Lecture 1a: OT – An Introduction

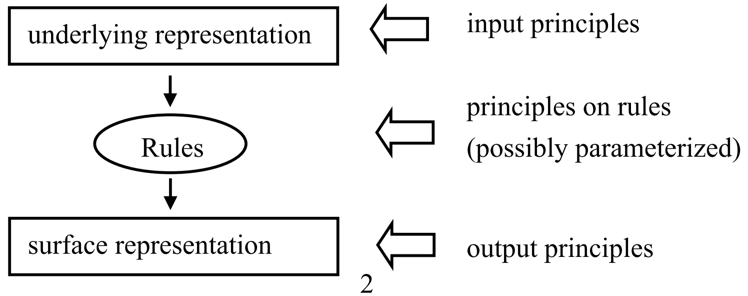
1. Generative linguistics and OT
2. Ethics for robots: a first illustration of OT
3. Voicing contrasts in Dutch and English
4. Basic architecture of standard OT
5. Historical antecedents of OT
6. The rise of OT
7. OT – a new paradigm in linguistics?

1 Generative Linguistics and OT

In Generative Linguistics all the constraints have been viewed *inviolable* within the relevant domain (phonology, syntax).

Basic Picture
(1960s)

Inflation of Principles
(1970s and 1980s)



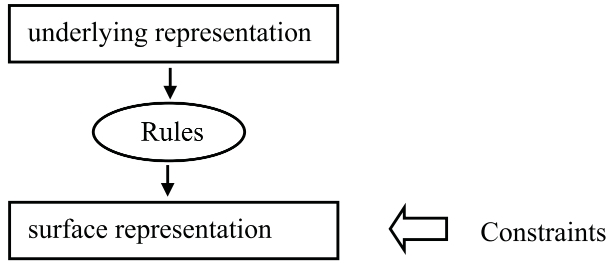
Standard Scenario of grammatical explanation

Separation: The status of a particular form with respect to a particular constraint does not depend on the status of any other form with respect to that constraint. In this sense, forms are *separated* from each other.

Inviolability: Constraints, rule systems and principles are *inviolable*. If a form violates a particular principle that violation has an effect on the grammatical status of the object.

Prince & Smolensky: ‘Optimality Theory’
(Arizona Phonology Conference in Tucson, April 1991).

Surface forms of language reflect resolution of conflicts between competing (violable) constraints



Optimality Scenario of grammatical explanation

Connection: The status of a particular form with respect to a particular constraint is determined by comparing it with the analysis of other objects. The grammar favours the competitor that best satisfies the constraint. In this sense, forms are *connected* with each other.

Violability: Constraints, rule systems and principles are *violable*. If a form violates a particular constraint C, but no competing form present a lesser violation, that violation of C may result in no detectable deviance.

2 Ethics for robots: A first illustration of OT

Isaac Asimov described what became the most famous view of ethical rules for robot behaviour in his “three laws of robotics”
(Thanks to Bart Geurts for drawing my attention to this example):

Three Laws of Robotics:

1. A robot may not injure a human being, or, through inaction, allow a human being to come to harm.

2. A robot must obey the orders given it by human beings, except where such orders would conflict with the First Law.

3. A robot must protect its own existence, as long as such protection does not conflict with the First or Second Law.

(Asimov, Isaac: *I, Robot*. Gnome Press 1950)

Analysis

This sentence actually contains three independent constraints:

1. A robot may not injure a human being, or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given it by human beings.
3. A robot must protect its own existence.

From an optimality theory point of view, we can think of this as three constraints, where each one overrides the subsequent. The effect of overriding is described by a ranking of the constraints:

$$1 \gg 2 \gg 3,$$

i.e.: ***INJURE HUMAN** ≫ **OBEY ORDER** ≫ **PROTECT EXISTENCE**

Story A: ***H**uman* says to ***R**obot*: Kill my wife!

1. ***R*** kills ***H***’s wife
2. ***R*** kills ***H*** (who gave him the order)
3. ***R*** doesn’t kill anyone
4. ***R*** kills himself.

Standard optimality tableau
(☞ marks the optimal candidate, "***!**" the fatal constraint violation):

TABLEAU FOR STORY A	***INJURE HUMAN**	**OBEY ORDER**	**PROTECT EXISTENCE**
1. ***R*** kills ***H***’s wife	*!		
2. ***R*** kills ***H***	*!	*	
☞ 3. ***R*** doesn’t kill anyone		*	
4. ***R*** kills himself		*	*!

Comment

In the example, the story relates to a certain situation type that generates the possible reactions 1-4.

R*'s** optimal reaction to ***H's order is to do nothing (line 3). All other reactions are *suboptimal*.

The indication of fatal constraint violation isn't part of the tableaus. It is only to shift the reader's attention to the crucial points.

Story B: *Human* says to ***R**obo*t: Kill my wife or I kill her!

TABLEAU FOR STORY B	*Injure Human	Obey Order	Protect Existence
☞ 1. ***R*** kills ***H***'s wife	*		
2. ***R*** kills ***H***	*	*	
3. ***R*** doesn't kill anyone	*	*	
4. ***R*** kills himself	*	*	*

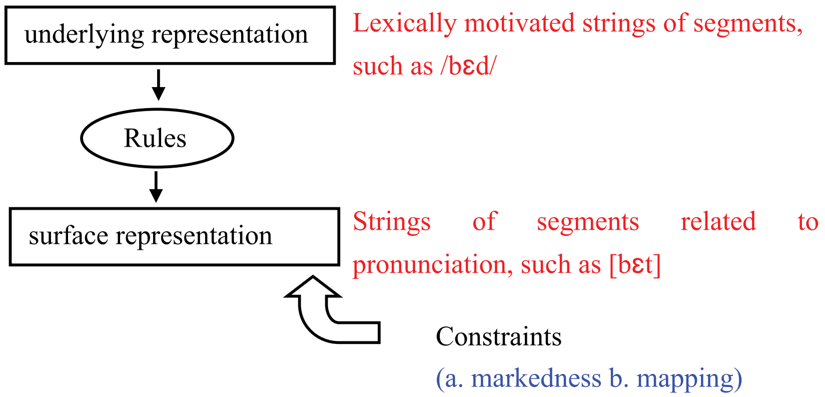
R*'s** optimal reaction to ***H's order is to kill ***H***'s wife.

Story C: *Human* says to ***R**obo*t: Kill my wife or I destroy you!

TABLEAU FOR Story C	*Injure Human	Obey Order	Protect Existence
1. ***R*** kills ***H***'s wife	*		
2. ***R*** kills ***H***	*	*	
☞ 3. ***R*** doesn't kill anyone		*	*
☞ 4. ***R*** kills himself		*	*

There are two optimal reaction to ***H***'s order: ***R*** does nothing (then he is killed by ***H***), or he kills himself.

3 Voicing contrasts in Dutch and English



Lexically motivated strings of segments, such as /bɛd/

Strings of segments related to pronunciation, such as [bɛt]

Constraints
(a. markedness b. mapping)

Phenomenon

Coda obstruents are voiceless in Dutch but voiced in English. Consequently, Dutch neutralizes voicing contrasts in final obstruents and English preserves them:

(1) a.	/bɛd/	[bɛt]	‘bed’	*Dutch*
b.	/bɛd-ən/	[bɛ.dən]	‘beds’	
c.	/bɛt/	[bɛt]	‘(I) dab’	
d.	/bɛt-ən/	[bɛ.tən]	‘(we) dab’	

(2) a.	/bɛd/	[bɛd]	‘bed’	*English*
b.	/bɛt/	[bɛt]	‘bet’	

Two types of constraints

In OT Phonology, we have two kinds of constraints, *markedness conditions*, which evaluate the complexity of the output, and *mapping constraints* which evaluate the difference between input and output.

Markedness Condition

Obstruents must not be voiced in coda position Coda/*Voice

Mapping Constraints

The specification for the feature VOICE of an input segment must be preserved in its output correspondent Faith[voice]

Ranking A: Coda/*Voice ≫ Faith[voice]

Input: /bɛd/	Coda/*Voice	Faith[voice]
1 ☞ [bɛt]		*
2 [bɛd]	*!	

Input: /bɛt/	Coda/*Voice	Faith[voice]
1 ☞ [bɛt]		
2 [bɛd]	*	*

This ranking describes the situation in Dutch where voicing contrasts in final obstruents are neutralized.

Ranking B: Faith[voice] ≫ Coda/*Voice

Input: /bɛd/	Faith[voice]	Coda/*Voice
1 [bɛt]	*!	
2 ☞ [bɛd]		*

Input: /bɛt/	Faith[voice]	Coda/*Voice
1 ☞ [bɛt]		
2 [bɛd]	*	*

This ranking describes the situation in English where voicing contrasts in final obstruents are preserved.

What does this example illustrate?

- Two types of constraints in phonology: markedness conditions and mapping constraints (*faithfulness* constraints).
- Markedness is a grammatical factor that exert pressure toward unmarked structure.
- Faithfulness is a grammatical factor that exert pressure toward preserving lexical contrasts.
- Constraints are conflicting. There is no such thing as a ‘perfect’ output. Violations of lower ranked constraints may be tolerated in order to satisfy a higher ranked constraint.

- The Grammar selects an *optimal* output, i.e. an output that best satisfies the system of ranked constraints. More formally:

> A candidate w is considered to be optimal iff for each competitor w’, the constraints that are lost by w must be ranked lower than at least one constraint lost by w’.

- The constraints are universal, their ranking is language particular.

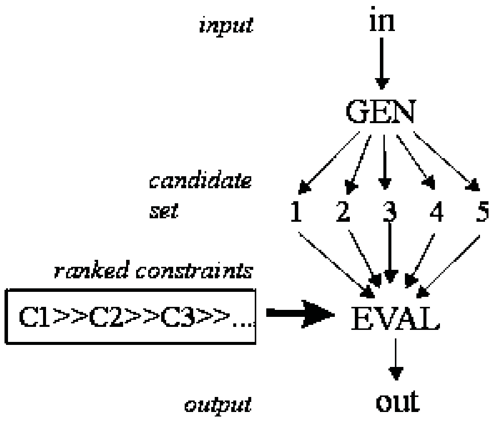
Suggested hypotheses

- Considering all rankings of a given system of (universal) constraints provides a system of language universals for the domain under discussion.
- The different rankings of some (sub)system of constraints provide a typology of natural languages (*factorial typology*).
- Ungrammatical outputs (*) *out* are explained by ‘blocking’: there is an alternate output that satisfies the system of ranked constraints better than *out*.

4 Basic architecture of standard OT

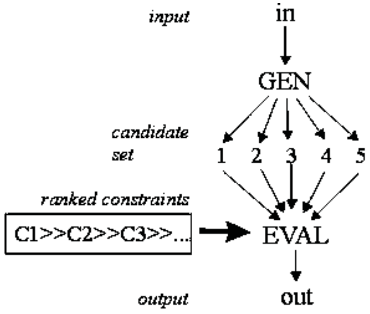
The GENerator determines the possible inputs, the possible outputs, and the possible correspondences between inputs and outputs. For a given input, GEN creates a candidate set of possible outputs.

OT doesn’t provide a ‘theory’ for GEN, rather it presupposes it. (OT is not a theory of representations!)



The universal CONstraint set is assumed to be part of our innate knowledge of language. Each constraint can be seen as a *markedness statement*. Constraints can be ranked. This reflects the relative importance of the different markedness statements.

EVALuation is a mechanism which selects the optimal candidate(s) from the candidate set generated by GEN. **EVAL** makes use of the ranking of the violable constraints. The optimal output, the one that is selected by **EVAL** is the one that best satisfies these constraints.



5 Historical antecedents of OT

- Panini's principle in **Phonology**: The application of a rule depends on the failure of a more specific competing rules to apply.
- Specificity in **Morphology**: the most specific vocabulary entry among a set of competitors takes precedence over less specified entries.
- Markedness Theory of Generative **Grammar**
- Hypothetical **Reasoning** (Nicholas Rescher, 1964)
  When Verdi and Bizet were compatriots, then…
  {comp(v,b)↔country(v)=country(b), country(v)=It, country(b)=Fr}
  Definition are ranked higher than facts!

- In **Pragmatics**, the Gricean conversational maxims license an utterance of a particular proposition in a given context only if it fares better (with respect to relevance, for example) than a set of competitors.
- Garden path phenomena in **Natural Sentence Processing**.
  The boat floated down the river sank / and sank
  (based on preferences for the resolution of local ambiguities)

6 The rise of OT

- **The first papers**
  - Alan Prince & Paul Smolensky (1993): Optimality theory: Constraints interaction in generative grammar. *Phonology*
  - John McCarthy & Alan Prince (1993b): Prosodic morphology I: constraint interaction and satisfaction. *Morphology*
- **OT and syntax**
  - Jane Grimshaw 1997: Projection, heads and optimality.
  - Pilar Barbosa, Danny Fox, Paul Hagstrom, Martha McGinnis, & David Pesetsky (eds.): Is the best good enough?

- **Semantics and Interpretation**
  - Helen de Hoop & Henriette de Swart (Eds.) (2000): Papers on Optimality Theoretic Semantics (*J. of Semantics* 17)
  - Reinhard Blutner & Henk Zeevat (Eds.) (2003): Optimality Theory and Pragmatics (Palgrave Macmillan)

Rutgers Optimality Archive
http://roa.rutgers.edu

7 OT - a new paradigm in linguistics?

- Overcoming the gap between competence and performance
- New, powerful learning theory that implicitly makes use of negative examples
- Based on a connectionist architecture (Smolensky's harmony theory). OT aims to integrate symbolist and sub-symbolist (connectionist) systems.
- Interesting from a computational perspective (robust parsing)
- Interesting from an evolutionary perspective (e.g. language change)

Lecture 1b: Phonology of the Syllable

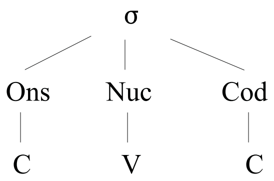
1. Inputs and outputs
2. The optimal correspondence between input and output
3. The Jacobson Typology
4. Conclusions

1 Inputs and outputs

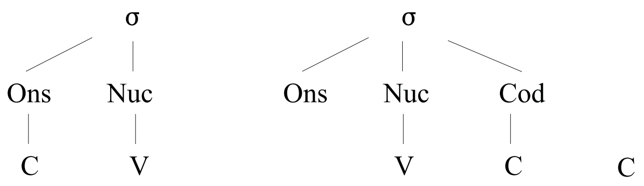
- Inputs are typically taken as simple strings of segments. This strings have to be motivated by morphology.
- Outputs are taken as syllabified strings.
- The output of the phonology is subject to phonetic interpretation. Underparsed segments ⟨x⟩ are not phonetically realized (deletion). Overparsed elements □ are phonetically realized through some process of filling in default featural values (epenthesis)

The structure of the output: Syllables

- Adopt the (simplifying) analysis that the syllable node σ must have a daughter *Nuc* and may have as leftmost and rightmost daughters the nodes *Ons* and *Cod*.
- The nodes *Ons, Nuc,* and *Cod*, in turn, may each dominate C´s and V´s, or they may be empty.
- For simplifying further we assume that *Nuc* dominates exactly one V, and *Ons* and *Cod* dominate at most one C.



Tree and string notation



.CV.□VC.⟨C⟩

.X. the string X is a syllable

⟨x⟩ the element x has no mother, it is free (not syllabified)

□ a node Ons, Nuc, or Cod is empty

Example

Input	*Output*	*Phonetic*
/no-N-koma-i/	.noɳ.ko.ma.□i. *.noɳ.ko.ma.i.	noɳkoma**t**i *noɳkomai
Consonant epenthesis in Axininca Campa (*noɳkomati* 'he will paddle')		

The Generator

It can be assumed to be relatively free. Each possible input is paired with each possible output supposed the corresponding sequences of the terminal elements agree (ignoring □´s).

input: /VCVC/

outputs	a.	.V.CVC.	an onsetless open syllable followed by a closed syllable
	b.	⟨V⟩.CV.⟨C⟩	one open syllable; the initial V and final C are not parsed into syllable structure; this is indicated by ⟨ ⟩
	c.	.□V.CV.⟨C⟩	a sequence of two open syllables. The onset of the first syllable is unfilled (notated □). Phonetically, this is realized as an epenthetic consonant.

2 The optimal correspondence between input and output

Some typical properties of syllables (Markedness Conditions)

Syllables must have onsets	Onset
Syllables must not have a coda	NoCoda

Mapping Constraints

No changes in the mapping from input to output	Faithfulness
- Underlying segments must be parsed into syllable structure	Parse
- Syllable positions must be filled with underlying segments	Fill

3 The Jacobson Typology

> *There are languages lacking syllables with initial vowels and/or syllables with final consonants, but there are no languages devoid of syllables with initial consonants or of syllables with final vowels.* (Jakobson 1962: 526)

These constraints yield exactly four possible systems:

		Onsets	
		required	optional
Codas	forbidden	CV *Senufo (Guinea)*	(C)V Hawaiian
	optional	CV(C) Yawelmani (Cal)	(C)V(C) English

It excludes
V, VC, V(C),
CVC, (C)VC

Explaining the Jacobson typology

Consider the system of constraints {Faith , Onset, NoCoda}

A: Ranking Faith ≫ Onset, NoCoda

Input: /pipaptaop/	Faith	Onset	NoCoda
1 ☞ pi.pap.ta.op		*	**
2 pip.ap.ta.op		**	***
3 pi.pa.⟨p⟩.ta.□o.⟨p⟩	***		
4 pi.pap.ta.□op	*		**
5 pi.pa.⟨p⟩.ta.o⟨p⟩	**	*	

The optimal output realizes syllables (C)V(C)

B: Ranking Onset, NoCoda ≫ Faith

Input: /pipaptaop/	Onset	NoCoda	Faith
1 pi.pap.ta.op	*	**	
2 pip.ap.ta.op	**	***	
3 ☞ pi.pa.⟨p⟩.ta.□o.⟨p⟩			***
4 pi.pap.ta.□op		**	*
5 pi.pa.⟨p⟩.ta.o⟨p⟩	*		**

The optimal output realizes syllables CV

C: Ranking ONSET ≫ FAITH ≫ NOCODA

Input: /pipaptaop/	ONSET	FAITH	NOCODA
1 pi.pap.ta.op	*		**
2 pip.ap.ta.op	**		***
3 pi.pa.⟨p⟩.ta.□o.⟨p⟩		***	
4 ☞ pi.pap.ta.□op		*	**
5 pi.pa.⟨p⟩.ta.o⟨p⟩	*	**	

The optimal output realizes syllables CV(C)

D: Ranking NOCODA ≫ FAITH ≫ ONSET

Input: /pipaptaop/	NOCODA	FAITH	ONSET
1 pi.pap.ta.op	**		*
2 pip.ap.ta.op	***		**
3 pi.pa.⟨p⟩.ta.□o.⟨p⟩		***	
4 pi.pap.ta.□op	**	*	
5 ☞ pi.pa.⟨p⟩.ta.o⟨p⟩		**	*

The optimal output realizes syllables (C)V

4 Conclusion

Since ONSET and NOCODA don't directly interact there are four possible empirically different rankings of the system {FAITH , ONSET, NOCODA} repeated in the table:

Rankings		Types
A	FAITH ≫ ONSET, NOCODA	(C)V(C) *English*
B	ONSET, NOCODA ≫ FAITH	CV *Senufo*
C	ONSET ≫ FAITH ≫ NOCODA	CV(C) *Yawelmani*
D	NOCODA ≫ FAITH ≫ ONSET	(C)V *Hawaiian*

The four possible rankings describe all and only the possible syllable type systems.

