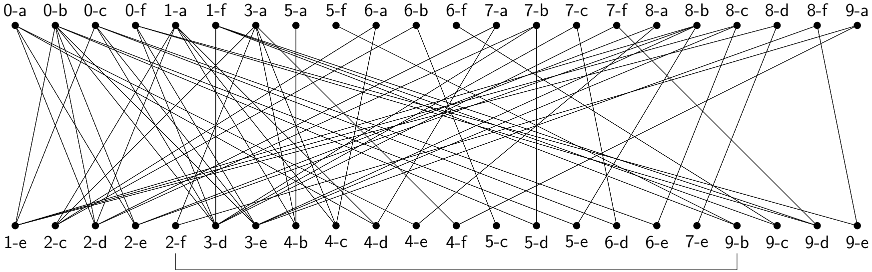
	a	b	c	d	e	f
0				×	×	
1		×	×	×		
2	×	×				
3		×	×			×





3rd condition: example





3rd condition: evaluation

- – It is of no help in the construction of S-alphabets with minimal number of markers.
- + It can be checked easily by an algorithm.

Getting back to Pāṇini's problem

a·i·uṇ| ṛ·ḷk| e·oṅ| ai·auc| hayavaraṭ|
laṇ| ñamaṅaṇanam| jhabhañ| ghaḍhadhaṣ| jabagaḍadaś|
khaphachaṭhathacaṭatav| kapay| śaṣasar| hal|

Q: Are the *Śivasūtras* minimal (with respect to length)?

What does minimal mean?

$a{\cdot}i{\cdot}u\d{n}|$ $\d{r}{\cdot}\d{l}k|$ $e{\cdot}o\dot{n}|$ $ai{\cdot}auc|$ $hayavara\d{t}|$
$la\d{n}|$ $\tilde{n}ama\dot{n}a\d{n}anam|$ $jhabha\tilde{n}|$ $gha\d{d}hadha\d{s}|$ $jabaga\d{d}ada\acute{s}|$
$khaphacha\d{t}hathaca\d{t}atav|$ $kapay|$ $\acute{s}a\d{s}asar|$ $hal|$

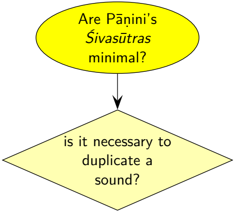
The *Śivasūtras* are minimal if it is **im**possible to rearrange the Sanskrit sounds in a new list with *anubandhas* such that

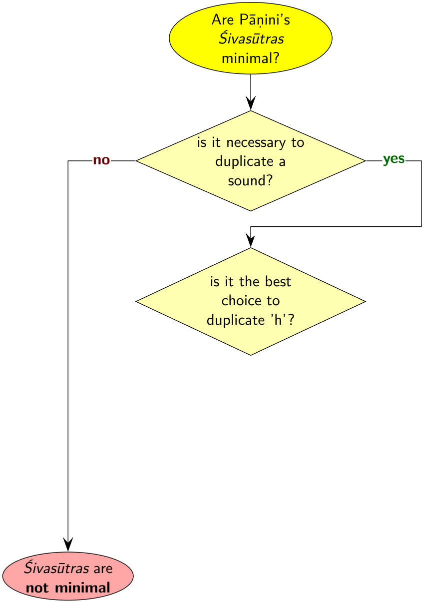
1. each *pratyāhāra* forms an interval ending before an *anubandha*,
2. no sound occurs twice

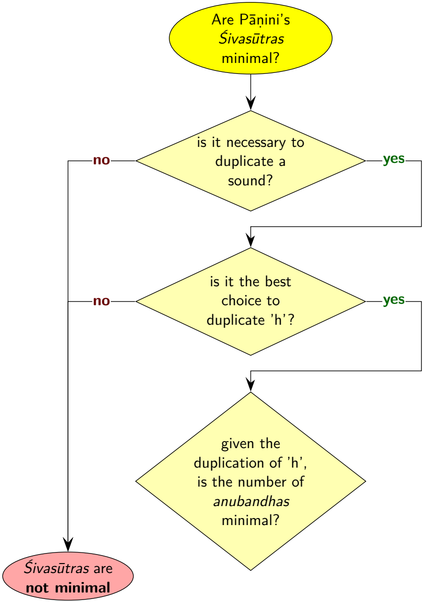
or one sound occurs twice but less *anubandhas* are needed.

