

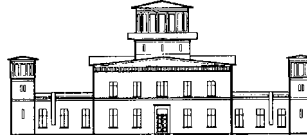


Theorien und Methoden der Phonologie

Prof. Dr.-Ing. Wolfgang Hess

Sprache und Kommunikation
Institut für Kommunikationswissenschaften
Universität Bonn
Poppelsdorfer Allee 47, 53115 Bonn

wgh@ikp.uni-bonn.de
<http://www.ikp.uni-bonn.de>



1. Grundbegriffe der Phonologie
2. Distinktive Merkmale
3. Generative Phonologie
4. Nichtlineare Phonologien
5. **Optimalitätstheorie**



Literaturempfehlung

Allgemein zur Phonologie

- Gussenhoven, Carlos / Jacobs, Haike (1998, ²2003): *Understanding Phonology*. Understanding Language Series (Arnold, London) **[abgekürzt GJ]**
- Kenstowicz, Michael (1994): *Phonology in generative grammar* (Blackwell, Oxford)
- Spencer, Andrew (1996): *Phonology* (Blackwell, Oxford)

Speziell zu Kapitel 5

- Archangeli, Diana / Langendoen, D. Terence (eds.) (1997): *Optimality theory: an overview* (Blackwell, Oxford) **[abgekürzt AL]**
- Boersma, Paul (1998): *Functional phonology* (PhD Diss., Univ. Amsterdam);
<http://fonsg3.hum.uva.nl/paul/papers/funphon.pdf>
- Boersma, Paul (2003): "Stochastic optimality theory", Manuscript,
<http://turing.wins.uva.nl/~rens/paul.pdf>
- Boersma, Paul / Hayes, Bruce (1999): "Empirical tests of the Gradual Learning Algorithm" [ROA #348]
- Boersma, Paul / Levett, Claartje (2003): "Optimality Theory and phonological acquisition", Manuscript,
http://fonsg3.hum.uva.nl/paul/papers/OT_phon_acq_ARLA2003.pdf
- Prince, Alan S. / Smolensky, Paul (1993): *Optimality theory: constraint interaction in generative grammar* (Rutgers Univ., New Brunswick, NJ, USA) [ROA #537 (seit November 2003)]
- Rutgers Optimality Archive (ROA): <http://roa.rutgers.edu/> (mit mehr als 500 Artikeln über Optimalitätstheorie und ihre Anwendungen, elektronisch zugänglich)



