Computational Phonology

Interpretative Computing – Mapping to Phonetics

Dafydd Gibbon

Bielefeld University
Jinan University, Guangzhou

Interpretative Computing – Mapping to Phonetics

Generative Phonologies
Optimality Theories
Finite State Phonologies: Two-Level, Cascaded

Linguistic and phonetic representations: a search space

- The search target:
  - A representation of a phonetic *type*
  - Judged by comparison with a phonetic *token*
    - by a speaker-hearer *directly*
    - uttered by a speaker-hearer and *measured*
- The two main search methods:
  - Generate a premise (Generative Phonology)
    - Change until the result matches the search target
  - Generate lots of premises (Optimality Theory)
    - Select smaller subsets until the best target match is found

Linguistic and phonetic representations: a search space

- The main deductive search methods:
  1) Generate a premise (logic, algebra)
     - Change with transformation rules until the outcome fits the intuitively determined search target
       → Generative Phonologies
       → Finite State Phonologies
  2) Generate all possible outcomes (set-theory)
     - Select smaller subsets with constraint rules until the outcome fits the intuitively determined search target
       → Optimality Theories
       → Preference Theories

Linguistic and phonetic representations: a search space

- The main inductive search methods:
  1) Supervised learning:
     - Measure properties of empirical inputs and classify them in terms of search targets
     - Measure new empirical inputs and statistically select most similar (‘most probable’) search target
  2) Unsupervised learning:
     - Measure properties of empirical inputs and classify them in terms of a hierarchy of similarities
- Both types of learning are used in
  1) Speech engineering (ASR, TTS, ...)
  2) Artificial Intelligence (person profiling)

Linguistic-Phonetic Mapping

either (Generative Phonologies)

- *generate* (lexicon, grammar) → underlying structure
- *interpret* (underlying structure) → *the* phonetic representation

or (Preference Theories, Markedness Theories, Default Theories)

- archiphoneme, phoneme and allophone relations
- define syntagmatic and paradigmatic markedness on the basis of frequency, ease of production/perception, e.g. for voicing

or (Optimality Theories)

- [LEXICON etc.: *make underlying structures* – in earlier theories]
- GEN: *generate* (something) → lots of phonetic representations for the underlying structures
- SEARCH:
  - CON: *define constraints* for filtering search space
  - EVAL: *filter* (lots of phonetic representations) → *best* representations