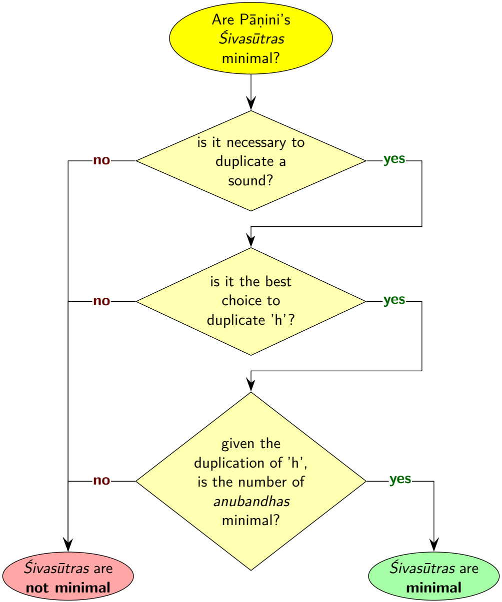
⇒ duplicating a sound is worse than adding *anubandhas*

Are Pāṇini's *Śivasūtras* minimal?


Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?


Are Pāṇini's
Śivasūtras
minimal?
is it necessary to
duplicate a
sound?
no
yes
is it the best
choice to
duplicate 'h'?
Śivasūtras are
not minimal

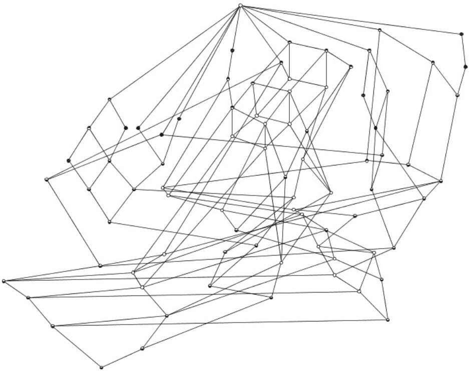

Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?
no
yes
is it the best choice to duplicate 'h'?
no
yes
given the duplication of 'h', is the number of *anubandhas* minimal?
Śivasūtras are **not minimal**


Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?
no
yes
is it the best choice to duplicate 'h'?
no
yes
given the duplication of 'h', is the number of *anubandhas* minimal?
no
yes
Śivasūtras are **not minimal**
Śivasūtras are **minimal**

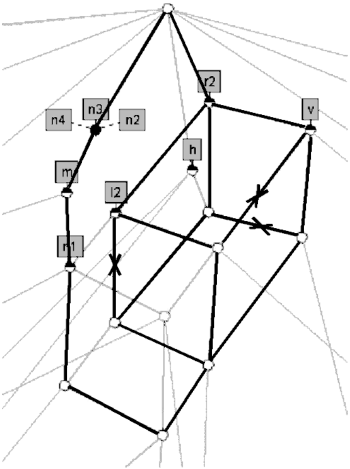
Is it necessary to duplicate a sound?

Main theorem on S-sortability (part 1a)

If a set of classes is S-sortable, then its concept lattice is Hasse-planar.

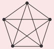


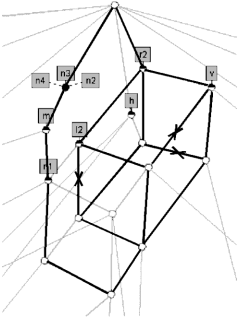
concept lattice of Pāṇini's *pratyāhāras*



Is it necessary to duplicate a sound?

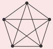
Criterion of Kuratowski

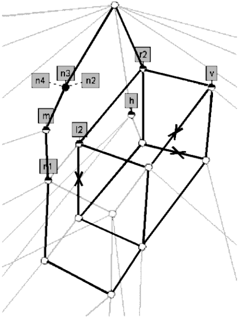
A graph which has the graph  as a minor is not planar.

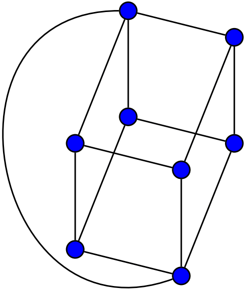


Is it necessary to duplicate a sound?

Criterion of Kuratowski

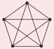
A graph which has the graph  as a minor is not planar.