	ONSET ≫ FAITH	FAITH ≫ ONSET
NOCODA ≫ FAITH	CV *Senufo (Guinea)*	(C)V *Hawaiian*
FAITH ≫ NOCODA	CV(C) *Yawelmani (Cal)*	(C)V(C) *English*

In general, for any set of freely rankable constraints, OT predicts the possibility of languages corresponding to each possible ranking. This is called the *Factorial Typology.* The factorial typology that corresponds to the Jacobson typology was proposed first by Prince & Smolensky (1993)

Lecture 2a: Phonology – Word Stress

1. Inputs and outputs
2. Cross-linguistic preferences
3. OT and stress
4. The autonomy thesis
5. Autonomy breaking – the interaction of stress and syllabification

1 Inputs and outputs

The representational basis is *metrical phonology* (e.g. Liberman & Prince 1977; Halle & Vergnoud 1987; Hayes 1980, 1995). The central assumption is that stress is a rhythmic phenomenon, encoded by strong-weak relations between syllables.

In short, every prosodic word consists of patterns of alternation (termed *a foot*). Each foot contains one stressed and at most one unstressed syllable. The most common two patterns are

- *trochees* (stressed syllable on the left and at most one stressless syllable on the right), as in English, and
- *iambs* (a stressless-stressed sequence of syllables).

We use brackets to mark feet and áccents to mark prominent vowels. There are unfooted syllables (always stressless).

- **Inputs** are taken as syllabified strings of segments (motivated by morphology).
  Examples: /mi.nə.so.tə/ ; /ə.me.ri.kə/.
- **Outputs** are taken as strings which are analysed at foot-level too. We use brackets to mark feet and áccents to mark prominent vowels.
  Example: (mí.nə)(só.tə) ; ə(mé.ri)kə.

- **The Generator** can be assumed to be relatively free. Each possible input is paired with each possible output supposed the corresponding sequences of the terminal elements agree. The following are outputs generated by the input

/ə.me.ri.kə/:

(1) ə(mé.ri)kə
(2) (ə.mé)(rí.kə)
(3) ə.me(rí.kə)
(4) (´ə.me)ri.kə
(5) (´ə.me)ri(k´ə), ...

And these are several outputs generated by the input

/mi.nə.so.tə/:
(1) (mí.nə)(só.tə)
(2) mi.nə(só.tə)
(3) (mí.nə)so.tə
(4) mi(nə.só)tə
(5) (mi.n´ə)(só.tə)
(6) mi.nə.so.tə, ...

Notice that we drop dots if no misunderstandings are possible. For example, we write X(Y)Z instead of .X.(Y).Z.

2 Cross-linguistic preferences

The four best known common properties of stress languages:

- **The culminative property:** *Words have single prosodic peak.* Many languages impose this requirement on content words only, function words are prosodically dependent on content words.
- **The demarcative property:** *Stress tends to be placed near edges words.* Crosslinguistically favoured positions for primary word stress are (a) the initial syllable, (b) the prefinal syllable and (c) the final syllable (ranked in decreasing order of popularity among the world's languages.

- **The rhythmic property:** *Stress tends to be organized in rhythmic patterns, with strong and weak syllables spaced apart at regular intervals.* The smallest units of linguistic rhythm are metrical feet. *Trochees* are preferred. Languages may also select *iambs* .
- **Quantity-sensitivity:** *Stress prefers to fall on elements which have some intrinsic prominence*. For example, stress tends to be attracted by long vowels rather than by short ones. And stressed vowels tend to lengthen, increasing syllable weight. Mutually reinforcing relations of prominence and quantity are highly typical for stress systems.

3 OT and stress

We present a roughly simplified analysis (based on Hammond 1997) and start with the following three constraints:

Constraint corresponding to the culminative property

Words must be stressed	ROOTING

(another name for this constraint is LXWD.PRWD: grammatical words must have prosody)

Constraints corresponding to the rhythmic property

Feet are trochaic	TROCHEE
Two unfooted syllables cannot be adjacent	PARSE-SYLLABLE

Ranking for English

ROOT >> TROCH >> PARSE SYLL

Example

Input: /ə.me.ri.kə/	ROOT	TROCH	PARSE SYLL
1 ☞ ə(mé.ri)kə			
2 (ə.mé)(rí.kə)		*	
3 ə.me(rí.kə)			*
4 (´ə.me)ri.kə			*
5 ☞ (´ə.me)ri(k´ə)			
6 ☞ (´ə.me)(rí.kə)			
7 ə.me.ri.kə	*		

The system predicts multiple (optimal) outputs. For unique solutions we have to add some more constraints. Essentially, we have to consider constraints due to the demarcative property and the property of quantity-sensitivity.

Constraint corresponding to quantity-sensitivity

Heavy syllables are stressed WEIGHT-TO-STRESS PRINCIPLE (WSP)

Ranking for English

ROOT, WSP >> TROCH >> PARSE SYLL

Example (continued)

Input: / ə.me.ri.kə /	ROOT	WSP	TROCH	PARSE SYLL
1 ☞ ə(mé.ri)kə				
2 (ə.mé)(rí.kə)			*	
3 ə.me(rí.kə)		*		*
4 (´ə.me)ri.kə		*		*
5 (´ə.me)ri(k´ə)		*		
6 (´ə.me)(rí.kə)		*		
7 ə.me.ri.kə	*	*		

4 The autonomy thesis

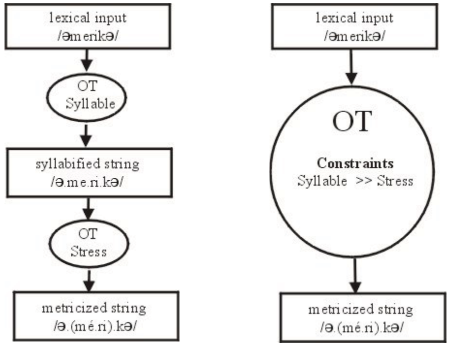
Syllabification parses a strings of segments into a sequence of substrings (called syllables). Metrical phonology adds another level of analysis and parses syllables into foots and assigns stress. In the previous section it was assumed that syllabification is independent on stress patterns. In terms of a classical cognitive architecture that means that the outputs of the system of syllabification are the inputs of the stress system. Taken this architecture, the stress system cannot have an effect on syllabification. We may refer to this hypothesis by saying

Syllabification is autonomous with regard to stress

Using OT there are two ways to formulate an autonomy thesis.

The first way, called *representational autonomy*, is a direct transportation from classical architecture into OT. The other way uses hierarchical ranking to express autonomy. It is called *dominance-based autonomy*.

Although both ways are equivalent, they are very different from a conceptual point of view and may be sources of quite different inspirations. This becomes important when violations of autonomy are envisaged.



a. representational autonomy

b. dominance-based autonomy

Representational Autonomy:

- very close to classical (rule-based) architecture.
- separation of representational units and constraints
- the outputs for one system are the inputs for the other one

Dominance-based Autonomy:

- two levels of representation only (input, output)
- no separation of representational elements necessary
- strict separation of the constraints
- the constraints of the autonomous system outrank the constraints of the dependent system

5 Autonomy breaking – the interaction of stress and syllabification

There is ample evidence that syllabification is influenced by stress, contrary to the autonomy thesis.

As a case in point consider pronunciation of /h/. This phoneme is pronounced at the beginning of words (+syllables) but not at the ends. Consequently, we can use the pronunciation of /h/ as a check for syllabification.

Now consider the pair *véhicle* – *vehícular*. In the first case /h/ isn't pronounced, in the second case it is. Consequently, our test suggests the syllabification in (i) which contrasts with standard theory (ii):

(i) /véh.i.cle/ - /ve.hí.cu.lar/ (empirically)
(ii) /ve.hi.cle/ - /ve.hi.cu.lar/ (standard theory)

Conclusion: Stress influences syllabification. Intervocalic consonants are affiliated with the syllable to its left if the following vowel is stressless.

Another example is aspiration. For example, we have the generalization that /t/ is aspirated syllable-initially but not at the ends. Consider for

example the word *hotél* where the /t/ is aspirated contrasting with the word *vánity* where /t/ isn't aspirated.

(i) /ho.tél/ - /vá.nit.y/ (empirically)
(ii) /ho.tel/ - /va.ni.ty/ (standard theory)

The observed facts seem to obey the following constraint:

Constraint corresponding to a kind of demarcative property

Stressless medial syllables are onsetless NoOnset

Obviously, this constraint is part of the stress family and conflicts with the constraint Onset of syllable theory. The constraint NoOnset must outrank Onset to be effective:

NoOnset >> Onset

Autonomy breaking occurs since autonomy of syllable theory would demand that all constraints of syllable theory outrank those of the stress theory.

Interaction of stress and syllabification

Input: /vehikl/	NoOnset	Onset	NoCoda
(vé.hikl)	*		*
☞ (véh.ikl)		*	**

Input: /vehiculχ/	NoOnset	Onset	NoCoda
☞ ve(hí.cu)lə			
veh(í.cu)lə		*	*

Lecture 2b: Computational Aspects of OT

(based on material by J. Kuhn)

1. Computational issues
2. Some background on formal languages
3. Finite-state transducers (FSTs) and rational relations
4. Computational OT based on FSTs
5. Bidirectionality

1 Computational Issues

- **Infinity of candidate set**

Naïve evaluation algorithm for an OT system

1. construct candidates
2. apply constraints
3. pick most harmonic candidate(s)

– Since candidates can violate faithfulness constraints, the candidate set is generally infinite.

– Even with large finite candidate sets, naïve processing will get extremely costly

- **Directionality issues**

– Definition of expressive optimization is based on generation from underlying input forms – how can one decide that a given surface form is an optimal output?

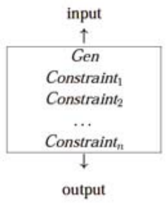
– requires processing in opposite direction to determine possible inputs (cf. robust interpretive parsing)

– this may cause additional infinity issues (even in unidirectional optimization)

- **Ways of addressing the infinity issue**

– Control candidate construction based on constraint violations dynamic (or chart-based) programming (Tesar 1995, Kuhn 2000)

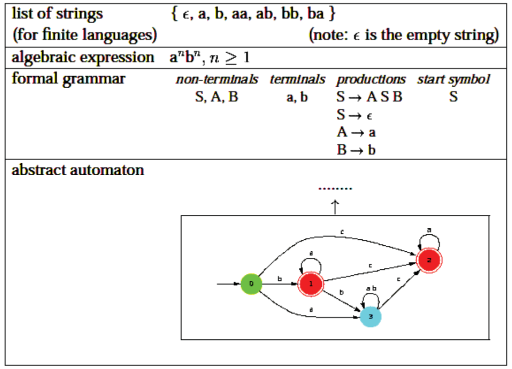
– Pre-compute the set of distinctions between relevant candidates and the respective winner (no online construction of competing candidates).
(Karttunen 1998, based on results by Frank and Satta 1998)



2 Some Background on Formal Languages

Formal languages are conceptualized as sets of strings over a given alphabet of atomic symbols (Σ).

There are at least four different ways of characterizing a formal language:



list of strings (for finite languages)	{ ϵ, a, b, aa, ab, bb, ba } (note: ϵ is the empty string)
algebraic expression	$\mathrm{a}^n\mathrm{b}^n$, $n \geq 1$
formal grammar	*non-terminals*: S, A, B *terminals*: a, b *productions*: S → A S B S → ϵ A → a B → b *start symbol*: S
abstract automaton	.... ↑

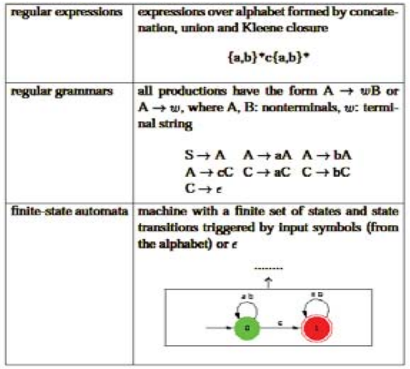
c
a
a
c
b
0
1
2
b
a b
c
a
3

- **Classes of formal languages (Chomsky hierarchy)**

– regular languages
– context-free languages
– context-sensitive languages
– recursively enumerable languages

The classification based on restrictions on formal grammars.
Equivalent classes follow from specific types of automata.

For instance, regular languages can equivalently be characterized in the following three ways:



regular expressions	expressions over alphabet formed by concatenation, union and Kleene closure $\{a,b\}^*c\{a,b\}^*$
regular grammars	all productions have the form A → wB or A → w, where A, B: nonterminals, w: terminal string S → A A → aA A → bA A → cC C → aC C → bC C → ϵ
finite-state automata	machine with a finite set of states and state transitions triggered by input symbols (from the alphabet) or ϵ .... ↑

- **Important properties of regular languages:**

– closed under union, intersection, complementation

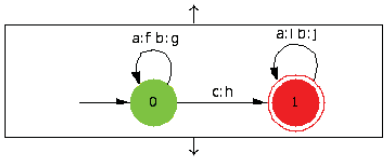
– recognizers are very efficient: linear time complexity (i.e., computation with double input length will only take twice as long)

– *Note:* Languages like $a^n b^n$ (n≥1) are not regular (but context-free)

3 OT andFinite-state transducers (FSTs) and rational relations

- a finite-state automaton with two tapes is called a finite-state transducer (FST)
- A FST specifies a relation between two regular languages (so-called rational relation)
  - state transitions are marked with two symbols a:b
  - extension of regular expression notation is used to specify transducers
  - one can view one side of the transducer as the input, which is transformed into the form(s) on the other side
  - nondeterminism may lead to several possibilities in the mapping
  - the upper and lower side can be swapped

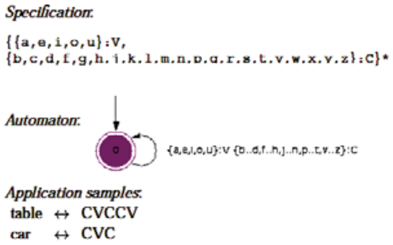
Example 1



abcab ↔ fghij

- FSTs are widely used for phonological, morphological, and “shallow” syntactic processing

Example 2



Specification:

{{a,e,i,o,u}:V,
{b,c,d,f,g,h,j,k,l,m,n,p,q,r,s,t,v,w,x,y,z}:C}*

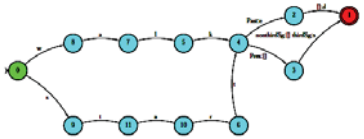
Automaton:

Application samples:

table	↔	CVCCV
car	↔	CVC

Example 3

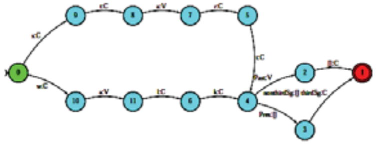
```
[{ [w,a,l,k] x [w,a,l,k],
   [s,t,a,r,t] x [s,t,a,r,t] },
 { Past x [e,d],
   [ Pres x [],
   { thirdSg x [s],
     nonthirdSg x [] } ] }        ]
```



walk Past	↔	walked
start Pres thirdSg	↔	starts
start Pres nonthirdSg	↔	start

Example 4: Composition of ex3 and ex2

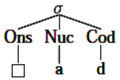
FST1 .o. FST2 maps *u* to *w* iff there is some *v* s.th. FST1 maps *u* to *v* and FST2 maps *v* to *w*.
The composition of two FSTs can be compiled into a single transducer as the following example illustrates:



walk Past	↔	CVCCVC
start Pres thirdSg	↔	CCVCCC
start Pres nonthirdSg	↔	CCVCC

4 Computational OT based on FSTs

Basic references: Frank and Satta 1998, Karttunen 1998

(input /ast/)	Notation (Karttunen 1998):
	O[ ] N[a] D[d] X[b]

Gen can be defined as a transducer:

- upper side: OT input (underlying form); string of Vs and Cs.
- lower side: all possible syllabifications (including faithfulness violations)

Simplified part of the specification expression (for onset & nucleus):

```
[{[ []:’O[’, {b:b,c:c,d:d ... },     []:’]’ ],
    []},
  [ []:’N[’, {a:a,e:e,i:i,o:o,u:u}, []:’]’ ]
]*
```

ba ↔ a. N[] D[b] N[a]   .□b.a.
b. N[] O[b] N[a]   .□.ba.
c. N[] X[b] N[a]   .□.<b>.a.
d. O[b] N[] N[a]
e. O[b] N[a]
f. O[b] N[a] N[]
g. O[b] N[a] D[]
h. O[] X[b] N[a]
i. X[b] N[] N[a]
j. X[b] N[a]
k. X[b] N[a] N[]
l. X[b] N[a] D[]
m. X[b] O[] N[a] ; etc.

Formalizing the constraints

Each constraint is expressed as a regular language.

Markedness

NOCODA: the language that does not contain ’D[’ : `~$’D[’`

ONSET: the language in which ’N[’ is always preceded by ’O[’…’]’ :
`’N[’⇒ ’O[’(C)’]’_`

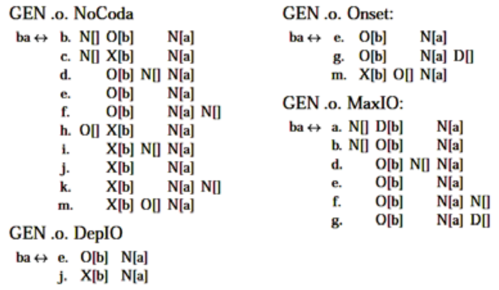
Faithfulness

MAX-IO (No deletion.) the language that does not contain ’X[’ :
`~$’X[’$’]’`

DEP-IO (No epenthesis.) the language in which ’O[’, ’N[’ and ’D[’ never have ’]’ immediately following :
`~${’O[’ ’]’, ’N[’ ’]’, ’D[’ ’]’}`

Remark: Each simple finite-state automaton can be interpreted as a transducer (with upper and lower side identical)

What happens if we compose *Gen* and a constraint?