Deductive Search

generate **(lexicon, grammar) → underlying structure**

interpret **(underlying structure) → phonetic representation**

Deductive computing: grammar + lexicon → phonetics

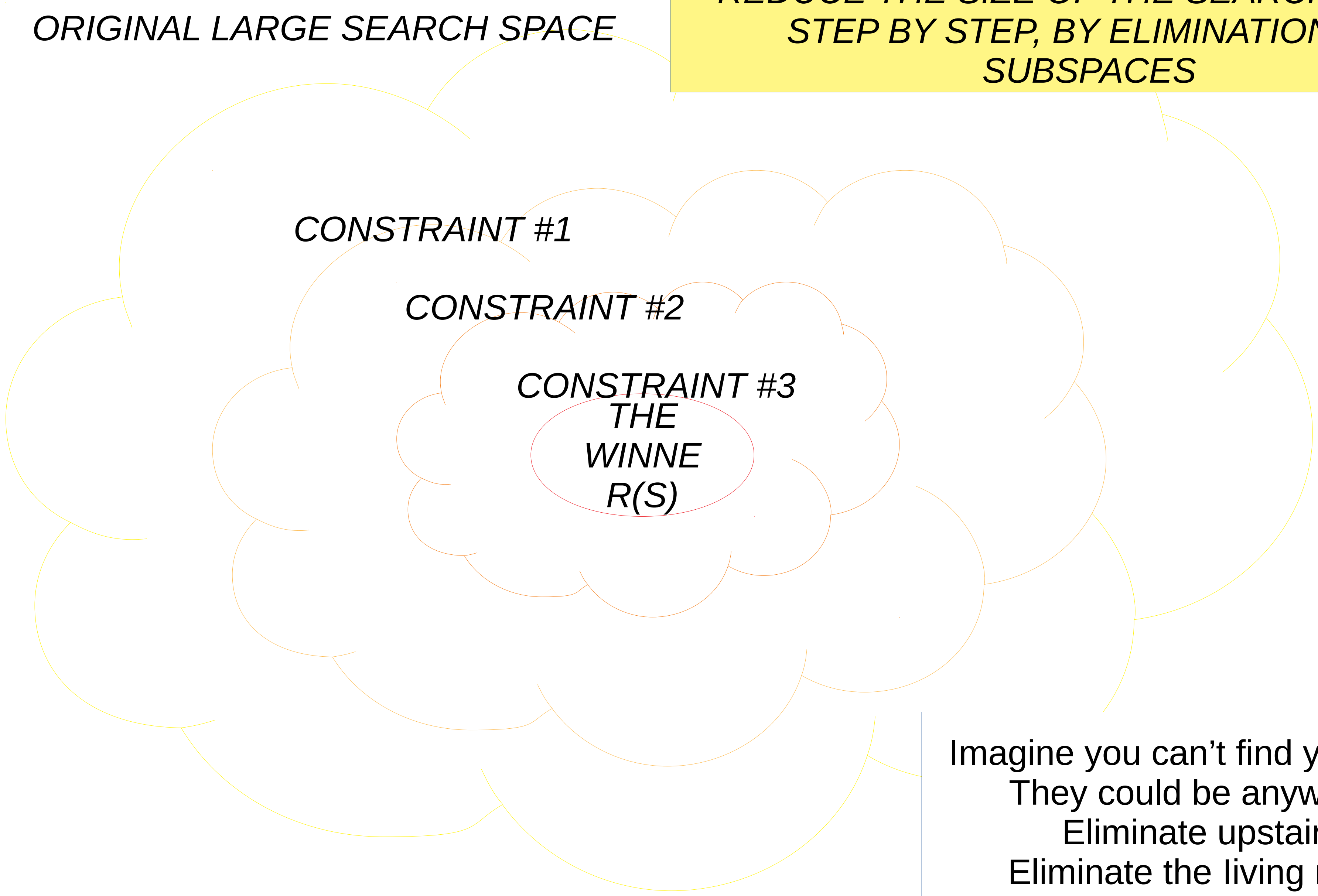
- Generative Phonologies, rule properties:
  - Lexical rules
    - Lexical Phonology
  - Post-lexical rules
    - Phonological Cycle
- Optimality Theory, constraint properties:
  - Faith
  - Markedness
- Two-level Phonologies
  - OT style: Koskenniemi
  - GP style: Kay and Kaplan

Deductive computing: grammar + lexicon → phonetics

- The main deductive search methods:
  - Generate a premise
    - Change until it fits the search target
      - → Generative Phonologies
      - → Finite State Phonologies
  - Generate all possible premises
    - Select smaller subsets until the target is found
      - → Optimality Theory
- The main inductive search methods:
  - Signal processing
  - Machine learning with Hidden Markov Models (HMMs):
    - Train – test – apply
  - Reduce ambiguity with top-down information:
    - Lexical
    - grammatical