Theorien und Methoden der Phonologie

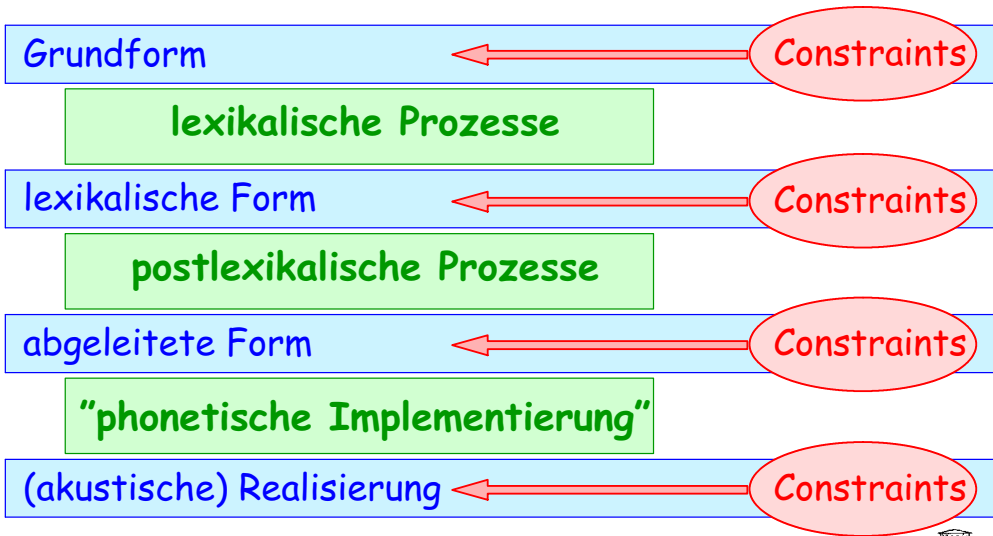
5. Optimalitätstheorie

- 5.1 **Regeln versus Constraints: Grundsätzliche Funktionsweise**
- 5.2 Optimalitätstheorie und die segmentale Ebene (einschließlich Silbenbildung)
- 5.3 Betonung und Optimalitätstheorie
- 5.4 *Constraints*, Kosten und ein statistischer Ansatz



Repräsentationen und Übergänge

Auf jeder Ebene bestehen Beschränkungen.



Im Rahmen phonologischer Prozesse existieren *Constraints* ... [AL:26]

- **in der Grundform:** Sie geben an, welche Laute erlaubt sind und wie sie kombiniert werden können
- **bei der Morphemverkettung:** Sie geben an, wie die Morpheme kombiniert werden
- **in den Regelwerken:** Sie geben an, welche Regeln angewendet werden können, und welche Laute daraus hervorgehen
- **an der Oberfläche:** Sie verhindern Muster, die in der Oberflächenform nicht existieren können.

Alle *Constraints* werden in der jeweiligen Ebene als unverletzlich betrachtet.

Durch die *Constraints* verliert das Regelwerk seine Transparenz.



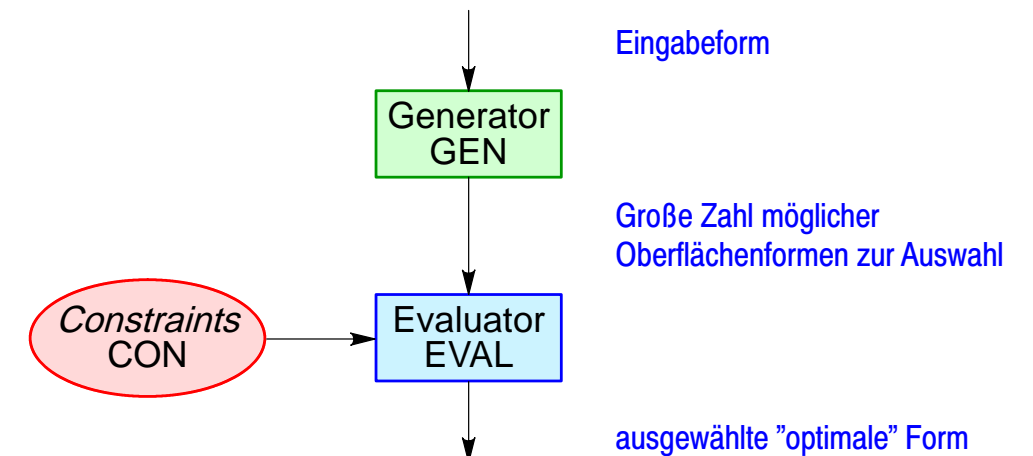
Motivierung der Optimalitätstheorie

In der Optimalitätstheorie wird auf die Anwendung von Regeln verzichtet.

- Die "Universalgrammatik" wird repräsentiert durch eine Menge universeller *Constraints*.
- Einzelne *Constraints* können zueinander in Widerspruch stehen und werden daher zwangsläufig verletzt.
- Die Rangfolge der *Constraints* und das Gewicht, das jedes von ihnen hat, sind sprachenspezifisch und für jede Sprache individuell festgelegt.
- Eine Form wird als die "optimale" betrachtet, wenn sie so wenig wie möglich hochrangige *Constraints* verletzt.



