Introduction to Speech Synthesis

Simon King

Simon.King@ed.ac.uk

Centre for Speech Technology Research, Informatics Forum

Contents

Introduction

Unit Selection: text selection, recording data, search techniques, system architecture

From speech coding to statistical parametric synthesis (using HMMs)

2

Text-to-speech (TTS)

Definition: a text-to-speech system must be

- Able to read **any** text
- Intelligible
- Natural sounding

The first of these puts a constraint on the method we can choose: playback of whole words or phrases is not a solution

The second is actually easier to achieve than the third

3

Methods

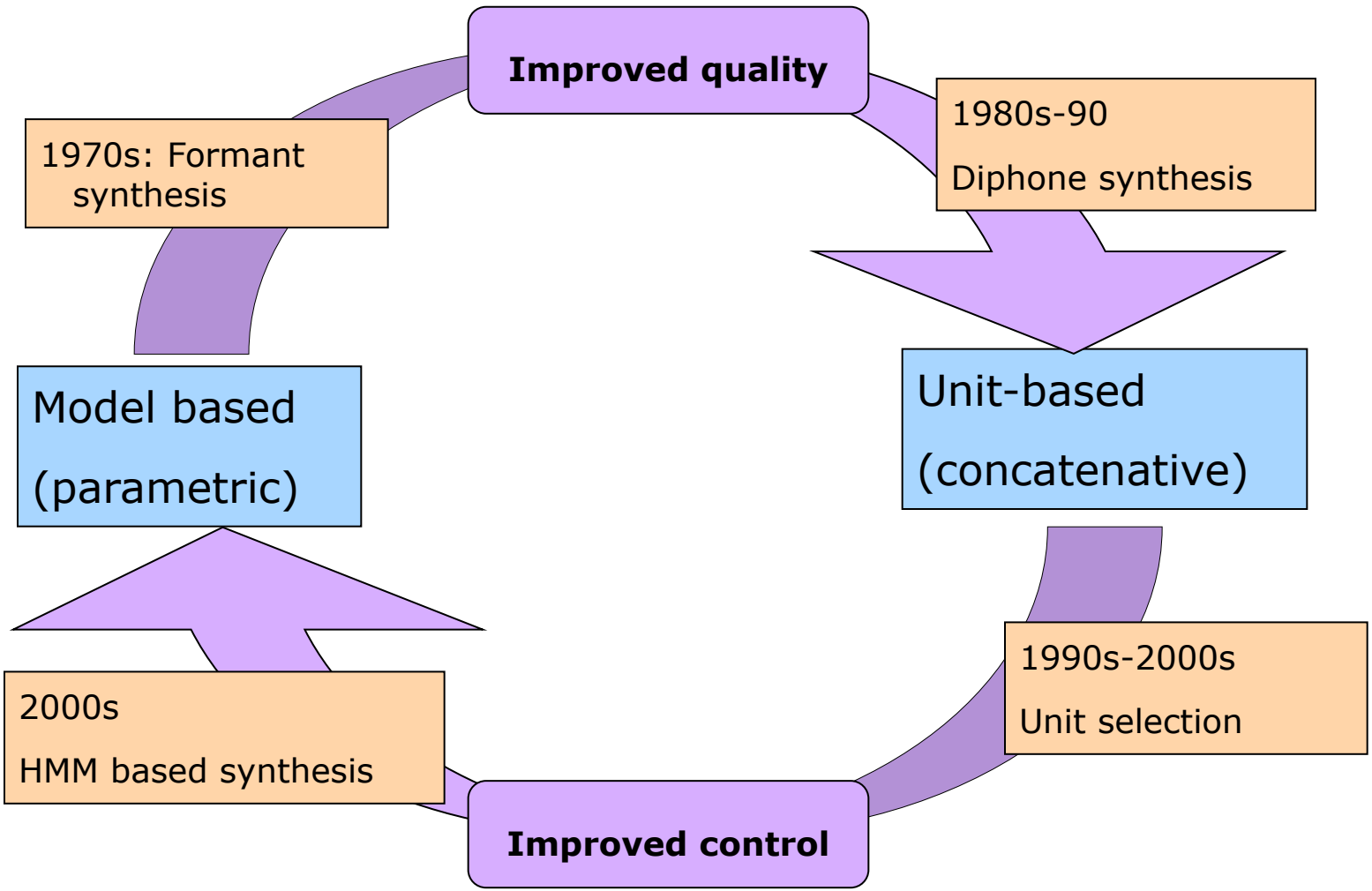
The methods available for speech synthesis fall into two categories:

1. Model-based, or parametric
2. Concatenative

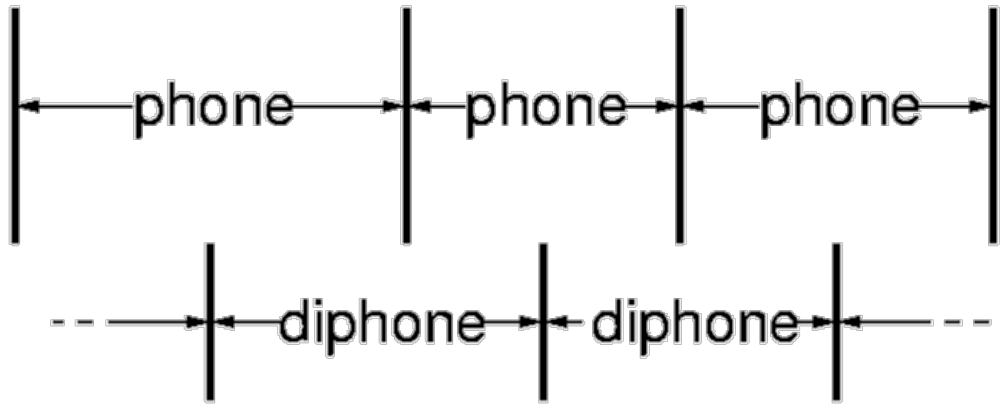
Model-based generally means some sort of simplified model of speech production, which requires values for a set of parameters (e.g. formant frequencies). These systems used to be driven by hand-crafted rules, but more recently tend to be trained on data

Concatenative systems use recorded examples of real speech

4


Improved quality
1980s-90
Diphone synthesis
Unit-based
(concatenative)
1990s-2000s
Unit selection
Improved control
2000s
HMM based synthesis
Model based
(parametric)
1970s: Formant
synthesis

Diphones



(time is running left to right in this diagram)

Why are diphones a good idea?

- Concatenation points (joins) are in the mid-phone position

Diphones are the second half of one phone plus the first half of the following phone

There is one join per phone

6

Text processing

Now we are going to look at the text processing aspect of speech synthesis.

Text processing breaks the input into units suitable for further processing, such as expanding abbreviations, part-of-speech (POS) tagging and letter-to-sound rules.

We end up with something that we can generate speech from, e.g., a phone sequence and a pitch contour.

7

Part-of-Speech

A morphological analysis can determine the part-of-speech (POS) information for many of the words, but some will have multiple possible POS categories.

We need to disambiguate the POS to determine pronunciation and to do syntactic analysis.

Why ?

1. The syntactic parser requires the POS tag for each word
2. Without POS information, pronunciation might be ambiguous e.g. “lives”
3. POS will also be used to predict the prosody later on

How ?

POS tagging is the process of determining a single POS tag for each word in the input.

There are two methods: deterministic and probabilistic.

8

The lexicon

The lexicon entries have three parts

1. Head word
2. POS
3. Phonemes

The POS is sometimes necessary to distinguish homographs, e.g.:

head	***POS***	***phonemes***
lives	noun	l ai v z
lives	verb	l I v z

9

Letter-to-sound

If lexical lookup fails, we fall back on letter-to-sound rules

Example:

The letter **c** can be realised as /k/, /ch/, /s/, /sh/, /ts/ or /ε/ [deleted]

We might write rules like:

- If the **c** is word-initial and followed by **I** then map it to phoneme /s/

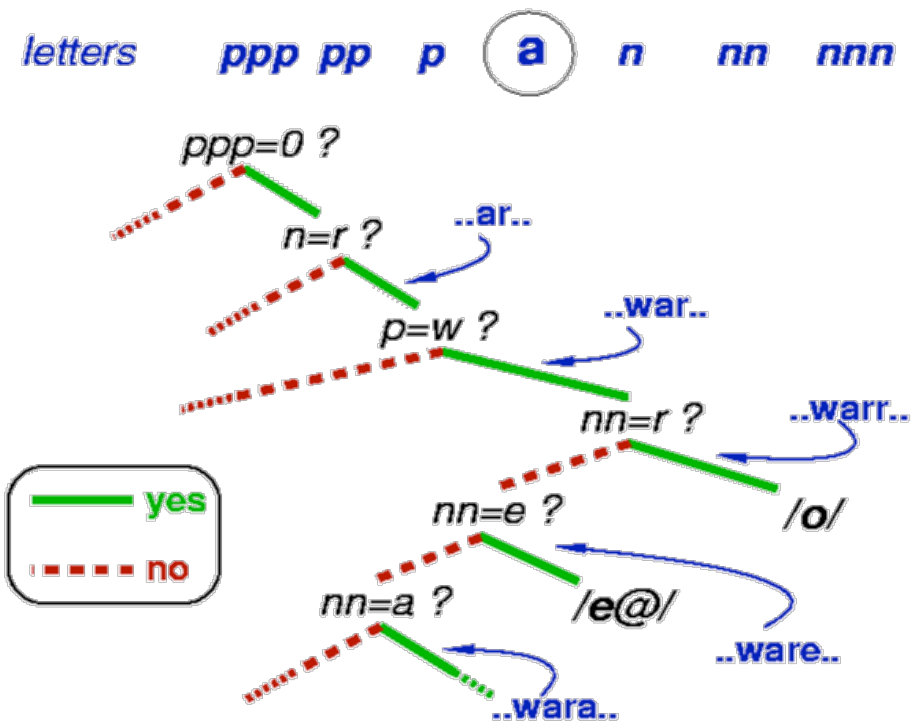
You can see why we might want to consider an automatic method for constructing these rules