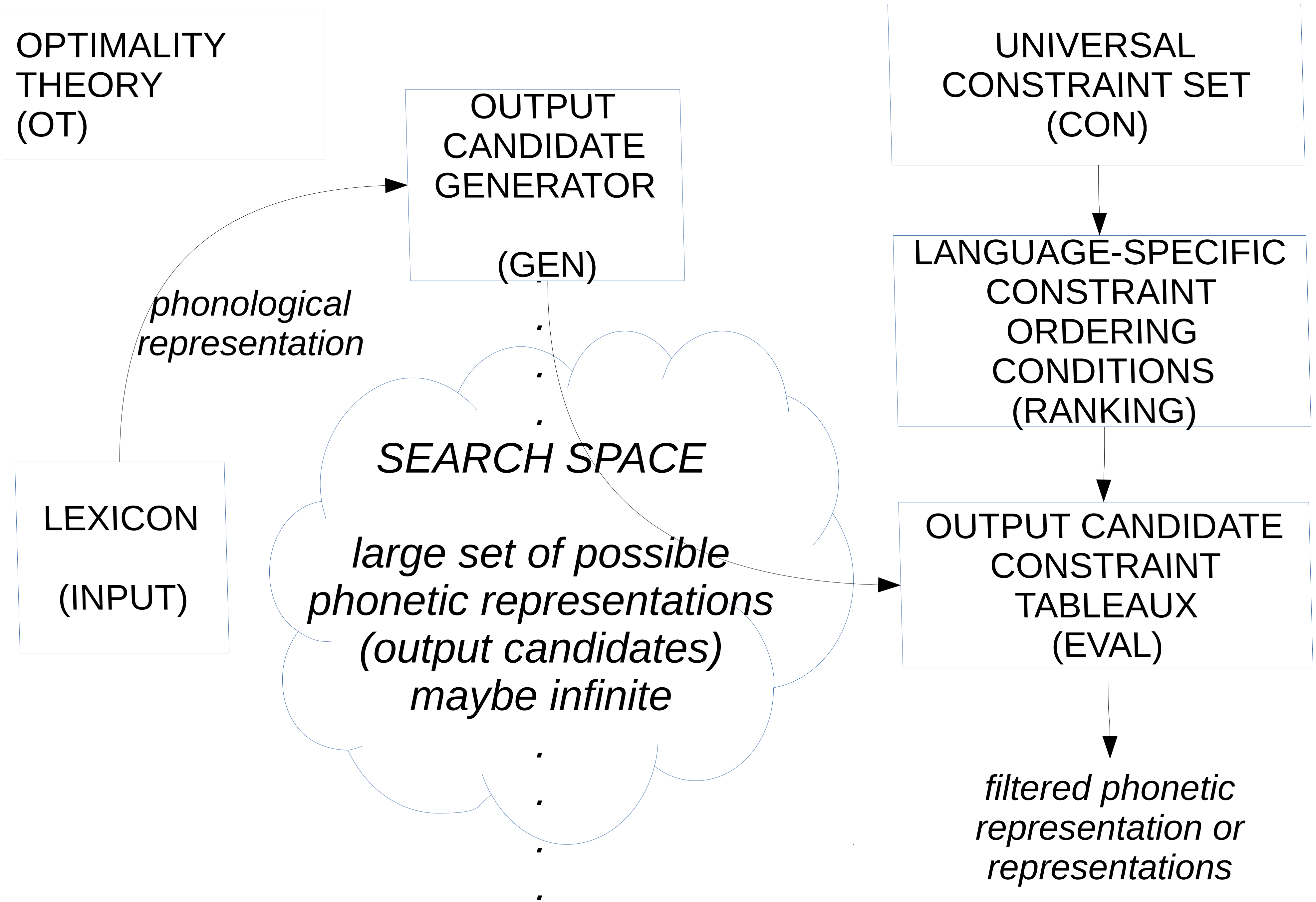
The Search Problem

REDUCE THE SIZE OF THE SEARCH SPACE STEP BY STEP, BY ELIMINATION OF SUBSPACES



Imagine you can’t find your keys.
They could be anywhere.
Eliminate upstairs.
Eliminate the living room.
Bingo: *IN THE FRIDGE* !!!!