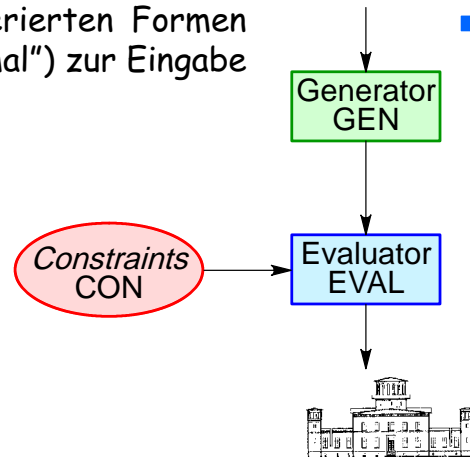
Optimalitätstheorie - Das formale Modell [AL:14]




Der **Generator** erzeugt für eine vorhandene Eingabe eine Menge potenzieller Ausgabeformen (im allgemeinsten Fall unbeschränkt, in der Praxis aber durch *Constraints* begrenzt).


Der **Evaluator** wählt aus den generierten Formen diejenige aus, die am besten ("optimal") zur Eingabe passt.

Die Auswahlkriterien für den Evaluator stammen aus einer Menge **universeller Beschränkungen ("Constraints")** mit sprachenspezifischer Rangfolge.



Eingabe	CONSTRAINT1	CONSTRAINT2	CONSTRAINT3	CONSTRAINT4
Gen1	*!			
Gen2			*!	
 Gen3				*
Gen4		*!		
Gen5			*!	*

Zeichenerklärung

- Genx aus der Eingabeform generierte Formen
- * Verletzung eines Constraints
- *! fatale Verletzung; führt zum Ausschluss der jeweiligen Form
-  als optimale Form selektiert
- (abgeschattet) als *Constraint* irrelevant

Faithfulness

Eine der wichtigsten Familien der *Constraints* verlangt die grundsätzliche Übereinstimmung von Eingabe- und Ausgabeform. Diese *Constraints* werden als Identitäts- oder *Faithfulness Constraints* bezeichnet.

- **MAX-IO** (*Maximum I to O*): Jedem Segment in der Eingabeform (*Input I*) muss ein Segment in der Ausgabeform (*Output O*) entsprechen. Die Eingabeform ist in der Ausgabeform maximal vertreten. Es darf also kein Segment elidiert werden.
- **DEP-IO** (*Dependency I to O*): Jedem Segment in der Ausgabeform *O* muss ein Segment in der Eingabeform *I* entsprechen. Die Ausgabeform ist von der Eingabeform komplett abhängig. Es darf also kein Segment eingefügt werden.
- **IDENT(F)** (*Identity of Form*): Ein Segment in der Eingabeform ist identisch zu dem entsprechenden Segment in der Ausgabeform. Es darf also kein Segment verändert werden.

Markiertheit

Sprachelemente, die seltener vorkommen oder schwieriger zu produzieren sind als andere, werden als "markiert" bezeichnet.

"Markierte" Elemente

- kommen bei einem Vergleich der Sprachen der Welt seltener vor;
- kommen in einer Sprache nur gemeinsam mit dem entsprechenden unmarkierten Element vor;
- werden beim Spracherwerb später erworben als unmarkierte Elemente;
- erfordern mehr artikulatorische Anstrengung als unmarkierte Elemente.

Markiertheit: Beispiel

Gerundete Vorderzungenvokale und ungerundete Hinterzungenvokale sind "markiert".

Beispiele: Vokale [y], [ɨ] und [ɯ]; (betrachtet: germanische, romanische, slavisches Sprachen, Finnisch, Ungarisch, Türkisch)

- [y]: germanische Sprachen außer Englisch; in den romanischen Sprachen nur Französisch; keine slavisches Sprache; Finnisch, Türkisch
- [ɨ]: einige slavisches Sprachen (Russisch, Polnisch)
- [ɯ]: Türkisch
- [i] und [u]: alle Sprachen

Fazit: In keiner Sprache ein [y] ohne [i] oder ein [ɯ] ohne [u] oder ein [ɨ] ohne sowohl [i] als auch [u].