- Classification trees

10

Part of Festival's LTS tree

Here is a fragment of the LTS tree from Festival: letter **a** for British English



11

Progress

text	Dogs	like	to	bark.	
token	(Dogs)	(like)	(to)	(bark)	(.)
POS	NNS	MD	TO	VB	.
Lex/lts	/d oh g z/)	/l ay k/	/t uw/	/b aa k/	
postlex	/d oh g z/	/l ay k/	/t ax/	/b aa k/	

12

Phrasing and accents: F0 and duration

Recap:

- We have processed the text into tokens and then into words
- We have determined a sequence of phones

We now turn to **suprasegmental** aspects of synthesis.

Some new terms:

- Intonation
- Prosody
- Accents
- Stress

For the discussion here, segments = phonemes

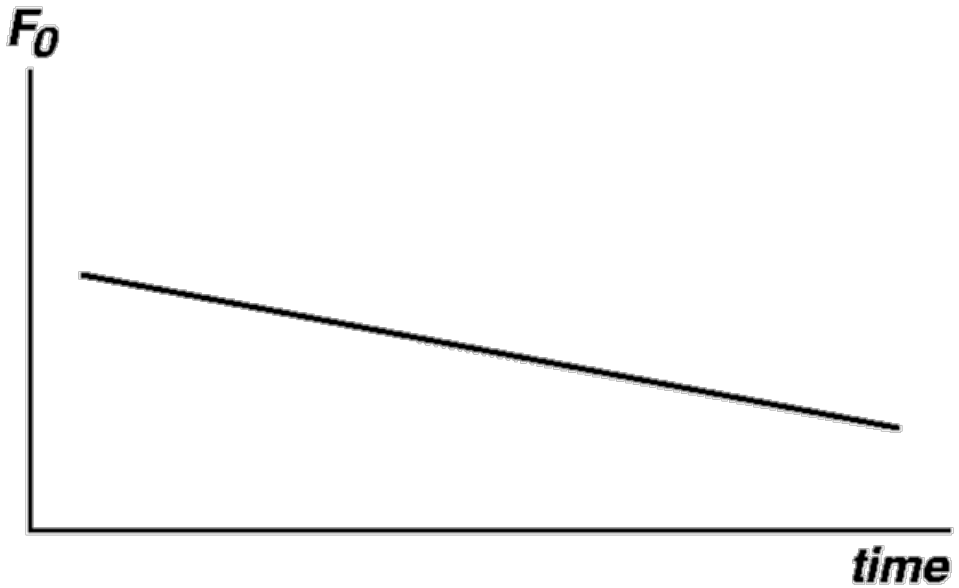
13

Tune

The **tune** of an utterance has two components:

- The global pitch contour shape
- Localised pitch accents

Most utterances show an overall downward trend in f0 called **declination**. We run out of breath, so air flow and pressure decrease and the vocal folds vibrate more slowly