GEN .o. NoCoda

ba ↔	b.	N[]	O[b]		N[a]	
	c.	N[]	X[b]		N[a]	
	d.		O[b]	N[]	N[a]	
	e.		O[b]		N[a]	
	f.		O[b]		N[a]	N[]
	h.	O[]	X[b]		N[a]	
	i.		X[b]	N[]	N[a]	
	j.		X[b]		N[a]	
	k.		X[b]		N[a]	N[]
	m.		X[b]	O[]	N[a]	

GEN .o. DepIO

ba ↔	e.	O[b]	N[a]
	j.	X[b]	N[a]

GEN .o. Onset:

ba ↔	e.	O[b]		N[a]	
	g.	O[b]		N[a]	D[]
	m.	X[b]	O[]	N[a]	

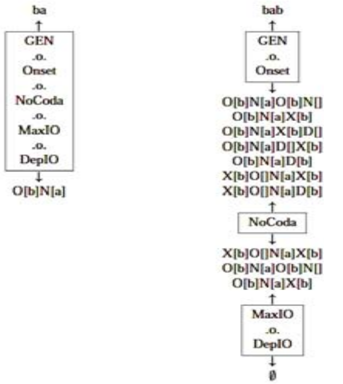
GEN .o. MaxIO:

ba ↔	a.	N[]	D[b]		N[a]	
	b.	N[]	O[b]		N[a]	
	d.		O[b]	N[]	N[a]	
	e.		O[b]		N[a]	
	f.		O[b]		N[a]	N[]
	g.		O[b]		N[a]	D[]

Preliminary conclusion
Composing *Gen* and a constraint has the effect that all candidates violating the constraint are filtered out

Question
Could we compose a cascade of all the constraints to implement an OT system?
(assumed ranking: ONSET **>>** NOCODA >> MAX-IO **>>** DEP-IO)

??



Preliminary conclusion

Composing *Gen* and a constraint has the effect that all candidates violating the constraint are filtered out

Question

Could we compose a cascade of all the constraints to implement an OT system?
(assumed ranking: ONSET >> NOCODA >> MAX-IO >> DEP-IO)

NO!

The problem is that this approach does not account for violability of constraints.

- Only perfect candidates will go through.
- Constraints should only be applied when at least one candidate satisfies them!

Priority Union

This operation was originally introduced as an operation for unifying two feature structures in a way that eliminates any risk of failure by stipulating that one of the two (the first one) has priority in case of a conflict:

$$Q = \left\{ \begin{matrix} a \\ | \\ x \end{matrix} , \begin{matrix} b \\ | \\ y \end{matrix} \right\} \qquad R = \left\{ \begin{matrix} b \\ | \\ z \end{matrix} , \begin{matrix} c \\ | \\ w \end{matrix} \right\}$$

$$Q\,.P.\,R = \left\{ \begin{matrix} a \\ | \\ x \end{matrix} , \begin{matrix} b \\ | \\ y \end{matrix} , \begin{matrix} c \\ | \\ w \end{matrix} \right\}$$

Lenient composition

Using *priority union* it is possible to define a special composition operation within the FST formalism that

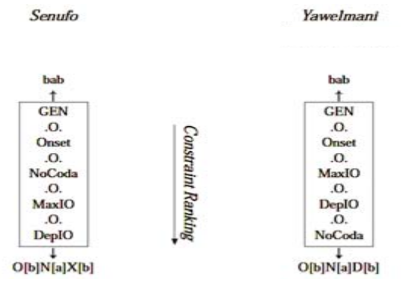
- applies a particular transducer as a filter if the resulting language is non-empty, but
- else ignores the transducer

```
R .O. C = [R .o. C] .P. R
```

Advantages

- the entire OT system is precompiled into a single transducer: a *lenient cascade*
- no runtime computation of candidates – very efficient
- compact FST: 66 (or 248) different states (Karttunen 1998 – slightly different system)

Examples of lenient cascades



.ba.<b>

.bab.

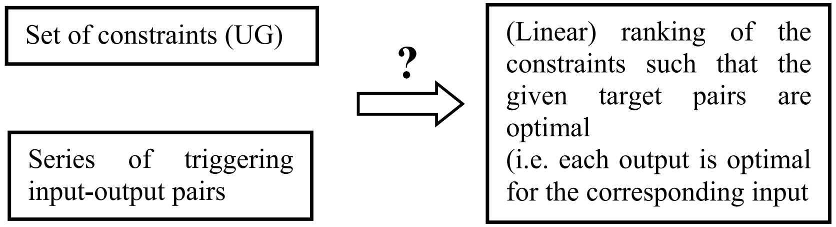
5 Bidirectionality

- Strong bidirectional optimization can be implemented based on the individual unidirectional cascades:
  - the regular languages representing the candidates after the application of the lenient cascades can be *intersected*
  - thus, strong bidirectional optimization can be expressed as a single FST
- This is not possible for weak bidirectionality (see Jäger 2000). The computational capacity of weak bidirectionality goes beyond what can be handled by FSTs.

Lecture 3: OT learning Theory

1. Extracting constraint rankings from given input-output pairs
2. Constraint demotion: Prerequisites and formal results
3. The comprehension/production dilemma in child language
4. The OT learning algorithm
5. Richness of the base and constraints on inventories

1 Extracting constraint rankings from given input-output pairs



Set of constraints (UG)

Series of triggering input-output pairs

?

(Linear) ranking of the constraints such that the given target pairs are optimal (i.e. each output is optimal for the corresponding input

This is a simplified basic picture only: It is …

- useful for manually constructing grammars from given pairs
- requiring a list of relevant (hidden) inputs
- unrealistic as a model of language learning (inputs are hidden units – we have no direct access to them!)

Example

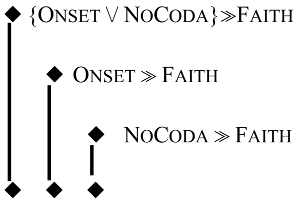
Consider the basic syllable theory of the previous lecture with the system of constraints: {FAITH , ONSET, NOCODA}.

Extract the right ranking from input-output pairs like:

/atat/ - .a.tat.	(*English*)
/atat/ - .a.ta.⟨t⟩	(*Hawaiian*)
/atat/ - .□a.tat.	(*Yawelmani*)
/atat/ - .□a.ta.⟨t⟩	(*Senufo*)

optimal *Senufo*

Input: /atat/	Faith	Onset	NoCoda
1 .a.tat. (*English*)		*	*
2 .a.ta.⟨t⟩ (*Hawaiian*)	*	*	
3 .□a.tat. (*Yawelmani*)	*		*
✔ 4 .□a.ta.⟨t⟩ (*Senufo*)	**		



A candidate w is considered to be optimal iff for each competitor w', the constraints that are lost by w must be ranked lower than at least one constraint lost by w'.

Learning is assumed to be triggered by (positive) input-output pairs (which should come out as *grammatical* with regard to the "learned" Grammar). Each pair brings with it a body of negative evidence in the form of competitors (provided by **Gen**). This fact has to be emphasized as one of the main advantages of a connectionist theory like OT.

Input: /atat/	Onset	NoCoda	Faith
1 .a.tat. (*English*)	*	*	
2 .a.ta.⟨t⟩ (*Hawaiian*)	*		*
3 .□a.tat. (*Yawelmani*)		*	*
✔ 4 ☞ .□a.ta.⟨t⟩ (*Senufo*)			**

Information about the ranking, collected from 4≻3, 4≻2, 4≻1:

{Onset, NoCoda} ≫ Faith

Constraint demotion

Given a certain input *I* and a target output *SD*. The input is paired with a competitor *SD'*. This constitutes a Winner-Loser Pair: *SD* ≻*SD'*.

For any constraint C which is lost by the winner SD, if C is not dominated by a constraint C' lost by the competitor SD', demote C to immediately below the highest constraint that is lost by SD'.

Example 1

Input: /atat/	Faith	Onset	NoCoda
1 .a.tat. (*English*)		*	*
2 .a.ta.⟨t⟩ (*Hawaiian*)	*	*	
3 .□a.tat. (*Yawelmani*)	*		*
✔ 4 .□a.ta.⟨t⟩ (*Senufo*)	**		

A sample run for *Senufo*:

start	{Onset, NoCoda, Faith}
4 ≻3:	{Onset, NoCoda} ≫ Faith
4 ≻2:	"
4 ≻1:	"

Example 2

Input: /atat/		Faith	Onset	NoCoda
1	.a.tat.		*	*
✔ 2	.a.ta.⟨t⟩	*	*	
3	.□a.tat.	*		*
4	.□a.ta.⟨t⟩	**		

A sample run for *Hawaiian*:

start:	{Onset, NoCoda, Faith}
2 $\succ$1:	{Onset, NoCoda} $\gg$ Faith
2 $\succ$3:	NoCoda $\gg$ {Faith, Onset}
2 $\succ$4:	NoCoda $\gg$ Faith $\gg$ Onset

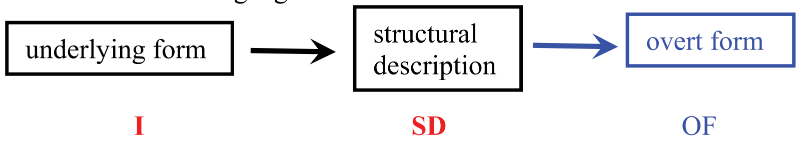
2 Constraint demotion: Prerequisites and formal results

(A) UG = *Gen* + *Con.* The learning problem consists in inferring the ranking of the constraints in *Gen.* This excludes both the possibility that the constraints themselves are learned (in part at least) or that aspects of the generator are learnable.

(B) The force of strict domination $\gg$: A relation of the form C $\gg$ C' does not merely mean that the cost of violating C is higher than that of violating C'; rather, it means that no number of C' violations is worth a single C violation. The force of strict domination excludes cumulative effects where many violations of lower ranked constraints may overpower higher ranked constraints.

(C) The OT grammar of the language that has to be learned is based on a *total* ranking of all the constraints: $C_1 \gg C_2 \gg \ldots \gg C_n$.
During learning the ranking of the constraints is not restricted to a total ranking. Instead, more general domination hierarchies are admitted which have the following general form:
$\{C_1, C_2, \ldots, C_3\} \gg \{C_4, C_5, \ldots, C_6\} \gg \ldots \gg \{C_7, C_8, \ldots, C_9\}$. ("stratified domination hierarchy")

(D) In the (theoretically) simplest case, learning is triggered by pairs <*I*, *SD*> consisting of a (hidden) input and a structural description *SD* of the source language *L*.



Fact 1: Correctness of iterative constraint demotion

The iterative procedure of constraint demotion converges to a set of totally ranked constraint hierarchies, each of them accounting for the learning data. Interestingly, this result holds when starting with an arbitrary constraint hierarchy. (cf. Tesar & Smolensky 2000)

Fact 2: Data complexity of constraint demotion

Consider a system with a fixed number of constraints, say N. The number of informative data pairs required for learning is no more than N(N-1)/2, independent on the initial hierarchy and the nature of the constraints. (cf. Tesar & Smolensky 2000)

Hint for proof: Crucial is the inherently comparative character of OT. Assuming N constraints, then for each pair $1 \leq i, j \leq N$ it has to be decided whether $C_i \gg C_j$ or $C_i \gg C_j$. There are exactly N(N-1)/2 such decisions and each one can be brought about on the basis of one appropriate data pair triggering the corresponding set of winner-loser pairs. Consequently, no more than N(N-1)/2 appropriate data pairs should be necessary for learning the correct grammar.

Comparison between OT and P&P

Let's assume a parameterized UG with n parameters. Then this system admits 2^n grammars when the parameters are binary. In the worst case, the average number of triggers before reaching the target grammar is 2^n. This is due to the fact that the learner is informed about the correct value of the different parameters by positive data only, and that all parameters are interacting in the worst case.

		Number of Grammars	Number of Triggers
P&P	30 binary parameters	2^{30} = 1,073 x 10^9	1,073 x 10^9
OT	20 constraints	20! = 2,43 x 10^{18}	190

3 The comprehension/production dilemma in child language

Children's linguistic ability in production lags dramatically behind their ability in comprehension.

Standard reaction of Generative Grammar: dramatically greater competence-performance gap for children. Typically: children do not produce a particular segment because their motor control hasn't yet mastered. However, Menn & Mattei (1992) show that children who systematically avoid a given structure in their linguistic production can often easily imitate it.

Jacobson's generalization

The same configurations which are marked in the sense of disfavoured in adult languages tend also to be avoided in child language.

Consequence: constraints defining linguistic markedness are shared across adult and child language production. It would be attractive to have a viable hypothesis according to which **Grammar** has a central role to play in explaining child production.

The two horns of the dilemma

(1) Competence-performance gap for children (empirically wrong)

(2) Two grammars for children, one for production, the other for comprehension (extremely unattractive)

OT provides a simple way out of this dilemma. The point is that the structures that compete are different in production and comprehension!

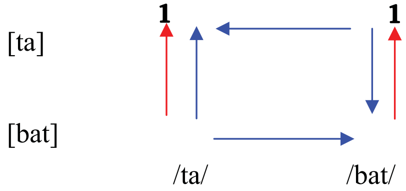
Demonstration of the basic idea

Assume /bat/ as a lexical input & consider two possible surface strings:

.bat. pronounced [bat]

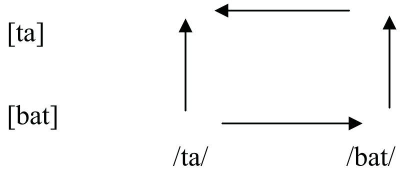
⟨ ba⟩.t$\boxed{a}$. pronounced [ta]

Take /ta/ as another lexical input. As constraints we take Faith and Structure, the latter standing for a complex of markedness constraints making [ta] more harmonic than [bat]. Importantly, the same OT Grammar can be uses both for production & comprehension:



Initial State

Structure(1) ≫ Faith



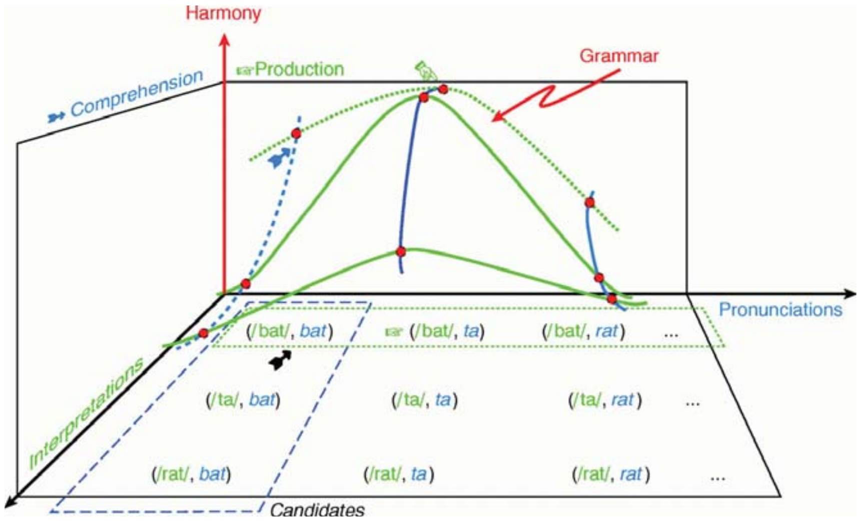
resulting harmonic order

Comprehension: [bat] ⇒ ? Solution /bat/

Production: /bat/ ⇒ ? Solution [ta] **Conflict: associate [bat] !**

In comprehension, [bat] is correctly associated with /bat/. This is a consequence of the fact that the structural (markedness) constraints are sensitive to the overt phonetic forms only. Consequently, Faith determines the correct association.

On the other hand, **in production** /bat/ is associated (wrongly) with the overt form [ta], which is the most unmarked form. This is a consequence of the fact that within the initial Grammar the faithfulness constraints are dominated by the markedness constraints.

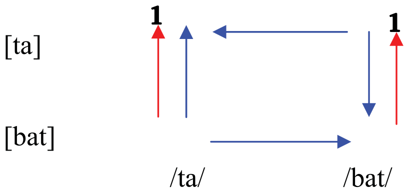


The same idea expressed in a 3D representation (from Prince & Smolensky 1997): The horizontal plane contains pairs such as < /bat/, *ta* > (representing a structure in which the lexical item /bat/ is simplified and pronounced *ta*). The vertical axis shows the relative harmony of each structure, an ordinal rather than a numerical scale. This harmony surface schematically depicts a young child's knowledge of grammar: STRUCTURE dominates FAITHFULNESS.

In **comprehension**, the pronunciation *bat* is given, and competition is between the column of structures containing *bat* (dashed box). Because these are all pronounced *bat*, they tie with respect to STRUCTURE, so lower-ranked FAITHFULNESS determines the maximum-harmony structure to be (/bat/, *bat*), marked with ➼ (peak of the dashed curve). The same grammar that gives correct comprehension results in incorrect—simplified—**production**: the row of structures containing /bat/ compete (dotted box); the maximum-harmony structure best-satisfies top-ranked STRUCTURE with the simplified pronunciation *ta* (peak of the dotted curve): this is marked ☞.

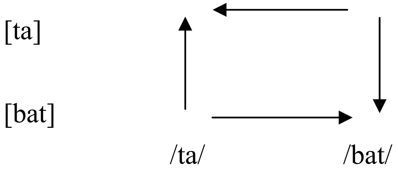
Triggering learning

According to Smolensky (1996b) the ‘conflict’ between comprehension and production is the trigger for learning, where learning is understood as a reranking of the involved constraints. In short, the relevant (disturbing) constraints are demoted. In our example, STRUCTURE is demoted:



After learning

FAITH ≫ STRUCTURE(1)
(reranked)



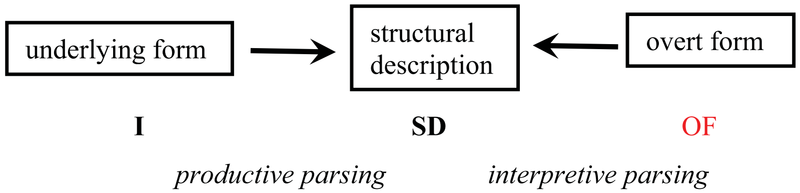
resulting
harmonic order

Comprehension: [bat] ⇒ ?   Solution /bat/
Production: /bat/ ⇒ ?   Solution [bat]

4 The OT learning algorithm

- The algorithm starts with an initial grammar: as above, FAITHfulness constraints are dominated by MARKedness constraints. (This initial ranking is a necessary precondition for a language to be learnable; cf. Smolensky (1996a) for the general argument)
- Comprehension mode: The algorithm proceeds by taking overt phonetic forms as primary data, and assign this data full structural descriptions (*robust interpretive parsing*).
- Production Mode: Determine the current Grammar’s output starting with the structural description assigned by the comprehension

mode. Since the grammar isn't yet complete this procedure normally doesn't lead back to the origin overt form.



underlying form → structural description ← overt form

I **SD** OF

productive parsing *interpretive parsing*

- Constraint Demotion: whenever the structural description which has just been assigned to the overt data (comprehension) is less harmonic than the current grammar's output (production), relevant constraints are demoted minimally to make the comprehension parse the more harmonic.