Typische Markiertheit-Constraints

Markiertheits-Constraints verbieten "markierte" Segmente. Sie können auf allen Ebenen definiert werden.

- ***COMPLEX**: verbietet Konsonantenfolgen mit mehr als einem Konsonanten
- ***DORSAL**: verbietet dorsale Konsonanten (typisches Kinder-Constraint beim Spracherwerb)
- ***VOICE(FINAL)**: verbietet stimmhafte Konsonanten in finaler Stellung (verlangt Auslautverhärtung)
- ***SIBSIB**: verbietet direkte Aufeinanderfolge zweier Sibilanten [sʃzʒ]
- ***[αVOICE][-αVOICE]**: verbietet Wechsel der Stimmhaftigkeit in Obstruenten



Markiertheit vs. Faithfulness

Markiertheits-Constraints sorgen dafür, dass "unmarkierte" Segmente vor "markierten" bevorzugt werden. **Beispiel [GJ:47]:** gäbe es keine "markierten" Segmente, würden im Extremfall alle Silben wie [ba] klingen.

Demgegenüber tendieren **Faithfulness-Constraints** dazu, die (im Extremfall unbeschränkte) Vielfalt der generierten Eingabeformen für die Ausgabe zu erhalten.

Faithfulness- und **Markiertheits-Constraints** verhalten sich demnach antagonistisch. Die segmentale Komplexität einer Sprache hängt davon ab, welchen Rang die einzelnen **Constraints** in dieser Sprache besitzen.



Aufspaltung von Constraints

Constraints lassen sich aufspalten. So kann man ihren Geltungsbereich auf einzelne Klassen von Segmenten oder einzelne Merkmale beschränken. **Beispiel Faithfulness:**

- **FAITH(V)**: Identitätsvorschrift für Vokale
- **FAITH(C)**: Identitätsvorschrift für Konsonanten
- **FAITH(VOICE)**: Identitätsvorschrift für Stimmhaftigkeit

Im Zweifelsfall geht diese Aufspaltung hinunter bis in das einzelne distinktive Merkmal.

Man strebt aber an, mit der geringstmöglichen Anzahl von **Constraints** auszukommen. Aufspaltungen werden erst dann vorgenommen, wenn sie sich als unvermeidlich erweisen.



Da die *Constraints* prinzipiell widersprüchlich sind, entstehen zwangsläufig **Konflikte**. Diese werden wie folgt gelöst.

- Jedes *Constraint* darf verletzt werden, wenn dadurch die Verletzung eines höherrangigen *Constraints* vermieden wird.
- Jede generierte Form wird irgendwo ein *Constraint* verletzen.
- **Optimal** - und damit **selektiert** - ist die Form, die nur **niedrig-rangige *Constraints* verletzt**.

Beispiel: Epenthese verletzt das *Constraint* DEP-IO. Wenn in einer Sprache Epenthese zugelassen sind, bedeutet dies, dass in der für diese Sprache spezifischen Rangfolge phonologische *Constraints* existieren, die die Epenthese fordern und höher rangieren als DEP-IO.



- 5.1 Regeln versus *Constraints*: Grundsätzliche Funktionsweise
- 5.2 **Optimalitätstheorie und die segmentale Ebene (einschließlich Silbenbildung)**
- 5.3 Betonung und Optimalitätstheorie
- 5.4 *Constraints*, Kosten und ein statistischer Ansatz



Beispiel: Pluralbildung Englisch [1]

[GJ:49]

Beteiligte Constraints:


- **DEP-IO**: Es darf kein Segment eingefügt werden (*Faithfulness*).
- **IDENT(F)**: Es darf kein Segment verändert werden (*Faithfulness*).
- ***SIBSIB**: Es dürfen keine zwei Sibilanten [szʒ] unmittelbar aufeinander folgen (Markiertheit).
- ***αVOICE/-αVOICE**: Aufeinanderfolgende Obstruenten müssen hinsichtlich ihrer Stimmhaftigkeit übereinstimmen (Markiertheit).
- **IDENT(STAMM)**: Der Wortstamm darf durch den Prozess nicht beeinflusst werden. (Dieses *Constraint* rangiert höher als alle anderen und ist in der folgenden *Tableau*-Darstellung bereits bei der Auswahl der generierten Formen berücksichtigt.)


Der Prozess [z] → [əz] verletzt **DEP-IO**.
 Der Prozess [z] → [s] verletzt **IDENT(F)**.




Beispiel: Pluralbildung Englisch [2]

[GJ:49]

kɪs-z	*SIBSIB	DEP-IO	*αVOICE/-αVOICE	IDENT(F)
kɪsz	*!			
 kɪsəz		*		
kɪss	*!			

bæk-z	*SIBSIB	DEP-IO	*αVOICE/-αVOICE	IDENT(F)
bækz			*!	
bækəz		*!		
 bæks				*

ɪn-z	*SIBSIB	DEP-IO	*αVOICE/-αVOICE	IDENT(F)
 ɪnz				
ɪnəz		*!		
ɪns				*!

Constraints zur Silbenbildung

- **PEAK**: Die Silbe muss einen Silbenkern (Vokal) besitzen.
- **ONSET**: Die Silbe muss einen Onset besitzen.
- ***CODA**: Die Silbe darf keine Coda besitzen.
- **FAITH(.)**: Ein Segment ist wie generiert auszusprechen.
- ***COMPLEX**: Onset und Coda dürfen nicht mehr als einen Konsonanten umfassen.
- **SONORITY**: Die Sonorität muss im Onset zum Silbenkern hin steigen und in der Coda vom Silbenkern weg fallen.
- **LICENSING(SYLL)**: Eine Wortform besteht aus Silben, und jedes Segment ist einer Silbe zuzuordnen.



Beispiel Yawelmani (1)

[AL:8]

Yawelmani ist eine Indianersprache aus Kalifornien. Die beobachtete Silbenstruktur ist wie folgt:

- Am Wortanfang steht immer genau ein Konsonant.
- Am Wortende steht höchstens ein Konsonant.
- Zwischen zwei Vokalen im Wortinneren stehen entweder ein oder zwei Konsonanten.


Relevante *Constraints*:

- **PEAK**: Jede Silbe besitzt einen Vokal.
- **ONSET**: Jede Silbe besitzt einen Onset.
- ***COMPLEX**: Onset und Coda jeder Silbe besitzen je höchstens einen Konsonanten.
- ***CODA**: Es existieren Silben mit und ohne Coda. Dieses *Constraint* ist also optional.



Beispiel Yawelmani (2)

[AL:12]


xat-en	PEAK	ONSET	*COMPLEX	FAITH(C)	FAITH(V)	*CODA
 xa.ten						*
xat.en		*!				
xa.te.n	*!					
xa.te				*!		
xa.te.ni					*!	




Beispiel Yawelmani (3)

[AL:12]

Beispiel [logw] Verbstamm "zermahlen" + [en] Suffix Futur

logw-en	PEAK	ONSET	*COMPLEX	FAITH(C)	FAITH(V)	*CODA
 log.wen						*
logw.en		*!	*			*
lo.gwen			*!			