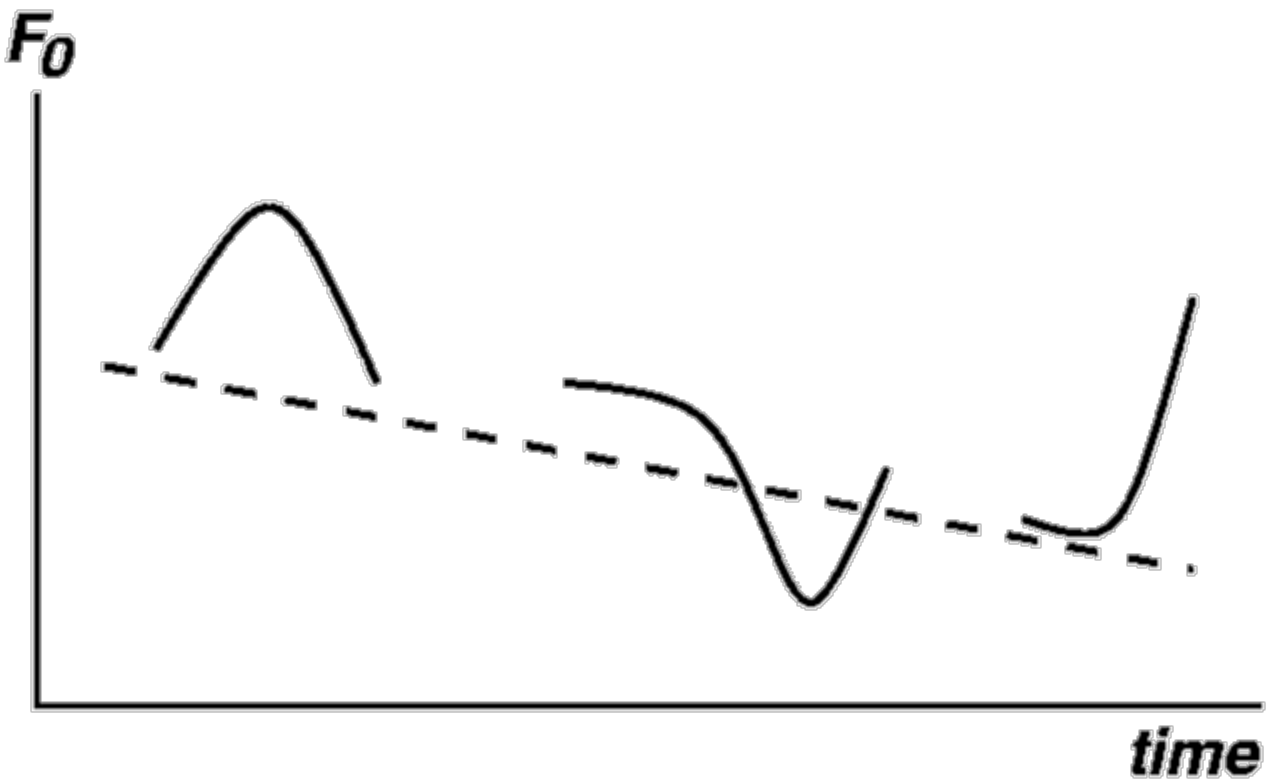


14

Tune: pitch accents

Superimposed on the baseline are **pitch accents**.

Pitch accents are located on (some) stressed syllables.



15

ToBI

Simplified description

- Two basic tones **H** (high) and **L** (low)
- Combine to get rise and fall patterns: **L+H**, **H+L**
- Use ***** to mark alignment with the stressed syllable
- Use % to mark a boundary tone

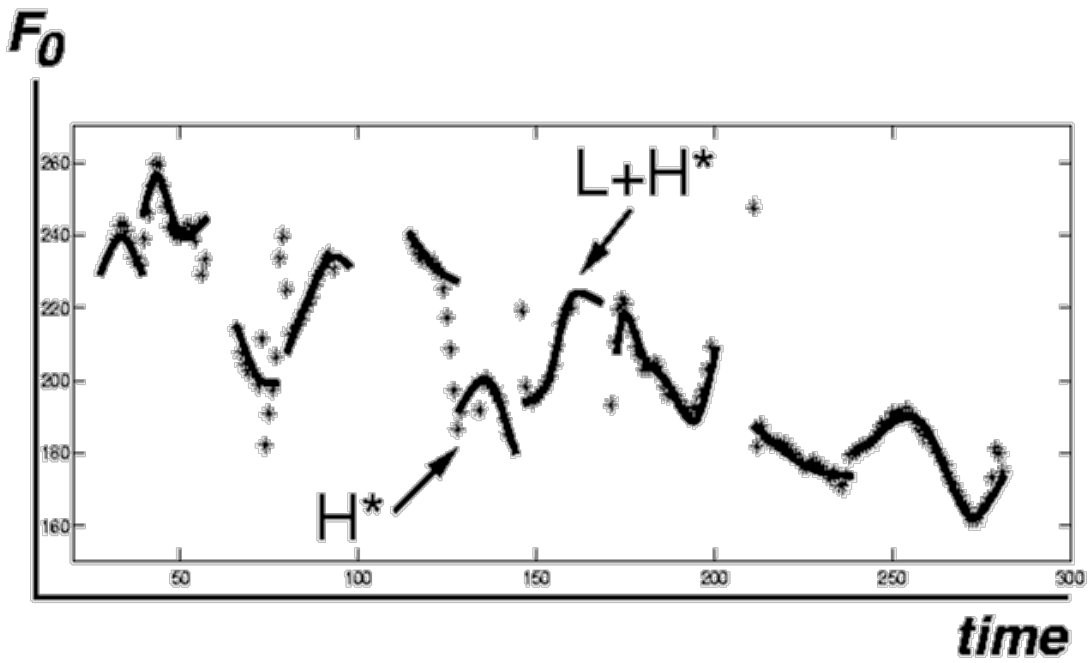
Final accent inventory **L***, **H***, **L*+H**, **L+H***, **H+L***