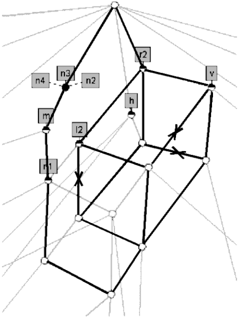


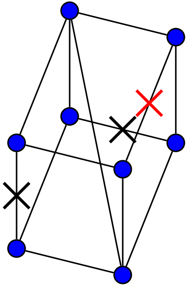


Is it necessary to duplicate a sound?

Criterion of Kuratowski

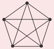
A graph which has the graph  as a minor is not planar.

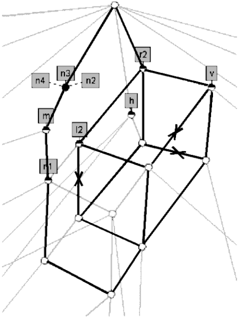


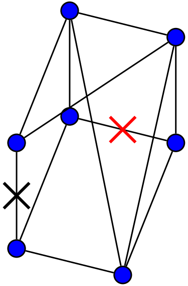


Is it necessary to duplicate a sound?

Criterion of Kuratowski

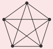
A graph which has the graph  as a minor is not planar.

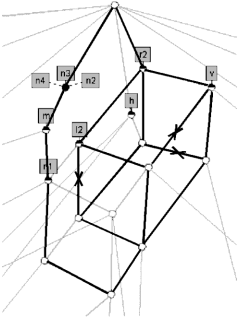


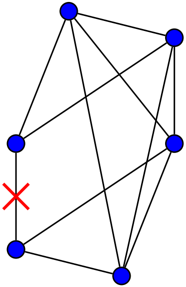


Is it necessary to duplicate a sound?

Criterion of Kuratowski

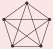
A graph which has the graph  as a minor is not planar.

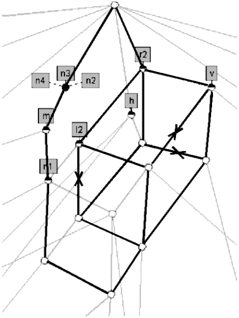


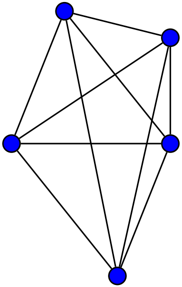


Is it necessary to duplicate a sound?

Criterion of Kuratowski

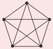
A graph which has the graph  as a minor is not planar.

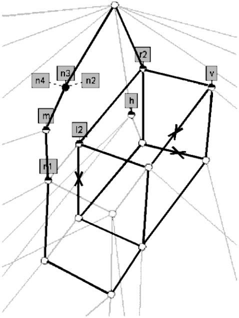


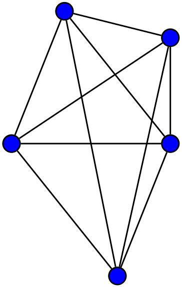


Is it necessary to duplicate a sound?

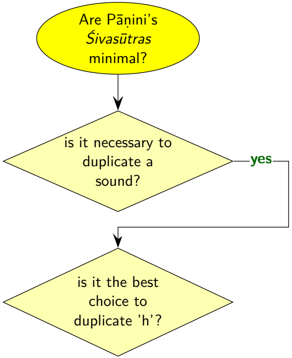
Criterion of Kuratowski

A graph which has the graph  as a minor is not planar.



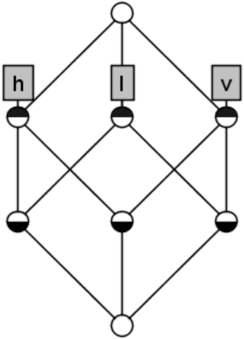


There is no S-alphabet for the set of classes given by Pāṇini's *pratyāhāras* without duplicated elements!