Beispiel [logw] Verbstamm "zermahlen" + [hin] Suffix Vergangenheit

logw-hin	PEAK	ONSET	*COMPLEX	FAITH(C)	FAITH(V)
logw.hin			*!		
log.whin			*!		
log.w.hin	*!				
log.hin				*!	
 lo.giw.hin					*

Das *Constraint* *COMPLEX nimmt im Englischen einen geringen Stellenwert ein.

limp	limpness	*lim_ness	*limp[i]ness
soft	softness	*sof_ness	*soft[i]ness
strange	strangeness	*stran_ness	*strang[i]ness

	limp-nəs	FAITH(V)	PEAK	FAITH(C)	*COMPLEX
☞	limp.nəs				*
	lim.nəs			*!	
	lim.pɪ.nəs	*!			
	lim.p.nəs		*!		

Merke: Sind alle *Constraints* FAITH(.) in der Hierarchie vorgekommen, so kommt kein weiteres *Constraint* mehr zum Zuge.



- 5.1 Regeln versus *Constraints*: Grundsätzliche Funktionsweise
- 5.2 Optimalitätstheorie und die segmentale Ebene (einschließlich Silbenbildung)
- 5.3 Betonung und Optimalitätstheorie
- 5.4 *Constraints*, Kosten und ein statistischer Ansatz



- LICENSING(FOOT): Jede Silbe wird einem Fuß zugeordnet (*Anmerkung*: Extrametrische Silben verletzen dieses *Constraint*!).
- *FINAL: Ein Fuß steht nicht am Ende des Wortes (*Anmerkung*: Dies ermöglicht extrametrische Silben).
- ROOTING: Jedes Wort besitzt eine betonte Silbe.
- RHTYPE(T): Füße haben Trochäusform.
- RHTYPE(I): Füße haben Iambusform.
- PARSESYLL: Ordne jede Silbe einem Fuß zu.
- FTBIN: Füße sind zweiwertig (zwei Silben oder zwei Morae).
- ALIGN(Σ , R, ω , R): Der rechte Rand jedes Fußes wird zum rechten Rand des phonologischen Wortes hin ausgerichtet.
- ALIGN(Σ , L, ω , L): Der linke Rand jedes Fußes wird zum linken Rand des phonologischen Wortes hin ausgerichtet.



/pataka/	ROOT	RHTYPE(T)	PARSESYLL	FTBIN	Beispiel
pataka	*!		*		-
pa[tá]ka				*!	-
pata[ká]			*!		-
☞ pa[tá][ká]					eléctron
☞ pa[táka]					vanílla
pa[taká]		*!			-
[pá]ta[ká]				*!	-
☞ [pá][tá][ká]					chimpánzée
☞ [pá][táka]					bàndána
[pá][taká]		*!			-
☞ [páta]ka					Cánada
[patá]ka				*!	-
☞ [páta][ká]					Tènnessée
[patá][ká]		*!			-

Garawa ist eine Sprache aus Nord-Australien.

Linksdominant / Quantitäts-insensitiv: Betonung im Fuß ist (*).

Zuweisung der Füße: erster Fuß links, alle anderen von rechts nach links.

Deakzentuierung: Übrigbleibende einsilbige Füße (stets die dritte Silbe)

Als Ausrichtungs-(Alignment)-Constraints formuliert:

- $ALIGN(\omega, L, \Sigma, L)$: Der linke Rand des phonologischen Wortes ist gleichzeitig linker Rand eines Fußes (Anmerkung: Dies bindet nur *einen* Fuß!)
- $ALIGN(\Sigma, R, \omega, R)$: Der rechte Rand jedes Fußes (soweit noch möglich!) wird zum rechten Rand des phonologischen Wortes hin ausgerichtet.

Constraint-Hierarchie:

- $ALIGN(\omega, L, \Sigma, L), FTBIN \gg PARSESYLL \gg ALIGN(\Sigma, R, \omega, R)$




$ALIGN(\omega, L, \Sigma, L), FTBIN \gg PARSESYLL \gg ALIGN(\Sigma, R, \omega, R)$

Füße	['ja.mi]	['pun.ja.la]	['wa.cim.,pa.ŋu]
Regel	(* .)	(* .)	(* .)(* .)
Füße	['ka.ma.la.rin.ji]	['ja.ka.,la.ka.,lam.pa]	
Regel	(* .) (* .)	(* .)(* .)(* .)	
Füße	['ŋan.ki.ri.,ki.rim.,pa.ji]		
Regel	(* .) (* .)(* .)		