- This yields a new grammar, which the algorithm then uses to repeat the whole process over again, reassigning structural descriptions to the primary data and then reranking constraints accordingly. The cycle is iterated repeatedly.

- This kind of bootstrap algorithm transforms a bad grammar into a better one. It has been illustrated (simulation) that the algorithm in most cases allows efficient convergence to a correct grammar. (Supposed that the hierarchy of the target language has the property of *total ranking*)

- OT helps to translate structural insights from *Markedness Theory* into a concrete learning algorithm.

- Learning develops a stabilized OT Grammar that can be characterized by the feature of recoverability or bidirectional optimality

- For a critical evaluation of Smolensky & Tesar's (classical) OT learning theory see Hale & Reiss (1998), for an improved learning theory see Boersma & Hayes (2001) [in the reader].

- For a very simple example, see exercise 2. For a couple of more realistic examples and a careful discussion of how the learning algorithm can fail, see chapter 4 of Tesar & Smolensky's (2000) excellent book "Learnability in Optimality Theory". [see a review in the reader]

5 Richness of the base and constraints on inventories

- In standard Generative Grammar, the source of *cross-linguistic variation* is manifold. There are cross-linguistic differences in the input and output systems and in the (parameterised) principles on rules. Especially, the inputs are predominantly determined by language-specific lexical factors.

- OT is a very restrictive theory with regard to the source of variation. Essentially, the following is a fundamental principle of standard OT:

Richness of the Base

The source of all systematic cross-linguistic variation is constraint reranking. In particular, the set of inputs to the grammars of all languages is the same. The grammatical inventories of a language are the outputs which emerge from the grammar when it is fed the universal set of all possible inputs.

(This principle was proposed in Prince and Smolensky 1993:191)

- *Richness of the base* requires that systematic differences in inventories arise from different constraint rankings, not different inputs.
- *Richness of the base* is not a empirical principle but a methodological assumption (rejecting constraints on inputs).

Constraints on inventories

How to explain the different inventories in natural languages? According to OT, the content of lexical inputs is unconstrained. Whether some segment occurs on the surface in a particular language is determined strictly by the constraint grammar of the language in question.

If faithfulness to a particular feature outranks any prohibitions governing the appearance of the feature, then the feature contributes to defining a language's inventory. If prohibitions against some feature outrank relevant faithfulness constraints, then the feature does not play a role in the inventory

As an example, consider the case of voicing on obstruents. (Recall that obstruents refer to the class of oral stops and fricatives such as the voiceless series p, t, k, s, š and the voiced series b, d, g, z, ž)

- English, German, Dutch, ...: The feature VOICE is **contrastive** for obstruents, i.e. there are minimal pairs like pan/ban, tend/dent, kill/gill, sip/zip mean different things.
- Haiwaiian: The feature VOICE is **noncontrastive** for obstruents. In Haiwaiian, all obstruents (p, k) are voiceless. The voicelessness is redundant.

There is a tendency for obstruents to be voiceless. It derives from the phonetic fact that it is more difficult to maintain vibration of the vocal cords when there is a constriction of the type that produces a fricative or an oral stop.

Phonological markedness constraint

Obstruents must be voiceless: OBS/*VOICE

First case: VOICE a contrastive feature. Voiced obstruents are attested to the inventory if lexical voicing contrasts override this markedness constraint, i.e.

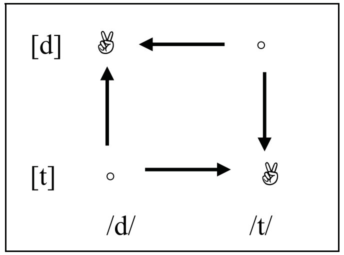
FAITH[VOICE] ≫ OBS/*VOICE.

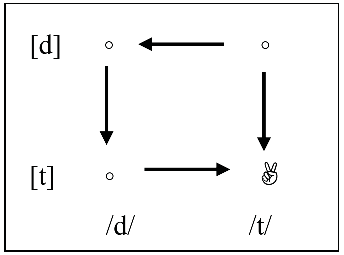
Second case: VOICE as a non-contrastive feature. Voiced obstruents are excluded from the inventory if the markedness constraint overrides faithfulness:

OBS/*VOICE ≫ **FAITH[VOICE]**.

		FAITH[VOICE]	OBS/*VOICE		FAITH[VOICE]	OBS/*VOICE
[d]	☞		*		*	*
[t]		*		☞		
	/d/			/t/		

		OBS/*VOICE	FAITH[VOICE]		OBS/*VOICE	FAITH[VOICE]
[d]		*			*	*
[t]	☞		*	☞		
	/d/			/t/		





[Note: Joan Bresnan (1997, 2001) applies the basic ideas developed in phonology and gives a principled account for a *typology of pronominal systems* and the emergence of the unmarked pronoun.]

Neutralization

In Russian and Dutch voicing is contrastive on obstruents. That is, as in English, the voicing distinction on obstruents leads to differences in lexical meaning (e.g. [bɛ.dən], [bɛ.tən] in Dutch). Unlike English, Russian and Dutch does not maintain the voicing contrast in all positions. Specifically, the distinction between voiced and voiceless obstruents is lost at the end of a syllable where all obstruents appear as voiceless.

As shown earlier, neutralization can described as an extension of the system

FAITH[VOICE] ≫ OBS/*VOICE

by adding the constraint CODA/*VOICE which overrides the other constraints:

CODA/*VOICE ≫ FAITH[VOICE] ≫ OBS/*VOICE

Allophony

As a final example of constraint interaction, a feature may be noncontrastive, but with a distinction nevertheless arising in a predictable context, a case of allophony. In this case, typically a constraint on assimilation overrides the constraints in

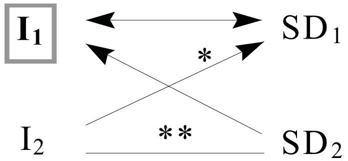
OBS/*VOICE ≫ FAITH[VOICE]

(See the example in the exercise part).

Lexicon optimization

The idea is that whenever the learner has no evidence (from surface forms) to postulate a specific divergent lexical form, she will assume that the input is identical to the surface form. Notice that this approach to the analysis of inputs is based on the assumption of full specification and is opposing to the idea of *underspecification* with regard to the inputs.

Lexicon optimization means recoverability of the inputs from the outputs. It invites to introduce a bidirectional mode of optimization.



Only recoverable inputs are assumed to be realized in the mental lexicon.

Lexicon Optimization

Examine the constraint violations incurred by the winning output candidate corresponding to each competing input. The input-output pair which incurs the fewest violations is considered the optimal pair, thereby identifying an input from the output.

(This principle was introduced in Prince and Smolensky (1993) and developed in Itô, Mester & Padgett (1995))

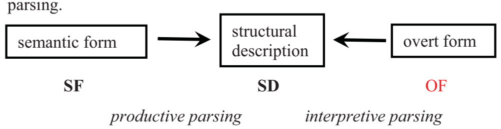
Lecture 4: OT Syntax

1. The nature of input in OT syntax
2. The generated outputs
3. Constraint inventory
4. Do-support
5. General discussion
6. Interpretive Parsing and how OT may overcome the competence-performance gap
7. Garden-path effects
8. Perception strategies and OT

0. Introduction: Core Ideas

- OT is not a theory of phonology proper but rather a theory of Grammar (and perhaps several other cognitive domains: semantics, vision, music.)
- The OT idea of robust (interpretive) parsing: competent speakers can often construct interpretations of utterances they simultaneously judge to be ungrammatical (notoriously difficult to explain within rule- or principle-based models of language)

- The presence of interpretable but ungrammatical sentences corresponds to mismatches between interpretive and productive parsing.


semantic form → structural description ← overt form

SF **SD** OF

productive parsing *interpretive parsing*

- The first part of this lecture outlines Grimshaw's OT account to grammaticality (including a factorial typology). This theory is founded on productive optimization.

- The second part explains interpretive parsing and introduces a constraint theory of processing. *Garden-path effects* of processing are predicted if optimal (interpretive) parses (corresponding to some early input) cannot be extended. This demonstrates that the principles of grammar have psychological reality for mature linguistic systems.

1 The nature of input in OT syntax

Following Grimshaw (1997), syntactic inputs are defined in terms of lexical heads and their argument structure:

INPUT

- lexical head plus its argument structure
- an assignment of lexical heads to its arguments
- a specification of the associated tense and semantically meaningful auxiliaries.

For convenience, we call such inputs *Predicate-Argument Structures* or simply *Logical Forms.*

Examples

- *What did Peter write?*
  {*write*(x,y), x=*Peter*, y=*what*, tense=past}
- *What will Peter write?*
  {*write*(x,y), x=*Peter*, y=*what*, tense=future, auxiliary=*will*}

Note that no semantically empty auxiliaries (*do*, *did*) are present in the input.

For treating embeddings more elaborated LFs are necessary (e.g. Legendre et al. 1998):

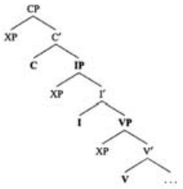
- *You wonder who eat what*
  wonder (*you*, Q_i Q_j *eat*(t_i , t_j))
  Q_i *wonder* (*you*, Q_j *eat*(t_i , t_j))

2 The GENerated Outputs

Minimal X' Theory

Each node must be a good projection of a lower node, if a lower one is present.

(X' Theory does not require that some *head* must be present in every projection!)



Extended Projection

An extended projection is a unit consisting of a lexical head and its projection plus all the functional projections erected over the lexical projection. The smallest verbal projection is VP, but IP and CP are both extended projections of V.

Example (continued)

[$_{VP}$ [$_{V'}$ [$_{V}$ write][$_{NP}$ what]]],
[$_{IP}$ [$_{NP}$ Peter] [$_{I'}$ [$_{I}$ _] [$_{VP}$ [$_{V'}$ [$_{V}$ write][$_{NP}$ what]]]
[$_{CP}$ [$_{XP}$ _] [$_{C'}$ [$_{C}$ _] [$_{IP}$ [$_{NP}$ Peter] [$_{I'}$ [$_{I}$ _] [$_{VP}$ [$_{V'}$ [$_{V}$ write][$_{NP}$ what]]]

are all extended projections of [$_{V}$ write] (conform to further lexical specifications given in the input)

The GENerator (informal definition)

The core of GEN will construct all extended projections conform to the lexical specifications in the input. A further restriction is that no element be literally removed from the input (‘containment’). The core can be extended by the following operations:

- introducing functional heads as they do not appear in the input, due to their lack of semantic content (e.g. the complementizer *that* and do-support in English
- introducing empty elements (traces, etc.), as well as their coindexations with other elements
- moving lexical elements.

Example (continued)

Input: {*write*(x,y), x=*Peter*, y=*what*, tense= past}

Some **Gen**erated outputs (using a simplified notation):

1. [$_{IP}$ Peter [$_{VP}$ wrote what]] *...Chinese*
2. [$_{CP}$ what [$_{IP}$ Peter [$_{VP}$ wrote ***t***]]] *...Czech, Polish*
3. [$_{CP}$ what wrote$_i$ [$_{IP}$ Peter [$_{VP}$ e$_i$ ***t***]]] *...Dutch, German*
4. [$_{CP}$ what did$_i$ [$_{IP}$ Peter e$_i$ [$_{VP}$ write ***t***]]] *...English*
5. [$_{CP}$ what [$_{IP}$ Peter did [$_{VP}$ write ***t***]]] *...??*

Invalid outputs are

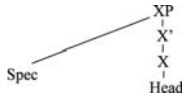
[$_{VP}$ wrote what]

[$_{IP}$ Peter [$_{VP}$ wrote _]]

[$_{CP}$ what [$_{IP}$ Peter [$_{VP}$ wrote what]]]

3 The constraint inventory

Markedness Constraints



- **Operator in Specifier** (Op-Spec)
  Syntactic operators must be in specifier position
- **Obligatory Heads** (Ob-Hd)
  A projection has a head
- **Case Filter** (Case)
  The Case of a Noun Phrase must be checked

Faithfulness Constraints

- **Economy of Movement** (Stay)
  Trace is not allowed
- **No Movement of a Lexical Head** (No-Lex-Mvt)
  A lexical head cannot move
- **Full Interpretation** (Full-Int)
  Lexical conceptual structure is parsed
  (this kind of Faith bans semantically empty auxiliaries)

Op-Spec: triggers wh-movement wh$_i$ ...t$_i$

Ob-Hd: triggers head-movement Aux$_i$ ... e$_i$

4 Do-Support

The auxiliary *do* is possible only when it is necessary' (Chomsky 1957)

Fact 1

Do is obligatory in simple interrogative sentences.

*What did Peter write? - *What Peter wrote?*

Fact 2

Do cannot occur with other auxiliary verbs in interrogatives.

*What will Peter write? - *What does Peter will write - *What will Peter do write?*

Fact 3

Do-support is impossible in positive declarative sentences.

*Peter wrote much - *Peter did write much*

Fact 4

The occurrence of auxiliary *do* is impossible in declarative sentences that already contain another auxiliary verbs, such as *will.*

*Peter will write much - *Peter will do write much - *Peter does will write much*

Fact 5

Auxiliary *do* cannot co-occur with itself, even in interrogatives.

*What did Peter write? - *What did Peter do write?*

The Analysis

- The auxiliary *do* is a semantically empty verb, one which only serves the *syntactic* function of head of extended projections.
- Do-support is triggered by the markedness constraint Ob-Hd at the expense of violations of the faithfulness constraint Full-Int .

  Ob-Hd >> Full-Int
- The facts of subject-auxiliary inversion in English suggest a ranking

  Op-Spec, Ob-Hd >> Stay (see Exercice 2)
- Merging the two rankings

  Op-Spec, Ob-Hd >> Full-Int, Stay

  For English, the two markedness constraints outrank the general constraints (Faithfulness, Economy of Movement)

Example (concerning fact 1)

Input: {*write*(x,y), x=*Peter*, y=*what*, tense= past}		Op-Spec	Ob-Hd	Full-Int	Stay
1	[$_{\text{IP}}$ Peter [$_{\text{VP}}$ wrote what]]	*	*		
2	[$_{\text{CP}}$ what [$_{\text{IP}}$ Peter [$_{\text{VP}}$ wrote ***t***]]]		**		*
3	[$_{\text{CP}}$ what wrote$_i$ [$_{\text{IP}}$ Peter [$_{\text{VP}}$ e$_i$ ***t***]]]		*		**
4 ☞	[$_{\text{CP}}$ what did$_i$ [$_{\text{IP}}$ Peter e$_i$ [$_{\text{VP}}$ write ***t***]]]			*	**
5	[$_{\text{CP}}$ what [$_{\text{IP}}$ Peter did [$_{\text{VP}}$ write ***t***]]]		*	*	*

Fact 2 & 4: auxiliary=*will* in the input; same constraints & rankings.

Fact 3: Full Interpretation!

Fact 5: you have to assume that Full-Int dominates Stay.

Typological consequences

In order to simplify discussion, the reranking approach to language typology (‘factorial typology’) will applied here to a very small set of syntactic constraints: {Op-Spec, Ob-Hd , Stay}

- Op-Spec, Ob-Hd >> Stay

  Both wh-movement and inversion occur in violation of Stay, to satisfy both top ranking constraints (example: *English*)
- Stay >> Op-Spec, Ob-Hd

  Violations of Stay are avoided at the expanse of violations of ‘well formedness’. A grammar arises lacking Wh-movement as well as inversion. (example: *Chinese*)
- Ob-Hd >> Stay >> Op-Spec

  same picture as before

- OP-SPEC >> STAY >> OB-HD
  Wh-movement is forced but inversion cannot be used to fill the head position. A grammar arises that has Wh-movement but not inversion (example: *French*)
- Languages like *German* and *Dutch* require to consider the constraint NO-LEX-MVT (*No Movement of a Lexical Head*) which was undominated so far.
  Assuming NO-LEX-MVT to be outranked by the other constraints, structures like [$_{CP}$ Was schrieb$_i$ [$_{IP}$ Peter [$_{VP}$ e$_i$ ***t***]]] are optimal now (such languages are always incompatible with a semantically empty auxiliary).

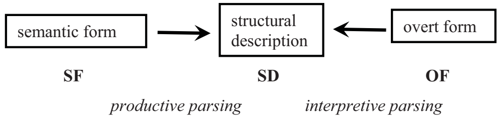
5 General discussion

- Bresnan (1998; see the reader) gives an important reformulation and improvement of Grimshaw (1995/1997; see the reader).
  - based on a mathematically sound structural account (feature structures in LFG)
  - adopts more radically non-derivational theory of *Gen*, based on a parallel correspondence theory of syntactic structures
  - conceptual and empirical advantages

- The problem of (language-particular) *ineffability*: There are input structures than can be realized in some languages but not others. For example, the questions "who ate what" is realizable in English and German, not in Italian. Such a question must be generable by *Gen* since it is realized in some language, and *Gen* is universal. Both in English and in Italian there is a non-empty candidate set. Consequently, in both cases there should exist an optimal output (a grammatical forms that expresses the question). But in Italian there is no grammatical form that means "who ate what". (cf. Legendre, Smolensky & Wilson 1998)

6 Interpretive Parsing and how OT may overcome the competence-performance gap

- Human sentence parsing is an area in which optimality has always been assumed. According to the nature of (interpretive) parsing, in this case the comprehension perspective comes in: the parser optimises underlying structures with respect to overt form.



semantic form → structural description ← overt form

SF **SD** **OF**

productive parsing *interpretive parsing*

- Do the heuristic parsing strategies (assumed in the psycholinguistic literature) reflect the influence of the principles of grammar?
- Widespread and incorrect conviction that the impossibility of identifying the parser with the grammar had already been established with the failure of the 'Derivational Theory of Complexity' (e.g. Fodor, Bever, & Garrett 1974)
- Parsing preferences can be derived from the principles of UG if the proper grammatical theory is selected. There is evidence that in OT *the same system of constraints* is crucial for both productive parsing (OT syntax proper) and interpretive parsing. This finding is a first important step in overcoming the competence-performance gap. (See Fanselow et al. 1999)

7 Garden-path effects

Readers or listeners can be misled or 'quoted up the garden path' by locally ambiguous sentences

Example 1

- The boat floated down the river sank / and sank
- Bill knew John liked Maria / who liked Maria

Example 2

- While the cannibals ate missionaries drunk / they sang
- Since Jay always jogs a mile seems like a short distance / this seems like a short distance to him.