Phonetic Interpretation as a Search Problem



GEN maps input to (mis)matching candidates

- Transduction constraints apply to segments, features, prosody:
  - Faithful
  - Feature differences
  - Order differences
  - Number of elements differences
  - No similarities at all
  - Possibly an infinite number of candidates
- Archangeli:
  - “in practice, linguists try to select the candidates that are closest to the winner and to show how these are eliminated by EVAL”
  - This appears to mean a <u>strong subjective element</u>
  - How to avoid this: <u>use objective computational methods.</u>

Types of OT constraint

- Classic types:
  - Faithfulness: similarity to lexical representation
  - Markedness: phonetic modifications
- Prosodic type:
  - alignment
- Later types:
  - antifaithfulness
  - local conjunction
  - ...

Specific types of constraint
in different linguistic domains
and different linguistic models
(sources: all over the internet)

General
Morphological
Syntactic
Antihomophony
Phonetic
Perceptual
Segmental
Phonotactic
featural
Autosegmental
Prosodic
Metrical
Accentual
Tonal
Intonational
Antialignment

Examples of OT <u>phonotactic</u> constraints

- Every CV syllable has
  - ONSET
    - syllable onset
  - PEAK
    - syllable nucleus
  - NOCODA
    - no syllable-final consonant
  - COMPLEX
    - syllable margins contain at most 1 consonant

Examples of OT faithfulness constraints

- Correspondence (transduction) constraints apply to segments, features, prosody:
  - MAX: input properties correspond to output properties
    - cf. Chomsky's Biuniqueness Condition on taxonomic phonologies
  - DEP: output depends on input
  - Both collapsed together as FAITH
  - CONTIGUITY
  - ALIGNMENT

OT example: final devoicing in German

- Constraints:
  - Markedness:
    - ~~VOP: All voiced obstruents forbidden!~~
    - *VF: Final voiced obstruents forbidden! *[+voiced]#
  - Faithfulness:
    - IDENTV: Don't change voicing! IDENT[voiced]

Input: tu:gend	*[+voiced]#	IDENT [voiced]
☞ a. tu:gent		*
b. du:gent		**!
c. tu:gend	*!	
d. du:gend	*!	*

OT example: final devoicing in German

- Constraints:
  - Markedness:
    - VOP: All voiced obstruents forbidden!
    - *VF: Final voiced obstruents forbidden!
  - Faithfulness:
    - IDENTV: Don't change voicing!

bed	*VF	IDENTV	VOP
☞ bet		*	*
pet		**!	
bed	*!		**
ped	*!	*	*

OT constraints in phonetic interpretation

- Constraints
  - are universal
    (not necessarily innate)
  - are either
    - phonologically grounded (FAITHFULNESS)
    - phonetically grounded (MARKEDNESS)
- Constraint ordering / ranking
  - is variable across languages
  - is language-specific
  - Discussion:
    - MARKEDNESS before FAITHFULNESS?

OT constraints in phonetic interpretation

- Each constraint is equivalent to an inference rule:
  - FOR each candidate:
    - IF match(candidate:constraint)
      THEN candidate → candidate+asterisk
- The constraint PARSE implies that the Input has
  - a phonotactic structure which may be a *tree*
  - segments with features like [+ voice], [-voice]
- Each Constraint can refer to a component of an input structure, such as
  - Feature: NO FINAL VOICING ≡ *[+voice]#
  - Category: NOCODA ≡ $^{*}{}_{\text{Coda}})_{\text{Syllable}})$

Problems

- Where does the input come from?
- How does OT relate to the lexicon?
- How do the constraints fit into an overall picture language architecture?
- What are the computational properties of OT in terms of time and space complexity?
- How to contain the combinatorial explosion of candidates, most of which are irrelevant?
- Maybe just generate candidates by using the constraints as rules in reverse?

Problems

- Where does the input come from?
- How does OT relate to the lexicon?
- How do the constraints fit into an overall picture language architecture?
- What are the computational properties of OT in terms of time and space complexity?
- How to contain the combinatorial explosion of candidates, most of which are irrelevant?
- Maybe just generate candidates by using the constraints as rules in reverse?

Good luck – and thanks for your attention!

To be continued ...