$ALIGN(\omega, L, \Sigma, L), FTBIN \gg PARSESYLL \gg ALIGN(\Sigma, R, \omega, R)$

ka.ma.la.rin.ji	ALI(·L)	FTBIN	PARSYL	ALI(·R)
(ka.ma)(la.rin)ji			*	*!
ka(ma.la)(rin.ji)	*!		*	
(ka.ma)(la.rin.ji)		*!		
 (ka.ma)la(rin.ji)			*	
(ka.ma)(la)(rin.ji)		*!		
(ka.ma)la.rin.ji			*	*!



- 5.1 Regeln versus Constraints: Grundsätzliche Funktionsweise
- 5.2 Optimalitätstheorie und die segmentale Ebene (einschließlich Silbenbildung)
- 5.3 Betonung und Optimalitätstheorie
- 5.4 Constraints, Kosten und ein statistischer Ansatz



Kostenfunktionen vs. Regeln

Viele Funktionen in sprachverarbeitenden Systemen sind regelbasiert.

- Die meisten regelbasierten Probleme können umformuliert werden als Minimierungsproblem einer Kostenfunktion [STÖBER, 1999].
- Regeln ergeben harte, üblicherweise lokale Entscheidungen.
- Kostenfunktionen gestatten demgegenüber flexiblere Entscheidungen [z.B. wird eine globale Optimierung oder die Integration verschiedener *Constraints* sehr viel einfacher].

Die Optimalitätstheorie zeigt, dass diese Sichtweise auch in der Linguistik und der Phonologie ihren Einzug gehalten hat.



Constraints als Sonderfall von Kostenfunktionen

Kosten- und Risikofunktionen dominieren weite Bereiche der mathematischen Optimierungsverfahren. Die Optimierung erfolgt durch Minimierung des Risikos.

Die Maximierung einer Wahrscheinlichkeits- oder Fitnessfunktion ist ein vergleichbares Verfahren mit umgekehrtem Vorzeichen.

Constraints und Kostenfunktionen können sich entsprechen.

Die Optimalitätstheorie liefert, wenn sie entsprechend erweitert wird, das Handwerkszeug für eine globale Optimierung größerer Formen, über die Anwendung auf lokaler Ebene hinaus. Sie erscheint auch statistischer Sichtweise zugänglich.

Constraint	Kosten
*!	unendlich
*	endlich
[OK]	Null



Statistische Optimalitätstheorie [1]

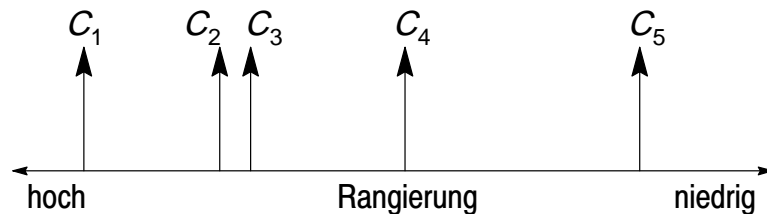
[BOERSMA, 2003]

Die Grammatik in der Optimalitätstheorie besteht aus einer Menge von *Constraints* und ihrer (festen) Hierarchie.