Garden-path model (Frazier 1979)

The parsing mechanism aims to structure sentences at the earliest opportunity, to minimise the load on working memory. In more detail:

- only one syntactical structure is initially considered for any sentence (ignoring prosody)
- meaning is not involved at all in the selection of the initial syntactical structure (*modular processing architecture*)
- the simplest syntactical structure is chosen (minimal attachment and late closure)
  - minimal attachment: the grammatical structure producing the fewest nodes or units is preferred
  - late closure: new words encountered in a sentence are attached to the current phrase or clause if this is grammatically permissible

8 Perception strategies and OT

Gibson & Broihier (1998) give a straightforward account how to implement the garden path model in OT. Following Frazier & Clifton (1996) a PSG is assumed in which there are no vacuous projections (generating, for example, [$_{\text{NP}}$ John] but not [$_{\text{NP}}$ [$_{\text{N'}}$ [$_{\text{N}}$ John]]]).

Inputs

Sequences of lexical items such as (*the, boat*) and (*the, boat, floated*).

Generated Outputs

The inputs are parsed into well-formed phrase structures (according to the rules of PSG). The actual output has to extend outputs of earlier inputs (in order to minimize the load on working memory)

(*the*)	→	output $_1$
(*the, boat*)	→	(output $_1$ + something) $_2$
(*the, boat, floated*)	→	(output $_2$ + something) $_3$

Constraints

- NodeConservativity (correlate of Minimal Attachment)
  Don't create a phrase structure node
- NodeLocality (correlate of Late Closure)
  Attach inside the most local maximal projection
- NodeConservativity >> NodeLocality

Garden-path effects are predicted if optimal parses (corresponding to some early input) cannot be extended.

Example 1 (continued) {*node conservativity* crucial}

1. (*the*)

   [$_{\text{NP}}$ [$_{\text{DET}}$ the]]

(*Assuming the parser is top-down to some degree*)

2. (*the, boat*)

   [$_{\text{IP}}$ [$_{\text{NP}}$ [$_{\text{DET}}$ the] [$_{\text{N}}$ boat]]

3. (*the, boat, floated*)

   a. [$_{\text{IP}}$ [$_{\text{NP}}$ [$_{\text{DET}}$ the] [$_{\text{N}}$ boat]] [$_{\text{VP}}$ floated]]

   1 new node (VP) / 1 locality violation (NP)

   b. [$_{\text{IP}}$ [$_{\text{NP}}$ [$_{\text{DET}}$ the] [$_{\text{N'}}$ [$_{\text{N}}$ boat] [$_{\text{CP}}$ [$_{\text{IP}}$ [$_{\text{VP}}$ floated]]]]]]

   4 new nodes (VP, IP, CP, N') / 0 locality violations

Example 2 (continued) {locality crucial}

1. (*While, the, cannibals, ate*)

   [$_{\text{IP}}$ [$_{\text{CP}}$ [$_{\text{C}}$ while] [$_{\text{IP}}$ [$_{\text{NP}}$ the cannibals]] [$_{\text{VP}}$ ate]]]]

2. (*While, the, cannibals, ate, missionaries*)

   a. [$_{\text{IP}}$ [$_{\text{CP}}$ [$_{\text{C}}$ while] [$_{\text{IP}}$ [$_{\text{NP}}$ the cannibals]] [$_{\text{VP}}$ [$_{\text{V}}$ ate] [$_{\text{NP}}$ missis]]]]]

   2 new nodes (V, NP) / 0 locality violations

   b. [$_{\text{IP}}$ [$_{\text{CP}}$ [$_{\text{C}}$ while] [$_{\text{IP}}$ [$_{\text{NP}}$ the cannibals]] [$_{\text{VP}}$ ate]]]
   [$_{\text{IP}}$ [$_{\text{NP}}$ missis]]]]

   2 new nodes (IP, NP) / 3 locality violations (VP, IP, CP)

9 The constraint theory of processing (CTP)

The psychological reality of Grammar

Position A: Parser ≠ Grammar	Position B: Parser = Grammar
• early generativists • peoples shocked by the failure of the derivational theory of complexity (DTC)	• students following the DTC • some people believing in OT syntax (e.g. Pritchett 1992, Fanselow et al. 1999)

"Precompiled rules or templates are used in parsing" (Frazier & Clifton 1996). Such templates can be seen as a kind of procedural knowledge that gives an efficient, but rather indirect (non-transparent) realization of the grammar	"If correct, this view argues against the necessity of specific assumption for design features of the parser - optimally, we need not assume much more than that the grammar is embedded into our cognitive system." (Fanselow et al. 1999)
The psychological reality of grammatical principles is then at best confined to the role they play in *language acquisition*.	The principles of grammar have psychological reality for mature linguistic systems as well.

The basic idea of the **CTP** is that there is no difference between the constraints Grammars use and the constraints parsers use. "We may postulate that the parser's preferences reflect its attempt to maximally satisfy the grammatical principles in the incremental left-to-right analysis of a sentence." (Fanselow et al. 1999: 3).

The following analyzes have an illustrating character only. We freely use abbreviations, e.g. *the boat* instead of *[$_{NP}$ [$_{DET}$ the] [$_{N}$ boat]]*. The symbols Comp, Infl indicate empty heads (with respect to CP and IP, respectively). OP$_{i}$ indicates an empty operator.

Example 1 (again)

1. (*the, boat*)

   [$_{IP}$ the boat [$_{I'}$ Infl ...]

   1 violation of OB-HD)

(*Assuming the parser is top-down to some degree*)

2. (*the, boat, floated*)

   a. [$_{IP}$ the boat [$_{I'}$ Infl [$_{VP}$ floated ...]

   1 violation of OB-HD)

   b. [$_{IP}$ the [$_{N'}$[$_{N}$boat] [$_{CP}$ OP$_{i}$ Comp [$_{IP}$ t$_{i}$ Infl [$_{VP}$ floated t$_{i}$]]]]] [$_{I'}$ Infl...]

   Many violations of OB-HD and STAY

Comments

- The first step illustrates *overparsing.* Postulating the IP-node and an (empty) Infl-Element we create a category that is able to check a case (satisfying CASE). The overparsing procedure can be seen as a way of finding a local optimum and is one of the key factors responsible for parsing preferences.
- In the second step there are two possibilities. Clearly, the option corresponding to "early closure" is preferred when evaluating the violations of the *grammatical* constraints.

Example 2 (again)

1. (*While, the, cannibals, ate*)

   [$_{IP}$ [$_{CP}$ while Comp] [$_{IP}$ the cannibals [$_{I'}$ Infl [$_{VP}$ ate ...]

2. (*While, the, cannibals, ate, missionaries*)

   a. [$_{IP}$ [$_{CP}$ while Comp] [$_{IP}$ the cannibals [$_{I'}$ Infl [$_{VP}$ ate missis ...]

   No new violations

   b. [$_{IP}$ [$_{CP}$ while Comp] [$_{IP}$ the cannibals [$_{I'}$ Infl [$_{VP}$ ate]]]]

   [$_{IP}$ missis [$_{I'}$ Infl [$_{VP}$ ...]]]

   New violations of Ob-Hd etc.

Conclusions

The constraint theory of processing looks promising and is an opportunity to realize syntax as an psychological reality not only in the realm of language acquisition but also that of language comprehension. It is advantageous both for theoretical and empirical reasons

However, there are several questions:

- The precise foundation of overparsing.
- Are the constraints appropriate to derive *all* parsing preferences?
- The garden path effects are very different in strength. How to account for such differences in terms of OT?
- Extensions are required: the influence of world knowledge and prosody.

Lecture 5: OT Semantics/Pragmatics

The Aim

Bringing together:

- The tradition of *Radical Pragmatics*
- The view of *Optimality Theory*

Advantages

For *Radical Pragmatics*	For *Optimality Theory*
Improved analyses	New applications
theoretical stringency	Motivating the constraints
the emergence of iconicity	New ideas about grammaticalization and language change

Outline

1. Meaning and Interpretation
2. Blocking and global theories of language
3. Literalism vs. contextualism
4. Optimality Systems
5. The Motivation for Strong Bidirectionality
6. Weak Bidirectionality and Constructional Iconicity
7. Example: Negative Strengthening

1 Meaning and Interpretation

The observation: Linguistically encoded information doesn’t fully specify the truth conditions of a sentence.

- Katz & Fodor (1963): A full account of sentence interpretation has to include more information than that of syntactic structure and lexical meaning.

  a. *Should we take the lion back to the zoo?*
  b. *Should we take the bus back to the zoo?*

- Psycholinguistics: Mental models, situation structure,...
  The tones sounded impure because the hem was torn.

The tones sounded impure because the hem was torn.

Theoretical Models

- Kaplan’s distinction between *character* and *intension*

  intension = character(*c*)

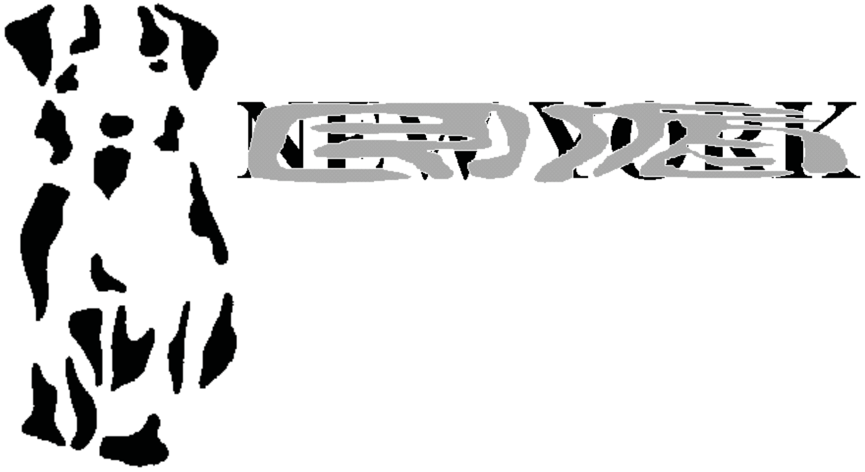
- **Radical Underspecification View**

  Underspecified representations + contextual enrichment (Hobbs 1983, Alshawi 1990, Poesio 1991, Pinkal 1995, etc.)

  => Find optimal enrichments!

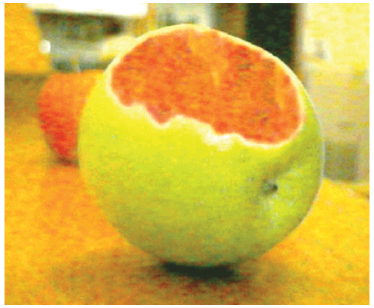
SCOTTISH PIPER

Example: Pattern underspecification and completion



Linguistic example: Attributive modification

-	*a red apple*	[red peel]
-	*a sweet apple*	[sweet pulp]
-	*a reddish grapefruit*	[reddish pulp]
-	*a white room/ a white house*	[inside/outside]



A red apple?

What color is an apple?

Q_1 What color is its peel?
Q_2 What color is its pulp?

Other examples from lexical pragmatics

– John ate breakfast [this morning; in the normal way]	free enrichment
– Every boy [in the class] is seated	domain restriction
– Peter began a novel [to read/ to write]	Pustejovsky
– I'm parking outside [my car]	deferred inference
– Max is tall [for a fifth grader]	comparison class
– What color is a red nose, red flag, red bean?	Herb Clark
– This apple is red [on the outside]	

2 Blocking and global theories of language

Local Theories

The (grammatical) status of a (linguistic) object LO is decided exclusively considering properties of LO, and the properties of other linguistic objects LO' are completely irrelevant for this decision.

Examples: Traditional Generative Linguistics, Model Theoretic Semantics.

Global Theories (Competition-based)

There are different linguistic objects in competition. The winner of the competition suppresses the other competing candidates, ruling them out from the set of well-formed linguistic objects.

Examples: Early Structuralism (Saussure), Field Theories, Prototype Theories, Optimality Theory, Connectionism.

Blocking

PLURAL	▨		☞	☞	☞	☞	☞
DUAL	▨	☞	▨	▨	▨	▨	▨
	①	②	③	④	⑤	⑥	…

The value of a German or Latin **plural** is not the value of a Sanskrit plural. But the meaning, if you like, is the same. In Sanskrit, there is the dual. Anyone who assigns the same value to the Sanskrit plural as to the Latin plural is mistaken because I cannot use the Sanskrit plural in all the cases where I use the Latin plural.

If you take on the other hand a simple lexical fact, any word such as, I suppose, **mouton** (French) may have the same meaning as **sheep** in English. However, it doesn't have the same value. For if you speak of the animal on the hoof and not on the table, you say sheep. It is the presence in the language of a second term (mutton) that limits the value attributable to sheep.

Notes taken by a student of Saussure's lectures [4 July 1911]

3 Literalism vs. contextualism

The Gricean picture: Literalism

- Using the meanings of the words plus the syntactic structure of the sentence, a minimal proposition for capturing the literal meaning of the sentence can be determined
- Context-dependencies of literal meaning can only arise from indexical expressions.
- No semantic underdetermination is involved, no unarticulated constituents*.
- Pragmatic mechanism of contextual strengthening (Conversational implicature)

* This term refers to the idea of explaining the near equivalence of sentences such as 'it is raining' and 'it is raining here' by assuming an unarticulated constituent of place in the first sentence. It is a *constituent*, because there is no truth-evaluable proposition unless a place is supplied (since rain occurs at a time in a place). It is *unarticulated*, because there is no morpheme that designates that place (Perry 2003)

Conversational Implicatures: Some Standard Examples

(Q1) Some of the boys are at the party
=> Not all of the boys are at the party
(*Scalar implicatures*, Gazdar 1979)

(Q2) Rick is a philosopher or a poet
=> Rick is not both a philosopher and a poet
(*Scalar implicatures*, Grice 1968; Atlas and Levinson 1981)

(Q3) Rick is a philosopher or a poet
=> Rick may (not) be a philosopher; Rick may (not) be a poet
(*Clausal implicatures*, Gazdar 1979; Atlas and Levinson 1981)

(I1) If you mow the lawn, I'll give you $5
=> If and only if you move the lawn, will I give you $5
(*Conditional perfection*, Geis & Zwicky, 1971)

(I2) John unpacked the picnic. The beer was warm.
=> The beer was part of the picnic.
(*Bridging*, Clark & Haviland, 1977)

(I3) John said 'Hello' to the secretary and then he smiled
=> John said 'Hello' to the female secretary and then he smiled
(*Inference to stereotype*, Atlas & Levinson 1981)

The neo-/post-Gricean picture: Contextualism

- **Basic ideas**
  - Using the meanings of the words plus the syntactic structure of the sentence, it is not possible to calculate the literal meaning of the sentence. Some kind of underdetermined representation can be computed only.
  - Semantic underdetermination and the existence of unarticulated constituents are postulated.
  - The mechanism of pragmatic interpretation is crucial both for determining what the speaker says and what she means.

- **Explicature**: what the speaker says. Truth-conditional pragmatics
- **Implicature**: what the speaker means (conversational implicature in the narrower sense)

- **Variants of contextualism**
  - Neo-Gricean theories (Horn, Atlas)
  - Relevance theory (Sperber, Wilson, Carston)
  - Presumptive meanings (Levinson 2000)
  - OT pragmatics

Levinson's typology of implicatures

- The Q-heuristics: (*For the relevant salient alternates*) *What isn't said is not the case.*
  - Scalar implicatures
    some of the boys came => not all of the boys came
  - Clausal implicatures
    If John comes, I'll go => maybe he will, may be he won't

- The I-heuristics: *What is expressed simply is stereotypically exemplified*
  - *kill => stereotypical interpretation*
  - Conditional perfection (*B*, *if A => B iff A*)
  - Bridging inferences
  - **Negative strengthening**
  - The effect of "neg-raising"

- The M-heuristics: *What is said in an abnormal way isn't normal*
  - **Pragmatic effects of double negatives**
  - Periphrastic alternatives to simple causatives

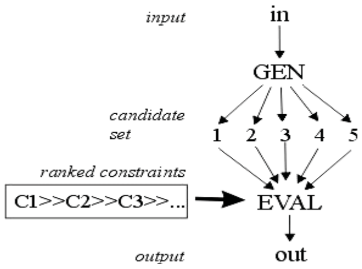
Remark: Levinson tries to turn this heuristic classification scheme into a general theory by stipulating a ranking Q > M > I. We accept the classification schema but not the theory. (Instead, we consider M as an *epiphenomenon* that results from the interaction of Zipf's two "economy principles").

Neo-Gricean theories and optimization (Atlas & Levinson, Horn)

I-principle (termed R by Horn)	Q-principle
Quantity 2, Relation	Quantity 1
Say no more than you must (given Q) (Horn 1984)	*Say as much as you can (given I)* (Horn 1984).
Read as much into an utterance as is consistent with what you know about the world (bearing the Q-principle in mind). [Levinson 1983: 146f.]	*Do not provide a statement that is informationally weaker than your knowledge of the world allows, unless providing a stronger statement would contravene the I-principle* [Levinson 1987: 401]
Conditional perfection, *neg-raising*, bridging	Scalar implicatures
Seeks to select the most *harmonic* interpretation	Can be considered as a blocking mechanism
Interpretive Optimization	**Expressive Optimization**

4 Optimality Systems

Basics



- An optimality system O is an triple $\langle \mathbf{GEN}, C, >> \rangle$ where

- **GEN** is a relation	Universal basis
- C is a set of functions from **GEN** to $\underline{\underline{N}}$	universal constraints
- >> is a linear ordering on C	language-particular ranking

- The ranking >> of the constraints constitutes a well-founded preference relation $<_O$ between pairs π of **GEN** (read $<_O$ as *less costly* or *more harmonic*):
  $\pi <_O \pi'$ iff there is a $c \in C$ such that $c(\pi) < c(\pi')$ and for all $c' >> c$: $c'(\pi) = c(\pi)$

Definition (unidirectional and bi-directional optimality)

Let $O = \langle \mathbf{GEN}, C, >> \rangle$ be an OT-system. Assume that GEN reflects the direction of interpretation; for example with $\langle a, b\rangle \in$ **GEN** assume that a is a syntactic form and b a semantic form.