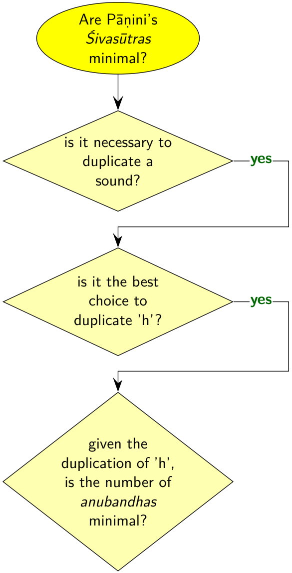

Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?
yes
is it the best choice to duplicate 'h'?

h and the independent triples

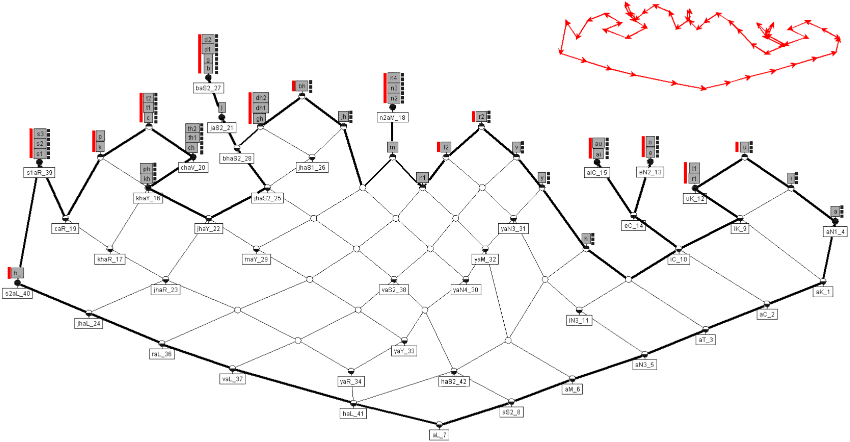
	h	l	v
$\{h, l\}$	×	×	
$\{h, v\}$	×		×
$\{v, l\}$		×	×



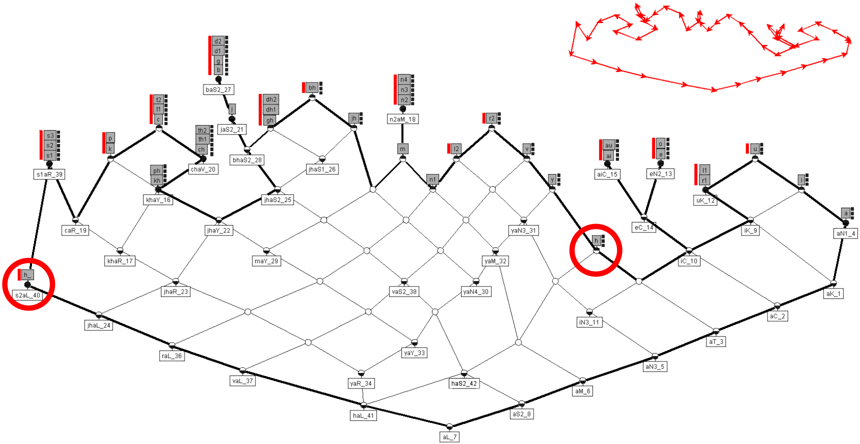
Altogether there exists 249 independent triples.
h is included in all of them.


Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?
yes
is it the best choice to duplicate 'h'?
yes
given the duplication of 'h', is the number of *anubandhas* minimal?

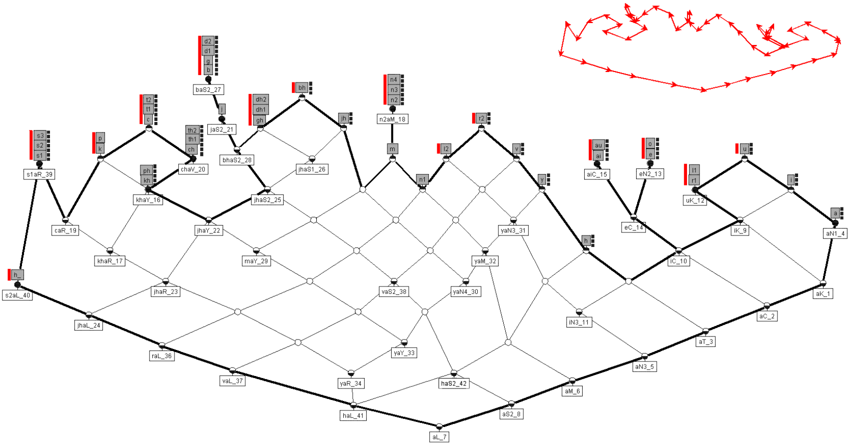
Concept lattice of Pāṇini's *pratyāhāras* with duplicated *h*