"Deterministische" OT: Feste Rangordnung ("the winner takes it all")

$$C_1 \gg C_2 \gg C_3 \gg C_4 \gg C_5$$

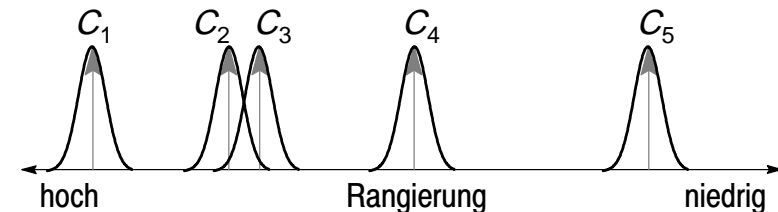
Erster Schritt zur "statistischen" OT: Rangordnung entlang einer kontinuierlichen Skala mit ungleichgewichtigen Distanzen



Statistische Optimalitätstheorie [2]

[BOERSMA, 2003]

Zweiter Schritt zur "statistischen" OT: Ersetzen eines festen Ranges durch eine Verteilung (damit ergeben sich "verrauschte" *Constraints*)



In diesem Beispiel kann durch die Überlappung der Verteilungen zwischen C_2 und C_3 die allgemeine Rangfolge für einzelne Formen umgekehrt werden, beispielsweise $C_2 \gg C_3$ in 90% der Fälle und $C_3 \gg C_2$ in 10% der Fälle.



(Neu)-Ordnung der Constraints im Spracherwerb

[BOERSMA/HAYES, 1999]

Spracherwerb [eigene Muttersprache oder Fremdsprache] bedeutet Ordnen und Neuordnen der *Constraint*-Hierarchie.

- Beim Erwerb der Muttersprache (L1) werden *Constraints* der kindlichen Sprachproduktion und der Wahrnehmung gegeneinander abgeglichen, bis die Rangfolge der *Constraints* die gleiche ist wie beim Erwachsenen. Dann stimmen wahrgenommene und produzierte Formen überein.
- Beim Erwerb einer Fremdsprache (L2) werden die für L1 geltenden *Constraints* mit dem System von L2 abgeglichen, bis der Schüler die Rangfolge der *Constraints* für L2 gelernt hat. Dann werden auch dort wahrgenommene und produzierte Formen übereinstimmen.

Als Formalismus ist die OT zur Beschreibung gut geeignet.



(Neu)-Ordnung der Constraints im Spracherwerb: Beispiel

[SMOLENSKY, 1996; hier nach BOERSMA/LEVELT, 2003]

Das Kind stellt bei der Produktion das Markiertheits-*Constraint* NoCODA zunächst über die *Faithfulness-Constraints*.

Produktion

kæt	NoCODA	DEPIO	IDENT(F)	MAXIO
kæt	*!			
☞ kæ				*
kæti		*!		
kæi			*!	

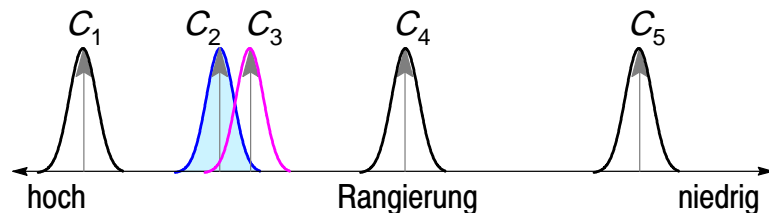
Perzeption

Kand.	Perzept	NoCODA	DEPIO	IDENT(F)	MAXIO
☞ kæt	kæt	*			
kæ	kæt	*	*!		
kæti	kæt	*			*!
dog	kæt	*		*!***	

Der offensichtliche Konflikt bringt das Kind zum Umsortieren der Constraints.

Die Statistische Optimalitätstheorie ermöglicht Aussprachevarianten.

Constraints, deren Toleranzbereiche sich überlappen, erlauben Optionen, d. h., dass die Oberflächenform mal so, mal anders erscheinen kann.



Insbesondere wenn *Constraints* rearrangiert werden, kann das so erfolgen, dass ein *Constraint* langsam die Skala hinab oder hinauf wandert. Im Bereich der Überlappung wird es zu Optionen kommen - die Aussprache ändert sich beim Spracherwerb nicht schlagartig.