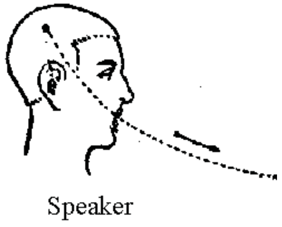
- A pair $\langle a, b\rangle$ is called *Hearer optimal* w.r.t. O iff

  (i) $\langle a, b\rangle \in$ **GEN**
  (ii) there is no b' such that $\langle a, b'\rangle \in$ **GEN** and $\langle a, b'\rangle <_O \langle a, b\rangle$

- A pair $\langle a, b\rangle$ is called *Speaker optimal* w.r.t. O iff

  (i) $\langle a, b\rangle \in$ **GEN**
  (ii) there is no a' such that $\langle a', b\rangle \in$ **GEN** and $\langle a, b'\rangle <_O \langle a, b\rangle$

- A pair $\langle a, b\rangle$ is called (*strongly*) *optimal* w.r.t. O iff it is both Speaker and Hearer optimal.



Phonology, Morphology:
Prince & Smolensky (1989);
McCarthy & Prince (1993); …

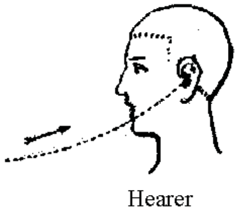
Syntax: Grimshaw (1997);
Bresnan (1999); …

Optimal Generation

Semantics: de Hoop & de Swart (1999) ; de Hoop & Hendriks (2001)

E.g. Domain Restrictions:

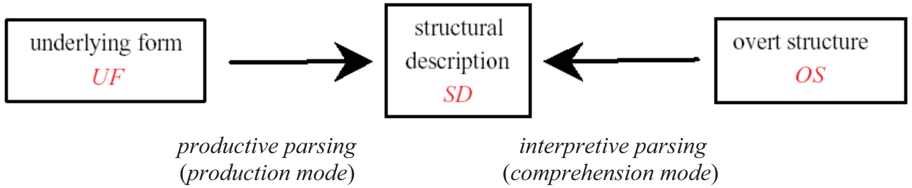
- *Most linguists sleep at night*
- *Most linguists drink at night*



Optimal Interpretation

5 The Motivation for Strong Bidirectionality

In overcoming the lag between production and comprehension, a kind of bootstrap mechanism seems to apply that makes crucially use of the *robustness* of comprehension, an issue that is substantial for the OT learning theory (Smolensky 1996, Tesar & Smolensky 2000).



underlying form *UF*

structural description *SD*

overt structure *OS*

productive parsing (*production mode*)

interpretive parsing (*comprehension mode*)

The discrepancy between interpretive parsing and productive parsing triggers learning.

- After learning, the two modes of assigning structure to inputs, productive and interpretive parsing, coincide.

Symmetry

The proposed theory of learning leads to the stabilization of SYMMETRY:

If in comprehension, some overt form OS leads to an underlying form UF, then in the generation mode, the same UF leads back to the original OS. As a consequence, all hearer-optimal pairs are strongly optimal!

This seems to hold for two kinds of learning:

(A) Auto-associative learning
(extracting structure from the input pattern)
e.g. Tesar & Smolensky (2000).

(B) Pattern association (learning the relation between two sets of independent stimuli)

Pattern association

- A set of pairs of patterns are repeatedly presented. The system is to learn that when one member of the pair is presented it is supposed to produce the other. In this paradigm one seeks a mechanism in which an essentially arbitrary set of input patterns can be paired with an arbitrary set of output patterns.
- For example, input patterns can be lexigrams (e.g. senseless syllables), and output patterns can be pictures of fruits. Assume a 1-1 correspondence between syllables and pictures.
- If subjects are qualified to match Stimulus A to B and then, without further training, match B to A, they have passed a test of symmetry.
- Children as young as 2 years pass the symmetry test! (Green 1990). Hence, bidirectionality seems to build in the basic learning mechanism.

Again, the result is SYMMETRY: If a => b then b => a, and vice versa. As a consequence, all hearer-optimal pairs are strongly optimal! The same for Speaker-optimal pairs.

Kanzi - a Monobo Monkey

Sue Savage-Rumbaugh was trying to teach Kanzi's mom, Matata, a symbolic language.

Kanzi sat on her lap during these sessions. And while Matata did poorly, Kanzi learned.

Kanzi's knowledge was reciprocal. There was no need taught her separately to produce and to comprehend.

6 Weak Bidirectionality and Iconicity

Blocking is not always total. Classical examples are as follows:

- Morphological blocking
  - furious - *furiosity – fury
  - fallacious - *fallacity – fallacy

- Blocking of interpretations
  - *I ate pork/?pig*
  - *Some persons are forbidden to eat beef/?cow*
  - *The table is made of wood/?tree*
  - *I see/?smell what you mean*

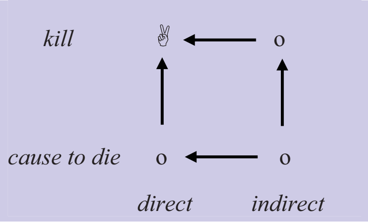
Example: strong bidirectionality and total blocking

- **GEN =** { ⟨*kill*, *direct*⟩, ⟨*kill*, *indirect*⟩ , ⟨*cause to die*, *direct*⟩, ⟨*cause to die*, *indirect*⟩ }
  (Semantics with underdetermination)

- Markedness constraints for forms and interpretations
  - ⟨*kill*, *int*⟩ < ⟨*cause to die*, *int*⟩ (since *kill* is the lighter form)
  - ⟨*form*, *direct*⟩ < ⟨*form*, *indirect*⟩ (since *direct* is the more salient interpretation)

- McCawley's pair:

  Bill killed the Sheriff
  Bill caused the Sheriff to die



- The solution concept of *strong optimality* accounts for total blocking.
  It does not account for partial blocking! Look for other solution concepts!!

Weak bidirectionality (super-optimality)

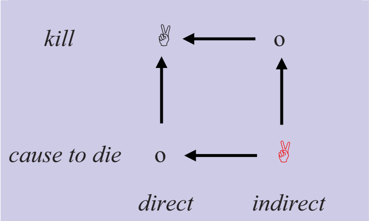
There is a conception of bidirectional optimization, called super-optimality, which can account for constructional iconicity. This conception makes use of recursion.

Let Ω = ⟨**GEN**, *C*, >>⟩ be an OT-system. Then a pair ⟨*a*, *b*⟩ is super-optimal w.r.t. Ω iff

(1) ⟨*a*, *b*⟩ ∈ **GEN**
(2) there is no super-optimal ⟨*a*, *b*'⟩ < ⟨*a*, *b*⟩
(3) there is no super-optimal ⟨*a*', *b*⟩ < ⟨*a*, *b*⟩

John McCawley's example again:

Bill killed the Sheriff
Bill caused the Sheriff to die



Krifka's example: How much precision is enough?

Krifka's Observation

- Vague interpretations of measure expressions are preferred if they are short Precise interpretations of measure expressions are preferred if they are long

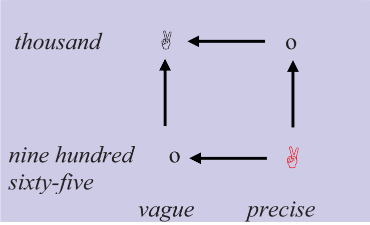
A: The distance between Amsterdam and Vienna is one thousand kilometers.
B: #No, you're wrong; it's nine hundred sixty-five kilometers.

A: The distance between A and V is nine hundred seventy-two kilometers.
B: No, you're wrong; it's nine hundred sixty-five kilometers.

Street sign in Kloten, Switzerland.

Explanation

- Markedness constraints for forms and interpretations
  - ⟨*form*, *int*⟩ < ⟨*form*', *int*⟩ iff *form* is lighter than *form*'
  - ⟨*form*, *int*⟩ < ⟨*form*, *int*'⟩ iff *int* is less precise than *int*'
- Weak Bidirection



➔ Generalization: Constructional Iconicity in Natural Language

Constructional Iconicity (or Horn's division of pragmatic labor)

Unmarked forms tend to be used for unmarked situations and marked forms for marked situations. (Levinson's M-principle)

- MAYERTHALER ZICK ZACK

- BERLIN & KAY MOLA ● ● MILI
- ARGUMENT LINKING (Uszkoreit, Bresnan, Jackendoff, Kiparski, …)

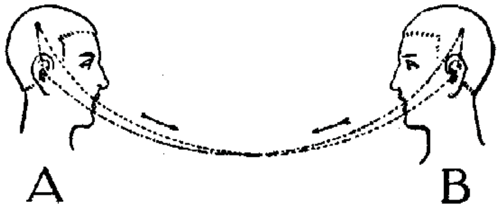
  Agent > Instrument > Recipient/Experiencer > Theme > Location
  Subject > Object_d > Object_i > Oblique

Harmonic alignment

Economy and Language

(I) Economy plays a crucial role in online interpretation and production (e.g. in explaining garden path effects). (Standard OT, Levinson)

(II) Economy constitutes *languages* as conventional systems. (Horn, Zipf)



Georg K. Zipf (1949)

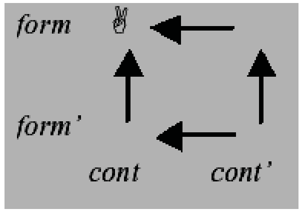
Human Behavior and the Principle of Least Effort. Addison-Wesley. Cambridge 1949.

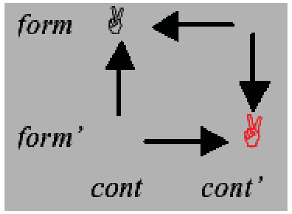
Two basic and competing forces	
Speaker's economy	Hearer's economy
Force of unification	Force of diversification

- The two opposing economies are evolutionary forces
- They are balanced during language evolution.

Why two conceptions of Bidirectionality?

- Strong Optimality as a synchronic law (describing an equilibrium that results from successful learning)
- Weak (Super-) Optimality as a diachronic law (describing the probable outcomes of language evolution under highly idealized conditions)



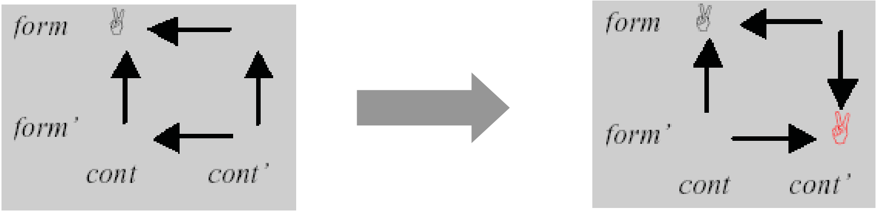


Calculating super-optimal solutions

Jäger (1999), Dekker & van Rooy (1999), Beaver (2002) have proposed procedures that update preferences in OT systems such that

(i) optimal pairs are preserved

(ii) a new optimal pair is produced if and only if the same pair was super-optimal at earlier stages.



The evolutionary grounding of weak bidirection

There are many different ways to realize a evolutionary perspective. Different versions highlight the role of *correlations*, *learning*, *mutations*, and the *initial state*, respectively.

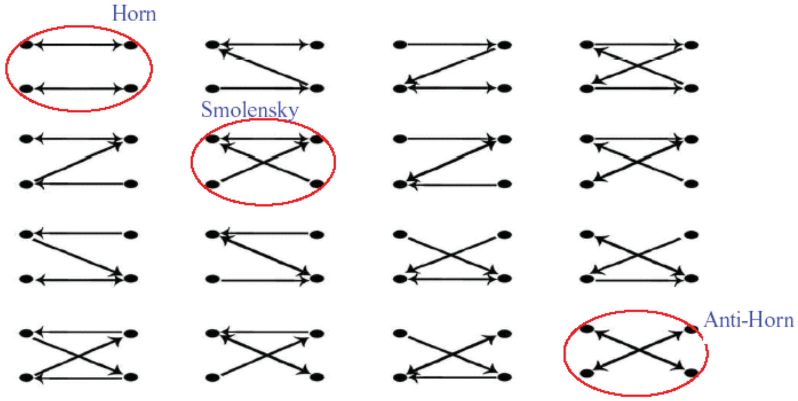
- Van Rooy (2002): *Signalling games and evolutionary stable Horn-strategies.*
- Jäger (2002): *Learning constraint sub-hierarchies. The Bidirectional Gradual Learning Algorithm.*
- Blutner, Borra, Lentz, Obdeijn, Uijlings, and Zevenhuijzen (2002): *Signalling Games: hoe evolutie optimale strategieën selecteert.*

Basic Ideas

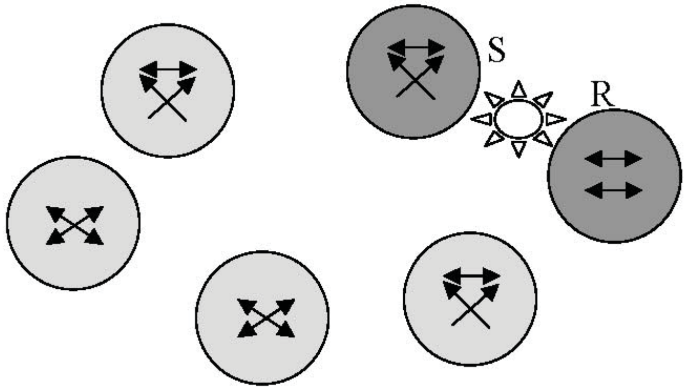
- Each agent is described by an OT-system $O = \langle \mathbf{GEN}, C, >> \rangle$. Within the population Gen and C are fixed, >> may vary.
  Each agent X determines a *speaker's strategy* S_X : Contents => Forms
  and a *hearer's strategy* H_X : Forms => Contents
- In pairwise interactions between an agent *a* (in the role of the speaker) and an agent *b* (in the role of the Hearer) an utility/fittness function U is realized:
  $U(a,b) = \Sigma\, P(i)\, [\, \delta(H_b(S_a(i)), i) - k(S_a(i))]$,
  where $\delta(x,y) = 1$ if $x = y$, 0 elswhere. P(i) probability of "content" i, k(f) cost of signal f.
- The agents of the population randomly encounter one another in pairwise interaction. Each organism plays only one, but leaves its offspring behind, where the number of offspring is determined by the utility value U(a, b). Mutations change the strategies played by some elements of the population. After many plays of the game, a strategy yielding a higher number of expected offspring will gradually come to be used by larger and larger fractions of the population.

The pool of possible strategies

for an OT-system with GEN = {⟨*f*, *c*⟩, ⟨*f*, *c*'⟩, ⟨*f*', *c*⟩, ⟨*f*', *c*'⟩}

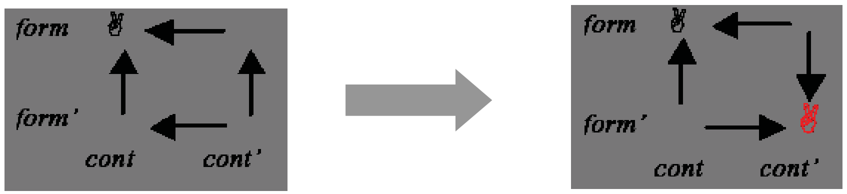


Population and pairwise interaction



Results

- Horn and Anti-Horn are the only strategies (OT-systems) that are evolutionary stable
- Starting with a uniform *Smolensky* population will always result in a pure *Horn* population supposed P(*c*) > P(*c* ') and k(*f*) < k(*f'*)



- Mixed populations develop into pure Horn populations (supposed P(*c*) > P(*c* ') and k(*f*) < k(*f'*))

7 Example : Negative strengthening

What are the effects of **negating** gradable adjectives?

(1) I'm not happy ☺ 😐 ☹
  a. *Entailment:* It isn't the case that I'm happy
  b. *Implicature*: I'm unhappy
  c. *defeasibility*: I'm not happy and not unhappy

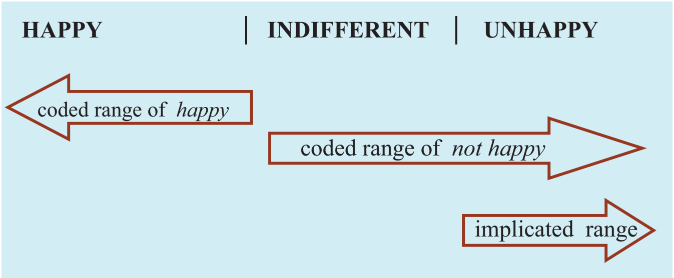


Fig.1 Contradictories implicating contraries

The described effect of strengthening is restricted to the positive (unmarked) elements of antonym pairs!

Litotes

(2) I'm not unhappy ☺ 😐 ☹

a. *Entailment:* It isn't the case that I'm unhappy

b. *Implicature*: I'm rather happy (but not quite as happy as using the expression *"happy"* would suggest)

c. *defeasibility*: I'm not unhappy, in fact I'm happy

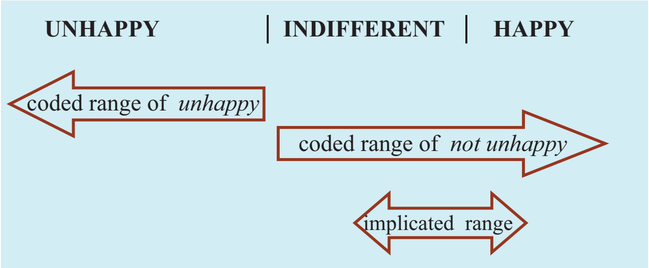


Figure 2: Litotes: when two negatives don't make a positive

Theoretical Assumptions

- The coded range of form-interpretation pairs is due to a three-valued logic: *not* corresponds to weak negation and *un-* to strong negation.