Concept lattice of Pāṇini's *pratyāhāras* with duplicated *h*



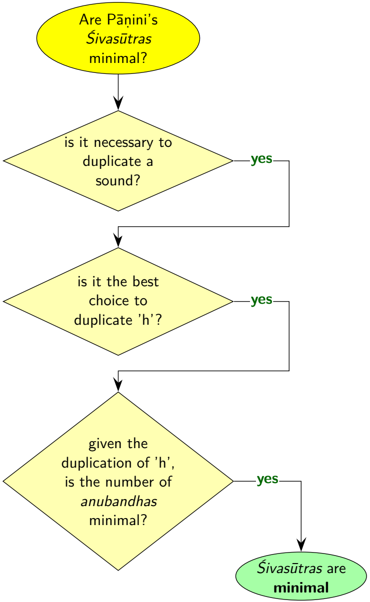
Concept lattice of Pāṇini's *pratyāhāras* with duplicated *h*



With the *Śivasūtras* Pāṇini has chosen one out of nearly 12 million minimal S-alphabets!

$$\langle a, i, u, M_1, \{ṛ, ḷ\}_1, M_2, \{\langle\{e, o\}_2, M_3\rangle, \langle\{ai, au\}_3, M_4\rangle\}_4,$$
$$h, y, v, r, M_5, l, M_6, ñ, m, \{ṅ, ṇ, n,\}_5, M_7, jh, bh, M_8,$$
$$\{gh, ḍh, dh\}_6, M_9, j, \{b, g, ḍ, d\}_7, M_{10}, \{kh, ph\}_8, \{ch, ṭh, th\}_9,$$
$$\{c, ṭ, t\}_{10}, M_{11}, \{k, p\}_{11}, M_{12}, \{ś, ṣ, s\}_{12}, M_{13}, h, M_{14}\rangle$$

$$\underset{\{\}_1}{2!} \times \underset{\{\}_2}{2!} \times \underset{\{\}_3}{2!} \times \underset{\{\}_4}{2!} \times \underset{\{\}_5}{3!} \times \underset{\{\}_6}{3!} \times \underset{\{\}_7}{4!} \times \underset{\{\}_8}{2!} \times \underset{\{\}_9}{3!} \times \underset{\{\}_{10}}{3!} \times \underset{\{\}_{11}}{2!} \times \underset{\{\}_{12}}{3!}$$
$$= 2 \times 2 \times 2 \times 2 \times 6 \times 6 \times 24 \times 2 \times 6 \times 6 \times 2 \times 6 = 11\,943\,936$$


Are Pāṇini's *Śivasūtras* minimal?
is it necessary to duplicate a sound?
yes
is it the best choice to duplicate 'h'?
yes
given the duplication of 'h', is the number of *anubandhas* minimal?
yes
Śivasūtras are **minimal**

Open problems

The story is much more intricate

- We have neither shown that Pāṇini's technique for the representation of sound classes is optimal
- nor that he has used his technique in an optimal way.
  - not all sound classes are denoted by *pratyāhāras*
  - rules overgeneralize
  - *sūtra* 1.3.10: *yathāsaṃkhyamanudeśaḥ samānām*

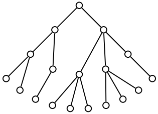
Open problems

The story is much more intricate

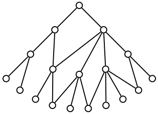
- We have neither shown that Pāṇini's technique for the representation of sound classes is optimal
- nor that he has used his technique in an optimal way.
    - not all sound classes are denoted by *pratyāhāras*
    - rules overgeneralize
    - *sūtra* 1.3.10: *yathāsaṃkhyamanudeśaḥ samānām*

Transfer

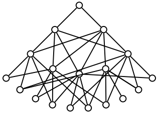
- For physical objects ‚duplicating‘ means ‚adding copies‘
- Adding copies is annoying but often not impossible
- Ordering objects in an S-order may
  - improve user-friendliness
  - save time
  - save space
  - simplify visual representations of classifications



tree



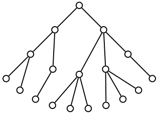
S-sortable



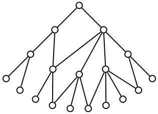
general hierarchy

Transfer

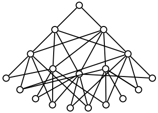
- For physical objects ‚duplicating' means ‚adding copies'
- Adding copies is annoying but often not impossible
- Ordering objects in an S-order may
  - improve user-friendliness
  - save time
  - save space
  - simplify visual representations of classifications



tree



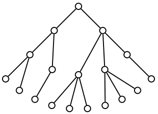
S-sortable



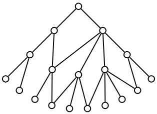
general hierarchy

Transfer

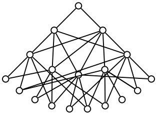
- For physical objects ‚duplicating' means ‚adding copies'
- Adding copies is annoying but often not impossible
- Ordering objects in an S-order may
  - improve user-friendliness
  - save time
  - save space
  - simplify visual representations of classifications



tree



S-sortable



general hierarchy

Transfer

Objects in libraries, ware-houses, and stores are only *nearly* linearly arranged:

⇒ Second (and third) dimension can be used in order to avoid duplications

Possible minimality criteria

total list:

a i u Ṇ ṛ ḷ K e o Ṅ ai au C h y v r Ṭ l Ṇ ñ m ṅ ṇ n M jh bh Ñ gh ḍh dh Ṣ j b g ḍ d Ś kh ph ch ṭh th c ṭ t V k p Y ś ṣ s R h L

1. total list is of minimal length;
2. sound list is of minimal length;
3. *anubandha* list is of minimal length;
4. total list is as short as possible while the *anubandha* list is minimal;
5. total list is as short as possible while the sound list is minimal;

Possible minimality criteria

sound list:

a i u ṛ ḷ e o ai au h y v r l ñ m ṅ ṇ n jh bh gh ḍh dh j b g ḍ d kh ph ch ṭh th c ṭ t k p ś ṣ s h

1. total list is of minimal length;
2. sound list is of minimal length;
3. *anubandha* list is of minimal length;
4. total list is as short as possible while the *anubandha* list is minimal;
5. total list is as short as possible while the sound list is minimal;

Possible minimality criteria

anubandha list:

Ṇ K Ṅ Ś C Ṭ Ṇ V Y M R Ñ L Ṣ

1. total list is of minimal length;
2. sound list is of minimal length;
3. *anubandha* list is of minimal length;
4. total list is as short as possible while the *anubandha* list is minimal;
5. total list is as short as possible while the sound list is minimal;

Possible minimality criteria

a i u Ṇ ṛ ḷ K e o Ṅ ai au C h y v r Ṭ l Ṇ ñ m ṅ ṇ n M jh bh Ñ gh ḍh dh Ṣ j b g ḍ d Ś kh ph ch ṭh th c ṭ t V k p Y ś ṣ s R h L

1. total list is of minimal length;
2. sound list is of minimal length;
3. *anubandha* list is of minimal length;
4. total list is as short as possible while the *anubandha* list is minimal;
5. total list is as short as possible while the sound list is minimal;

Possible minimality criteria

a i u Ṇ ṛ ḷ K e o Ṅ ai au C h y v r Ṭ l Ṇ ñ m ṅ ṇ n M jh bh Ñ gh ḍh dh Ṣ j b g ḍ d Ś kh ph ch ṭh th c ṭ t V k p Y ś ṣ s R h L

1. total list is of minimal length;
2. sound list is of minimal length;
3. *anubandha* list is of minimal length;
4. total list is as short as possible while the *anubandha* list is minimal;
5. total list is as short as possible while the sound list is minimal;

⇒ duplicating sounds is worse than adding *anubandhas*

Principle of economy

Staal 1962

Another general principle is also implicitly used by Pāṇini. This is the famous economy criterion [. . .] In accordance with this principle each linguistic rule should be given in the shortest possible form, whereas the number of metalinguistic symbols should be reduced as far as possible.

⇒ 5. criterion of minimality: total list is as short as possible while the sound list is minimal

Example: semi-formal argument

Kiparsky 1991

The reasoning from economy goes like this. To be grouped together in a pratyāhāra, sounds must make up a continuous segment of the list. Economy requires making the list as short as possible, which means avoiding repetitions of sounds, and using as few markers as possible.
Consequently, if class A properly includes class B, the elements shared with B should be listed last in A; the marker that follows can then be used to form pratyāhāras for both A and B. In this way the economy principle, by selecting the shortest grammar, determines both the ordering of sounds and the placement of markers among them.