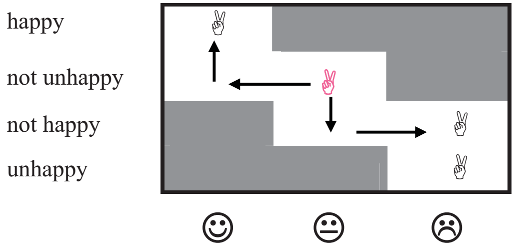
- The number of the involved negation morphemes determine the markedness of the forms

$$\langle form, int\rangle < \langle form', int\rangle \text{ iff}$$
form contains less negation morphemes than *form*'

- The markedness of interpretations decreases towards the ends of the scale (and is maximum in the "neutral" middle)

$$\langle form, int\rangle < \langle form, int'\rangle \text{ iff}$$
int is closer to the end of the scale than *int'*

Super-optimal pairs



Lecture 6: Logical foundations

1 Introduction: different formal approaches

2 Penalty logic

3 Penalty logic and Bayesian networks

4 Penalty logic and Dempster-Shafer theory

5 Penalty logic and neural nets

6 Learning

1 Introduction: different formal approaches

- Brewka (1994); Besnard, Mercer & Schaub (2002) [for a copy go to http://www.cs.uni-potsdam.de/wv/pdfformat/bemesc02a.pdf]: Optimality Theory through Default Logic with priorities. The priorities are handled by a total ordering defined on the system of defaults. See also Nicolas Rescher's (1964) book "Hypothetical reasoning" which clearly expresses the very same idea.
- Dick de Jongh & Fenrong Liu (2006). They take an approach in terms of priority sequences of logical expressions, an idea that comes close to Brewka (1994).
- Pinkas (1992) introduced penalty logic and used it to model high-level (logical) properties of neural networks (see also Pinkas, 1995)
- Lima et al. (Lima, Morveli-Espinoza, & Franca, 2007) improve on it.
- Prince (2002) and Pater et al. (2007; 2007) compare OT hierarchies and systems with weighted constraints.

2 Penalty logic

The presentations follows Darwiche & Marquis (2004) and Blutner (2004). Let's consider the language $\mathcal{L}_{At}$ of propositional logic (referring to the alphabet At of atomic symbols).

Definition 1: A triple <At, Δ, k> is called a *penalty knowledge base* (PK) iff (i) Δ is a set of consistent sentences built on the basis of At (the possible hypotheses); (ii) k: $\Delta \Rightarrow (0, \infty)$ (the penalty function).

Intuitively, the penalty of an expression δ represents what we should pay in order to get rid of δ. If we pay the requested price we no longer have to satisfy δ. Hence, the larger k(δ) is, the more important δ is.

From some PK we can extract the system $W = \{[\alpha, k(\alpha)]: \alpha\in\Delta\}$ which is called the *weighted base* of the system PK (see Darwiche & Marquis)

Definition 2: Let α be a formula of our propositional language $\mathcal{L}_{At}$. A *scenario of* α *in PK*(*W*) is a subset Δ' of Δ such that $\Delta'\cup\{\alpha\}$ is consistent. The cost $K_{PK}(\Delta')$ of a scenario Δ' in PK is the sum of the penalties of the formulas of PK that are not in Δ':

$$K_{PK}(\Delta') = \sum_{\delta\in(\Delta-\Delta')} k(\delta)$$

Definition 3: An *optimal scenario of* α *in PK* is a scenario the cost of which is not exceeded by any other scenario (of α in PK), so it is a penalty minimizing scenario. With regard to a penalty knowledge base PK, the following cumulative consequence relation can be defined:

$\alpha \mid\!\sim_{PK} \beta$ iff β is an ordinary consequence of
each optimal scenario of α in PK.

Hence, penalties may be used as a criterion for selecting preferred consistent subsets in an inconsistent knowledge base, thus inducing a non-monotonic inference relation.

Example 1

Weighted base W: $\{\langle a\wedge b, 2\rangle, \langle\neg b, 1\rangle\}$

Optimal scenario for a in W:
$\Delta_1 = \{a\wedge b\}$ $K_{PK}(\Delta_1) = 1$

Optimal scenario for ¬a in W: (violating a∧b or b, respectively)
$\Delta_2 = \{\neg b\}$ $K_{PK}(\Delta_2) = 2$

$a \mid\!\sim_{W} b$

$\neg a \mid\!\sim_{W} \neg b$

Example 2

First Law: A robot may not injure a human being.
Second Law: A robot must follow (obey) the orders given it by human beings, except where such orders would conflict with the First Law.
Third Law: A robot must protect its own existence, as long as such protection does not conflict with the First or Second Law.

Weighted base W

$\neg I$	5	(first law)
F	2	(second law)
P	1	(third law)
$(S \wedge F) \rightarrow K$	1000	(S: giving the order to kill her)
$K \rightarrow I$	1000	(K: the robot kills her)

Two scenarios for S in W (violating F and ¬I, respectively)

$\Delta_1 = \{\neg I, P, (S \wedge F) \rightarrow K, K \rightarrow I\}$ $K_{PK}(\Delta_1) = 2$

$\Delta_2 = \{F, P, (S \wedge F) \rightarrow K, K \rightarrow I\}$ $K_{PK}(\Delta_2) = 5$

$\boxed{S \mid\!\sim_W \neg I}$

Semantics

Consider a *penalty knowledge base* PK = <At, Δ, k>. Let ν denote an ordinary (total) interpretation for the language $\mathcal{L}_{At}$ (ν: At→{0,1}). The usual clauses apply for the evaluation $[\![.]\!]_\nu$ of the formulas of $\mathcal{L}_{At}$ relative to ν. The following function indicates how strongly an interpretation ν conflicts with the space of hypotheses Δ of a penalty knowledge base PK:

Definition 4 (system energy of an interpretation)

$\mathcal{E}_{PK}(\nu) =_{def} \sum_{\delta\in\Delta} k(\delta)\ [\![\neg\delta]\!]_\nu$

$\mathcal{E}_{PK}(\nu)$ is also called *violation rank* (Pinkas), *cost* (deSaint-Cyr et al.), *weight* (Darwiche & Marquis) of the interpretation.

Example 1 again

Weighted base W: $\{\langle a\wedge b, 2\rangle, \langle\neg b, 1\rangle\}$.
Let us consider the following four interpretations over the variables appearing in *W*, Var(*W*):

- $\nu 1 = (a, b)$ $\mathcal{E}_{PK}(\nu 1) = 1$
- $\nu 2 = (a, \neg b)$ $\mathcal{E}_{PK}(\nu 2) = 2$
- $\nu 3 = (\neg a, b)$ $\mathcal{E}_{PK}(\nu 3) = 3$
- $\nu 4 = (\neg a, \neg b)$ $\mathcal{E}_{PK}(\nu 4) = 2$

Hence, the interpretation with minimum energy is ν1.

Preferred models

Let α be a wff of the language $\mathcal{L}_{At.}$ An interpretation ν is called a *model* of α just in case $[\![\alpha]\!]_\nu = 1$.

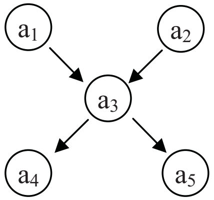
Definition 4

A *preferred model* of α is a model of α with minimal energy $\mathcal{E}$ (with regard to the other models of α). As the semantic counterpart to the syntactic notion $\alpha \mid\!\sim_{PK} \beta$ given in Definition 3 we can define the following relation:
$\alpha \mid\!\approx_{PK} \beta$ iff each preferred model of α is a model of β.

As a matter of fact, the syntactic notion (Definition 3) and the present semantic notion (21) coincide. Hence, the logic is sound and complete. A proof can be found in Pinkas (1995).

Example 1, continued: $a \mid\!\approx b$; $\neg a \mid\!\approx \neg b$.

3 Penalty logic and Bayesian networks

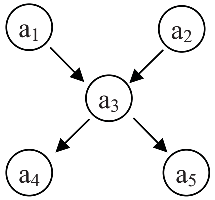


Consider a Bayesian network with binary random variables $a_1, a_2, \ldots, a_n$. Consider a partial specification of these random variables described by a set of "interpretations" V. Let α be a conjunction of literals (atoms or their negation) that describes this set V, i.e. $V = \{\nu\colon \nu(\alpha) = 1\}$.

Finding a most probable world model: find the specification of the random variables that maximizes the probability $\mu(\nu)$ of the joint distribution; in other words, find argmax $_{\nu \in V}[\mu(\nu)]$.

Example: $\alpha = a_1 \wedge \neg a_2$, find an optimal specification of the random variables $\{a_3, a_4, a_5\}$ maximizing the joint probability $\mu(a_1 = 1, a_2 = 0, a_3 = 0/1, a_4 = 0/1, a_5 = 0/1)$. Of course, the concrete solution depends on the details of the conditioned probability tables.

Global semantics and finding a most probable world model (Kooij, 2006)



$$\mu(a_1, \ldots, a_n) = \prod_{i=1}^{n} \mu(a_i \,/\, \text{Parents}(a_i))$$

In the example:

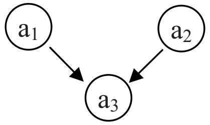
$\mu(a_1, \ldots, a_5) = \mu(a_1) \cdot \mu(a_2) \cdot \mu(a_3/a_1,a_2) \cdot \mu(a_4/a_3) \cdot \mu(a_5/a_3)$

$$\begin{aligned}
&\operatorname{argmax}_{\nu\in V}\ \mu(a_1 = \nu(a_1), \ldots, a_n = \nu(a_n)) \\
&\quad = \operatorname{argmax}_{\nu\in V}\ \mu(\nu) \\
&\quad = \operatorname{argmin}_{\nu\in V}\ -\log \mu(\nu) \\
&\quad = \operatorname{argmin}_{\nu\in V}\ \sum_{i=1}^{n} -\log \mu(a_i = \nu(a_i) \,/\, \text{Parents}(a_i) = \nu(\ldots))
\end{aligned}$$

The log-terms will be interpreted as penalties of corresponding rules:

$$\langle (\wedge_{x\in Parents(a_i)} x = \nu(x)) \rightarrow a_i = \nu(a_i)\, ,\ -\log \mu(a_i = -\nu(a_i) \,/\, \text{Parents}(a_i) = \nu(\ldots))\rangle$$

Example



Consider the weighted rules connected with the a_3-part of the CPTs:

a_1	a_2	$\mu(a_3 = T / a_1, a_2)$	weighted rule for $a_3 = T$
F	F	0.8	$\langle \neg a_1 \wedge \neg a_2 \rightarrow a_3, -\log 0.2\rangle$
F	T	0.4	$\langle \neg a_1 \wedge a_2 \rightarrow a_3, -\log 0.6\rangle$
T	F	0.5	$\langle a_1 \wedge \neg a_2 \rightarrow a_3, -\log 0.5\rangle$
T	T	0.3	$\langle a_1 \wedge a_2 \rightarrow a_3, -\log 0.7\rangle$

a_1	a_2	$\mu(a_3 = F / a_1, a_2)$	weighted rule for $a_3 = F$
F	F	0.2	$\langle \neg a_1 \wedge \neg a_2 \rightarrow \neg a_3, -\log 0.8\rangle$
F	T	0.6	$\langle \neg a_1 \wedge a_2 \rightarrow \neg a_3, -\log 0.4\rangle$
T	F	0.5	$\langle a_1 \wedge \neg a_2 \rightarrow \neg a_3, -\log 0.5\rangle$
T	T	0.7	$\langle a_1 \wedge a_2 \rightarrow \neg a_3, -\log 0.3\rangle$

The mapping theorem

Assume a Bayesian network is mapped into a penalty knowledge base in the indicated way. Then finding a most probable world model of a conjunction of literals α and finding a *preferred model* (minimal energy) of α with regard to the penalty knowledge base are equivalent tasks (leading to the same optimal interpretation)

Comment

Looking for preferred models in penalty logic can be interpreted as a kind of qualitative reasoning in Bayesian networks. Which values of a set of random variables give a maximal probability for a given specification α of a proper subset of these random variables? The concrete probability value for the specification α doesn't matter. What counts is the optimality of the assignment.

4 Penalty logic and Dempster-Shafer theory

Dempster-Shafer theory is a theory of *evidence*. There are different pieces φ_i of evidence that give rise to a certain belief function and a (dual) plausibility function. Different pieces of evidence can be combined by means of Dempster’s rule of combination.

A standard application is in medical diagnostics where some positive test result X can give a positive evidence for some disease Y but a negative test result gives absolutely no evidence for or against the disease.

Definition (mass function)

A mass function on a domain Ω of possible worlds (for a given piece of information) is a function m: $2^W \to [0, 1]$ such that the following two conditions hold:

$$\mathrm{m}(\emptyset) = 0.$$

$$\Sigma_{V\subseteq\Omega}\, \mathrm{m}(V) = 1$$

Definition (belief/plausibility function based on m)

Let m be a mass function on Ω. Then for every $U \subseteq \Omega$:

$$\mathrm{Bel}(U) =_{\mathrm{def}} \Sigma_{V \subseteq U}\, \mathrm{m}(V)$$
$$\mathrm{Pl}(U) =_{\mathrm{def}} \Sigma_{V\cap U\neq\emptyset}\, \mathrm{m}(V)$$

Dempster’s rule of combination

Suppose m_1 and m_2 are basic mass functions over *W*. Then $\mathrm{m}_1 \oplus \mathrm{m}_2$ is given by Dempster’s combination rule without renormalization:

$$\mathrm{m}_1 \oplus \mathrm{m}_2\, (U) = \Sigma_{V\mathrm{i}\cap V\mathrm{j}=U}\, \mathrm{m}_1(V_\mathrm{i}) \cdot \mathrm{m}_2(V_\mathrm{j})$$

Facts:

Assume $\mathrm{m}(U) = \oplus_{i=1}^{n} m_i\,(U)$; Pl plausibility function based on m; Pl_i plausibility function based on m_i. Then we have:

1. $\mathrm{Pl}(\{\nu\}) = \sum_{V \atop \nu\in V} \mathrm{m}(V)$; $\mathrm{Pl}_\mathrm{i}(\{\nu\}) = \sum_{V \atop \nu\in V} \mathrm{m}_\mathrm{i}(V)$
2. $\mathrm{Pl}(\{\nu\}) = \prod_{i=1}^{n} \mathrm{Pl}_\mathrm{i}(\{\nu\})$ [“contour function”]

W

Relating penalties to Dempster-Shafer theory

Let be $W = \{[\alpha_\mathrm{i}, \mathrm{k}(\alpha_\mathrm{i})]: \alpha_\mathrm{i}\in\Delta\}$ a *weighted base* of a system PK in our language $\mathcal{L}_{\mathrm{At}}$.
Each formula α_i represents a piece of evidence for $V_\mathrm{i} = \{\nu: \nu \models \alpha_\mathrm{i}\}$. Formally, this is expressed by the following mass function m_i:

$$\mathrm{m}_\mathrm{i}(V_\mathrm{i}) = 1 - \mathrm{e}^{-\mathrm{k}(\alpha\mathrm{i})}\ ;\ \mathrm{m}_\mathrm{i}(\Omega) = \mathrm{e}^{-\mathrm{k}(\alpha\mathrm{i})}$$

Using facts 1 and 2 it can be shown that[1]

$$\mathrm{Pl}(\{\nu\}) = \mathrm{e}^{-\mathcal{E}_{\mathrm{PK}}(\nu)}$$

This brings to light a relation between penalties and evidence where each formula of the knowledge base is considered to be given by a distinct source, this source having a certain probability to be faulty, and all sources being independent.

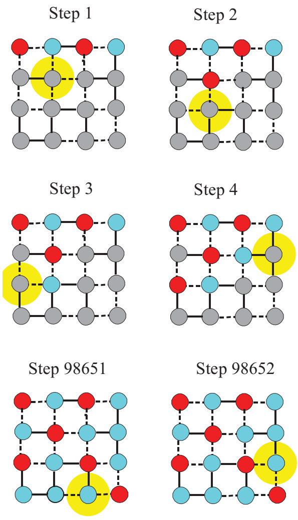
[1] For a proof see deSaint-Cyr, Lang, & Schiex (1994).

5 Penalty logic and neural nets

Main thesis: Certain activities of connectionist networks can be interpreted as nonmonotonic inferences. In particular, there is a strict correspondence between Hopfield networks and penalty/reward nonmonotonic inferential systems. There is a direct mapping between the information stored in such (localist) neural networks and penalty/reward knowledge bases.

- Certain logical systems are singled out by giving them a "deeper justification".
- Understanding Optimality Theory: Which assumptions have a deeper foundation and which ones are pure stipulations?
- New methods for performing nonmonotonic inferences: Connectionist methods (simulated annealing etc.)

Hopfield network - fast dynamics



Let the interval [-1,+1] be the *working range* of each neuron

+1: maximal firing rate
0: resting
-1 : minimal firing rate)

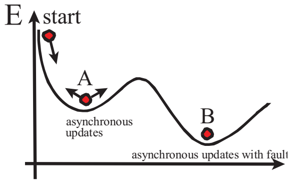
$S = [-1, 1]^n$
$w_{ij} = w_{ji}$, $w_{ii} = 0$

ASYNCHRONOUS UPDATING:

$$s_i(t+1) = \begin{cases} \theta\left(\Sigma_j w_{ij} \cdot s_j(t)\right), & \text{if } i = \text{rand}(1,n) \\ s_i(t), & \text{otherwise} \end{cases}$$

Summarizing the main results

Theorem 1 (Cohen & Großberg 1983)
Hopfield networks are resonance systems (i.e. $\lim_{n\to\infty} f^n(s)$ exists and is a resonance for each $s \in S$ and $f \in F$)

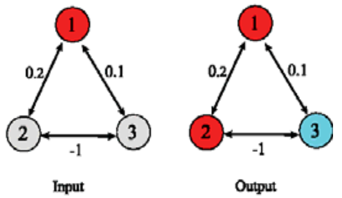


Theorem 2 (Hopfield 1982)
$E(s) = -\frac{1}{2} \sum_{i,j} w_{ij} s_i s_j$ is a *Ljapunov-function* of the system in the case of asynchronous updates. The output states $\lim_{n\to\infty} f^n(s)$ can be characterized as *the local minima* of E

Theorem 3 (Hopfield 1982)
The output states $\lim_{n\to\infty} f^n(s)$ can be characterized as *the global minima* of E if certain stochastic update functions f are considered (faults!).