Example: semi-formal argument

Kiparsky 1991

The reasoning from economy goes like this. To be grouped together in a pratyāhāra, sounds must make up a continuous segment of the list. Economy requires making the list as short as possible, which means avoiding repetitions of sounds, and using as few markers as possible. Consequently, if class A properly includes class B, the elements shared with B should be listed last in A; the marker that follows can then be used to form pratyāhāras for both A and B. In this way the economy principle, by selecting the shortest grammar, determines both the ordering of sounds and the placement of markers among them.

Example: semi-formal argument

Śivasūtras:

a i u Ṇ ṛ ḷ K e o Ṅ ai au C h y v r Ṭ l Ṇ ñ m ṅ ṇ n M jh bh Ñ gh ḍh dh Ṣ j b g ḍ d Ś kh ph ch ṭh th c ṭ t V k p Y ś ṣ s R h L

$aK = \{a, i, u, ṛ, ḷ\}$, $iK = \{i, u, ṛ, ḷ\}$ and $uK = \{u, ṛ, ḷ\} \Rightarrow a < i < u < ṛ, ḷ$

but:

$jhL =$
$\{h, s, ṣ, ś, p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhR =$
$\{s, ṣ, ś, p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhY = \{p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhŚ = \{d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$ and

$jhṢ = \{dh, ḍh, gh, bh, jh\}$

$\Rightarrow h < s, ṣ, ś < p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d < ḍ, g, b, j <$ $dh, ḍh, gh, bh, jh$

Example: semi-formal argument

Śivasūtras:

a i u Ṇ ṛ ḷ K e o Ṅ ai au C h y v r Ṭ l Ṇ ñ m ṅ ṇ n M jh bh Ñ gh ḍh dh Ṣ
j b g ḍ d Ś kh ph ch ṭh th c ṭ t V k p Y ś ṣ s R h L

$aK = \{a, i, u, ṛ, ḷ\}$, $iK = \{i, u, ṛ, ḷ\}$ and $uK = \{u, ṛ, ḷ\} \Rightarrow a < i < u < ṛ,ḷ$

but:

$jhL =$
$\{h, s, ṣ, ś, p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhR =$
$\{s, ṣ, ś, p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhY = \{p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$

$jhŚ = \{d, ḍ, g, b, j, dh, ḍh, gh, bh, jh\}$ and

$jhṢ = \{dh, ḍh, gh, bh, jh\}$

$\Rightarrow h < s, ṣ, ś < p, k, t, ṭ, c, th, ṭh, ch, ph, kh, d < ḍ, g, b, j <$
$dh, ḍh, gh, bh, jh$

Literature

- Kiparsky, P. (1991), Economy and the construction of the Śivasūtras. In: M. M. Deshpande & S. Bhate (eds.), *Pāṇinian Studies*, Michigan: Ann Arbor.
- Petersen, W. (2008), Zur Minimalität von Pāṇinis *Śivasūtras* – Eine Untersuchung mit Mitteln der Formalen Begriffsanalyse. PhD thesis, university of Düsseldorf.
- Petersen, W. (2009), On the Construction of Sivasutra-Alphabets. In: A. Kulkarni and G. Huet (eds.): *Sanskrit Computational Linguistics.* LNCS 5406, Springer.
- Staal, F. (1962), A Method of Linguistic Description. Language **38**, 1-10.
- Zschalig, C. (2007), Bipartite Ferrers-graphs and planar concept lattices. In: S. O. Kuznetsov and S. Schmidt (eds.): *Proceedings of the 5th ICFCA.* LNCS 4390, p. 313–327, Springer.

Origin of Pictures

- libraries (left): http://www.meduniwien.ac.at/medizinischepsychologie/bibliothek.htm
- libraries (middle): http://www.math-nat.de/aktuelles/allgemein.htm
- libraries (right): http://www.geschichte.mpg.de/deutsch/bibliothek.html
- warehouses: http://www.metrogroup.de/servlet/PB/menu/1114920_l1/index.html
- stores: http://www.einkaufsparadies-schmidt.de/01bilder01/