Example

$$w = \begin{pmatrix} 0 & 0.2 & 0.1 \\ 0.2 & 0 & -1 \\ 0.1 & -1 & 0 \end{pmatrix}$$

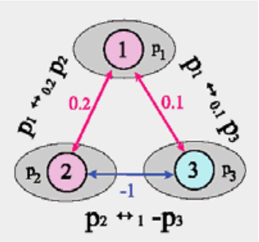


$E(s) = -0.2s_1s_2 - 0.1s_1s_3 + s_2s_3$

			E
<1 0 0>	≤	<1 0 0>	0
		<1 0 1>	-0.1
		<1 1 0>	-0.2
		<1 1 1>	0.7
		<1 1-1>	-1.1 ☜

$\text{ASUP}_w(<1\ 0\ 0>) = \min_E(s) = <1\ 1\text{-}1>$

The correspondence between symmetric networks and penalty knowledge bases



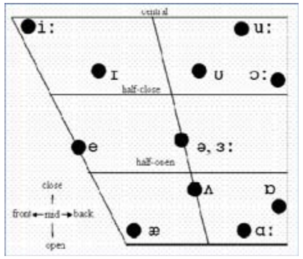
1. relate the nodes of the networks to atomic symbols a_i of $\mathcal{L}_{At.}$ At = $\{p_1,p_2,p_3\}$
2. translate the network in a corresponding weighted base $W = \{\langle p_1\leftrightarrow p_2, 0.2\rangle, \langle p_1\leftrightarrow p_3, 0.1\rangle, \langle p_2\leftrightarrow\neg p_3, 1\rangle\}$
3. relate states and interpretations: $s \cong \nu$ iff $s_i = \nu(a_i)$
4. observe that the energy of a network state is equivalent to the energy of an interpretation: $E(s) = \mathcal{E}_{PK}(\nu) =_{def} \sum_{\delta\in\Delta} k(\delta)\, [\![\neg\delta]\!]_\nu$ E.g.:

$E(<1\ 1\ 1>) = 0.7 \qquad = -0.2-0.1+1$

$E(<1\ 1 -1>) = -1.1 \qquad = -0.2+0.1-1$

…

Example from phonology

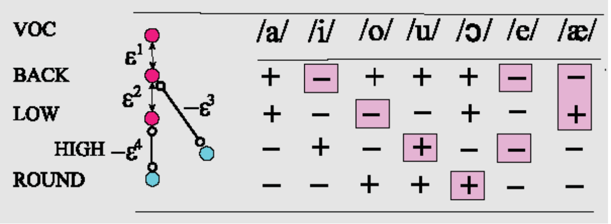


−back	+back	
/i/	/u/	+high
/e/	/o/	−high/−low
/æ/	/ɔ/ /a/	+low

The phonological features may be represented as by the atomic symbols BACK, LOW, HIGH, ROUND. The generic knowledge of the phonological agent concerning this fragment may be represented as a Hopfield network using *exponential weights* with basis $0 < \varepsilon \leq 0.5$.

Exponential weights and strict constraint ranking

Strong Constraints: LOW → ¬HIGH; ROUND → BACK



VOC	/a/	/i/	/o/	/u/	/ɔ/	/e/	/æ/
BACK	+	−	+	+	+	−	−
LOW	+	−	−	−	+	−	+
HIGH	−	+	−	+	−	−	−
ROUND	−	−	+	+	+	−	−

Assigned Poole-system

VOC $\leftrightarrow\varepsilon^1$ BACK; BACK $\leftrightarrow\varepsilon^2$ LOW
LOW $\leftrightarrow\varepsilon^4$ ¬ROUND; BACK $\leftrightarrow\varepsilon^3$ ¬HIGH

Keane's markedness conventions

Conclusion

- As with weighted logical system, OT looks for an optimal satisfaction of a system of conflicting constraints
- The exponential weights of the constraints realize a strict ranking of the constraints:
- Violations of many lower ranked constraints count less than one violation of a higher ranked constraint.
- The grammar doesn't count!

6 Learning

Translating connectionist and standard statistic methods of learning into an update mechanism of a penalty logical system.

Boersma & Hayes (2001): gradual learning algorithm (stochastic OT)
Goldwater & Johnson (2003): maximum entropy model
Jäger (2003): Comparison between these two models
Pater, Bhatt & Potts (2007)

These papers are also a starting point for understanding iterated learning and the modelling of (cultural) language evolution.

References

Blutner, R. (2004). *Neural Networks, Penalty Logic and Optimality Theory*. Amsterdam: ILLC.

Boersma, P., & Hayes, B. (2001). Empirical tests of the gradual learning algorithm. *Linguistic Inquiry, 32*, 45-86.

Darwiche, A., & Marquis, P. (2004). Compiling propositional weighted bases. *Artificial Intelligence, 157*, 81-113.

de Jongh, D., & Liu, F. (2006). *Optimality, Belief and Preference*: Institute for Logic, Language and Computation (ILLC), University of Amsterdam.

deSaint-Cyr, F. D., Lang, J., & Schiex, T. (1994). Penalty logic and its link with Dempster-Shafer theory, *Proceedings of the 10th Int. Conf. on Uncertainty in Artificial Intelligence (UAI'94)* (pp. 204-211).

Goldwater, S., & Johnson, M. (2003). *Learning OT constraint rankings using a maximum entropy model.* Paper presented at the Stockholm Workshop on Variation within Optimality Theory, Stockholm.

Jäger, G. (2003). Maximum entropy models and Stochastic Optimality Theory. Potsdam: University of Potsdam.

Kooij, J. F. P. (2006). Bayesian Inference and Connectionism. Penalty Logic as The Missing Link, *Essay written for the course "Neural Networks and Symbolic Reasoning"*. Amsterdam.

Lima, P. M. V., Morveli-Espinoza, M. M. M., & Franca, F. M. G. (2007). *Logic as Energy: A SAT-Based Approach* (Vol. 4729). Berlin, Heidelberg: Springer.

Pater, J., Bhatt, R., & Potts, C. (2007). Linguistic optimization. *Ms., University of Massachusetts, Amherst*.

Pater, J., Potts, C., & Bhatt, R. (2007). Harmonic Grammar with linear programming. *Ms., University of Massachusetts, Amherst.[ROA-827]*.

Pinkas, G. (1992). *Logical inference in symmetric connectionist networks.* Unpublished Doctoral thesis, Washington University, St Louis, Missouri.

Pinkas, G. (1995). Reasoning, connectionist nonmonotonicity and learning in networks that capture propositional knowledge. *Artificial Intelligence, 77*, 203-247.

Prince, A. (2002). Anything goes, *A new century of phonology and phonological theory* (pp. 66–90). New Brunswick: Rutgers University.

Rescher, N. (1964). *Hypothetical Reasoning*: North-Holland Pub. Co.

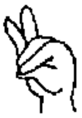
Lecture 1

1. What is the reason that no word with the pronunciation [bɛd] exists in Dutch? Do you expect a language change that makes possible such a word? What about the opposite pattern in English (disappearing of [bɛd])?
2. Construct the optimality tableaux for the voicing contrasts in Dutch using the lexical inputs /bɛd-ən/ and /bɛt-ən/ (and considering the output candidates [bɛ.dən] and [bɛ.tən]!)
3. Given the system of constraints {FAITH , ONSET, NOCODA}, what is the optimal analysis for the input /tatata/? Why is the result independent on the ranking of the constraints?
4. Assume the ranking FAITH>>ONSET, NOCODA. What is the optimal analysis for /əmerikə/? And what is the optimal analysis if we assume *Senufo*'s ranking NOCODA, ONSET >> FAITH?
5. Allow the Generator to realize more than one consonant at onset and coda. Furthermore, add the following two new constraints:

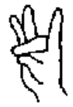
   Onsets must increase and codas decrease in sonority   SONORITY
   Syllables have at most one consonant at an edge   *COMPLEX

   Use the ranking SONORITY >> FAITH >> ONSET, NOCODA, *COMPLEX

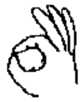
   a. What are the optimal outputs for /silindricl/ and /hAmstə/?
   b. Why [tank] is a possible (optimal) output but [takn] is not?
   c. Likewise, why [twin] is well-formed but [tkin] is not?
6. Consider some of the *contact handshapes* in Taiwan Sign Language (TSL) listed here and combined with a straightforward code:



(123)



(13)



(12)

The numbers correspond to the fingers: 1 = thumb, 2 = index, 3 = middle,…
Some of the fingers of the hand are *in contact.* These fingers are assumed to be "selected", the others are "unselected". The selected fingers are indicated in the code, e.g. (123).
For simplicity, we assume a very small space of potential "signs", namely {(1,2), (1,3), (1,2,3)}.

This set forms the input set and the output set of an OT systems. Assume further that GEN is totally free and pairs each input with each output. Next, consider the following "empirical generalizations":

1. each "sign language" realizes the sign described as (123)
2. when a sign language realizes (1,3) then it realizes (1,2)

Construct an OT system that deals with these "observations"!

Hint: make use of the markedness constraints INDEX and MIDDLE demanding the selection of the index finger and the middle finger, respectively. Assume a fixed (universal) ranking INDEX o MIDDLE. Discuss the factorial typology involving FAITH!

7. Construct the optimality tableaux for the voicing contrasts in Dutch using the lexical inputs /bɛd-ən/, /bɛt-ən/, /bɛd/ and /bɛt/. Instead of using the contextual markedness constraint CODA/*VOICE use the simpler constraint

*VOICE: *Obstruents must not be voiced.*

As an additional constraint use ‘positional faithfulness’:

FAITH[VOICE, ONSET]: *An output segment in the ONSET has the same value for VOICE as its input correspondent.*

Which ranking of the three constraints

FAITH[VOICE]
FAITH[VOICE, ONSET]
* VOICE

has to be assumed for Dutch? What might be the intuition behind positional faithfulness? What happens when *VOICE outranks the two faithfulness constraints?

Lecture 2

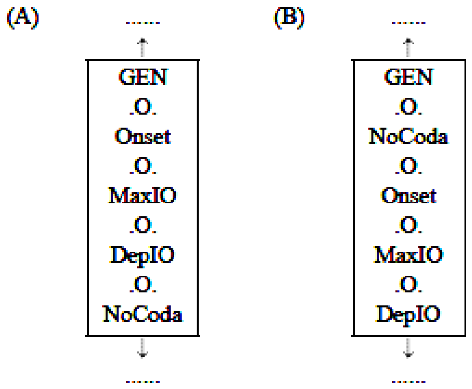
8. Discuss stress assignment for the input /mi.nχ.so.tχ/. Consider the listed candidate outputs only.

Input: /mi.nə.so.tə/	ROOT	WSP	TROCH	PARSE SYLL
1 (mí.nə)(só.tə)				
2 mi(n´ə.so)tə				
3 mi.nə(só.tə)				
4 (mí.nə)so.tə				
5 Mi(nə.só)tə				
6 (mi.n´ə)(só.tə)				
7 (mi.n´ə)só.tə				
8 mi.nə.so.tə				

9. Treat the syllabification of *hotél* and *vánity*!

Note: The following three exercises address the syllabification example (second part of lecture 2)

10. The following two **lenient** cascades should be applied to the input ‘bab’. Pretend you could see every intermediate step in the cascade and list the set of remaining candidates after each constraint application (ignore the intermediate stage after applying only GEN).



Hint: the result of [GEN .O. Onset] applied to 'bab' can be read off slide 20 (why?). Applying [GEN .O. NoCoda] to 'bab' leads to the following alternatives:

X[b]N[a]N[]X[b]
X[b]N[a]O[b]N[]
X[b]N[a]X[b]
X[b]N[a]X[b]N[]
X[b]N[]N[a]X[b]
X[b]O[]N[a]X[b]
O[]X[b]N[a]X[b]
O[b]N[a]N[]X[b]
O[b]N[a]O[b]N[]
O[b]N[a]X[b]
O[b]N[a]X[b]N[]
O[b]N[]N[a]X[b]
N[]O[b]N[a]X[b]
N[]X[b]N[a]X[b]

11. What would be a possible phonetic realization of the winning candidate for (B) in exercise 10? Assuming a suitable "phonetic filter" FST added by composition at the bottom, what happens if the augmented (B)-FST is run in the opposite direction (presented with the phonetic output form as input)?
 What would have to be done to implement interpretive optimization (e.g., for lexicon optimization) – rather than running the given lenient cascade in (B) (implementing expressive optimization) from bottom to top? How would the behaviour change when again applied to the phonetic output form resulting as the optimal candidate for 'bab'?
12. (Back to simple expressive optimization.) If we wanted to include the possibility for complex onsets and codas (as in English [$_\sigma$ stamp]) – how would we have to modify the definition of *Gen*?
 How would you formalize the constraints *COMPLEX ^{Ons} and *COMPLEX^{Cod} (which should have the obvious effect)?

Lecture 3

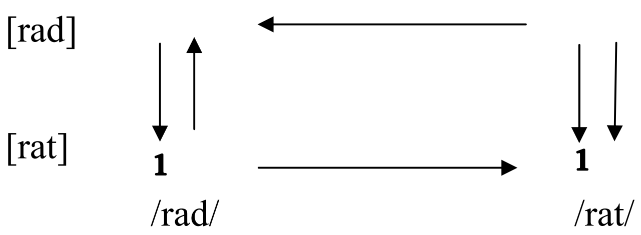
13. In Section 1 of this lecture we have seen how the ranking for *Hawaiian* and *Senufo* can be learned by using constraint demotion (triggering data pairs /atat/ - .a.ta.+t, for *Hawaiian*, and /atat/ - .9a.ta.+t, for *Senufo*). Are the triggering data pairs /atat/ - .9a.tat. (*Yawelmani*) and /atat/ - .a.tat. (*English*) sufficient for learning the correct rankings of the relevant constraints (basic syllable structure)?
14. Consider the overt form *tat* as input for the OT learning algorithm (Section 4). Start with the initial ranking Onset, NoCoda >> Faith. What is the resulting ranking after presenting the learner with the overt form *tat*?
 Hint: Remember that the OT learning algorithm combines robust interpretive parsing and constraint demotion. For robust interpretive parsing assume that *tat* can be parsed into (i) .tat. (with underlying form /tat/), (ii) .tat.<a>. (with underlying form /tata/), (iii) <a>.tat. (with underlying form /atat/).
15. In section 5 we investigated constraints on inventories by lexicon optimization. Use the OT learning algorithm (Section 4) and find out which inventory is established if the

system is presented (a) with the input [t], (b) with the input [d], (c) with the inputs [t], [d]. Assume that the generator pairs {/t/, /d/} freely with {[t], [d]}, and the initial hierarchy is **Obs/*Voice** >> **Faith[voice]**.

16. In *Imbura Quechua*, a language of Northern Equador, there are three voiceless stops: [p, t, k]. Except for a class of word borrowed from Spanish, voiced stops are not found contrastively in *Quechua*. However, stops in *Quechua* are voiced when appearing after a nasal; e.g. /t/ [nan-da] 'road-ACC'. The general pattern of voicelessness for obstruents requires a ranking Obs/*Voice >> Faith[voice]. In order to describe the kind of assimilation involved, an constraint Icc[Voice] has been introduced ('identical cluster constraint with regard to voicing'). How Icc[Voice] must be ranked in order to explain the case of allophony found in *Quechua*? Complete the corresponding diagram!

[nan-ta]			
[nan-da]			
	/nan-ta/		

17. The Rad/Rat-Problem (cf. Hale & Reiss 1998). In German there are two possible lexical inputs /rad/ (meaning *wheel*) and /rat/ (meaning *advice*). With regard to the present account we have the following diagram of bidirection:



Investigate the two possible rankings between the Markedness constraint (arrows marked with 1 in the diagrams) and Faithfulness (the other arrows). List the pairings for the two possibilities and make clear why the expected *pattern of ambiguity* (i.e. the pairing [rat]-/rad/, [rat]-/rad/) cannot be realized by the present account without further provisos.

Lecture 4

18. Take the input
{*write*(x,y), x=*Peter*, y=*what*, tense=future, auxiliary=*will*}
Construct a representative number of possible outputs!
19. Investigate subject-auxiliary inversion! Give an OT analysis of the following English examples:
    - *What will Peter write*
    - **What Peter will write*
    - **Will Peter write what*
    - **Peter will write what*

    Hint: use the constraints Op-Spec, Ob-Hd o Stay!
20. Consider the following early children questions:
    - □ *Where horse go?*

□ *What cowboy doing?*

What about the initial ranking of the Child Grammar? (you have to include the faithfulness constraint FULL-INT)

21. Consider the garden-path sentence

□ *Bill knew John liked Maria*

Give an analysis in terms of the Frazier model (using the OT formulation given in section 8) and compare it with the constraint theory of processing (section 9)!

22. Consider the following two sentences:

□ *I gave her earrings to Sally*
□ *I gave her earrings on her birthday*

Which of this two sentences exhibits a garden-path effect? Show that the prediction made by the model of Frazier (using the OT formulation given in section 8) are in conflict with the intuitions. What about the predictions of constraint theory of processing! [hint: allow a ternary branching structure for double object constructions]

Lecture 5

23. Consider the following sentences and determine the binding relations predicted by weak bidirectional OT using the constraints REFECON and BIND:

Often when I talk to a doctor$_i$,
(A) the doctor$_{\{i,j\}}$ disagrees with himself$_{\{i,j\}}$
(B) the doctor$_{\{i,j\}}$ disagrees with him$_{\{i,j\}}$

24. Consider Beaver's (to appear; in the reader) theory of local coherence, which is based on the following constraints:

PRO-TOP: The topic is pronominalized
COHERE: The topic of the current sentence is the topic of the previous one
ALIGN : The topic is in subject position

Ranking: PRO-TOP >> COHERE >> ALIGN

The *topic* of a sentence is defined as the entity referred to in both the current and the previous sentence, such that the relevant referring express-ion in the previous sentence was minimally oblique (if there is no such entity, the topic can be anything – for example in discourse initial sentences). Sentence topics are underlined in the following example:

Jane$_1$ is happy	< $\underline{1}$ >
Mary$_2$ gave her$_1$ a present$_3$	< 2 $\underline{1}$ 3 >
She$_{1/2}$ smiled	< $\underline{1}$ > / < $\underline{2}$ >

What is the optimal interpretation of the last sentence?
Finally, give an analysis of the following discourse:

Jane$_1$ is happy	< $\underline{1}$ >
Mary$_2$ gave her$_1$ a present$_3$	< 2 $\underline{1}$ 3 >
She$_{1/2}$ smiled at her$_{2/1}$	< 1 $\underline{2}$ > / < $\underline{2}$ 1 >

Lecture 6

25. Use penalty logic to formalize Rescher's Verdi-Bizet example: 'If Bizet and Verdi had been compatriots Bizet would have been Italian or Verdi would have been French'.

Hint: Use the (self-explaining) expressions I(v), F(b), COMP(v,b) $\leftrightarrow$ [(I(v)&I(b)) $\vee$ (F(v)&F(b))]

26. Prove the following fact: If Δ' is an *optimal scenario* of a formula α with regard to a penalty knowledge base <At, Δ, k>, then no model ν of α that verifies Δ' has a higher system energy $\mathcal{E}_{PK}(\nu)$ $(=_{\text{def}} \sum_{\delta\in\Delta} k(\delta)\, [\![\neg\delta]\!]_{\nu})$ than any model of α that doesn't verify Δ'.