Final boundary tones: **L%**, **H%**

Additionally, words are given a **break index** of 0-4, to mark strength of boundary

16

How does ToBI help



ToBI gives us a stylised symbolic representation suitable for computation

17

Waveform generation

Now we have got:

- Sequence of phonemes (from which we can get diphones)
- F0 and duration for all phonemes

All that remains is to concatenate the recorded speech units and impose the required f0 and duration using signal processing

This is called **waveform generation**

18

TD-PSOLA duration and f0 modification

- Modify duration by duplicating or deleting pitch periods
- Modify f0 by changing the spacing between pitch periods

In practice, the pitch periods are windowed to give smooth joins (we in fact deal with a window length of two pitch periods)

We also have to compensate by adding or deleting pitch periods when modifying f0

19

Linear predictive synthesis

- An alternative to time-domain PSOLA
- Still a concatenative method
  - Uses a recorded database
- Overcomes some problems of TD-PSOLA
- Widely used
  - E.g., Festival
- With a few tweaks, can sound very good

20

Speech Synthesis

Unit selection

Simon King

Simon.King@ed.ac.uk

Centre for Speech Technology Research, Informatics Forum

Unit Selection

Unit selection speech synthesis is all about deciding **what to record**:

- what speech sounds do we need?
- in other words, what are the basic units of speech, from which we could construct any utterance?

and then **how to use** that recorded speech:

- search the recorded database for the best sequence of speech sounds
- join the units together

22

Units of speech

- It is convenient to think about speech as a linear sequence of units
  - enables a concatenative approach to speech synthesis
  - in speech recognition, allows us to string together models of small units (e.g. phonemes) to make models of larger units (e.g. words)

23

Units of speech

- But the speech signal we observe (the waveform) is the product of interacting processes operating at different time scales
  - at any moment in time, the signal is affected not just by the current phoneme, but many other aspects of the ***context*** in which it occurs
- How can we reconcile this conflict, when we want to simultaneously:
  1. model speech as a simple string of units
  2. take into account all the long-range effects of context, from both before and after the current moment in time

24

Context is the key

- Context-dependent units offer a solution
- We can engineer our system in terms of a simple linear string of units
- We then account for context by having a different version of each unit for every ***different context***

- But, how do we know what all the different contexts are?

25

Context

- If we enumerate all possible contexts, they will be practically infinite
  - there are an infinite number of different sentences in a language
  - context potentially spans the whole sentence (or further)
- However, what is important is the **effect** that the context has on the current speech sound
  - now, we can think about reducing the number of different contexts

26

Some contexts are (nearly) equivalent

- This is the key to unit selection
  - and to HMM-based synthesis too, as we will see later
- We cannot record and store a different version of every speech sound in every possible context
  - but we can have each speech sound in a variety of different contexts

27

Unit selection speech synthesis

Record a database of speech with each diphone in many different contexts

At synthesis time, search for the most appropriate sequence of diphones to use

(other unit types are possible, such as half phones; we may come back to this later)

28

Unit selection

- At synthesis time, if we can't find the speech sound from a precisely matching context, then choose a version of that sound from a **similar** context
  - in other words, a context that will have a **similar effect** on the sound
- For example:
  - can't find "phrase-final [a] in the context [n]_[t]"
  - choose "phrase-medial [a] in the context [m]_[d]"

29

Labelling the database

Can get phone labels from forced alignment using ASR techniques - this works very well.

Phone labels are not enough, we also need to know

- prosodic factors, syllable structure, etc.

because these are important context factors

- i.e., they affect the speech sounds

Use the TTS front-end to predict this from the text

- won't exactly match the speech
- but prosodic labelling from speech signals typically has low accuracy

30

Using the database for synthesis

At synthesis time we must produce speech corresponding to the input text

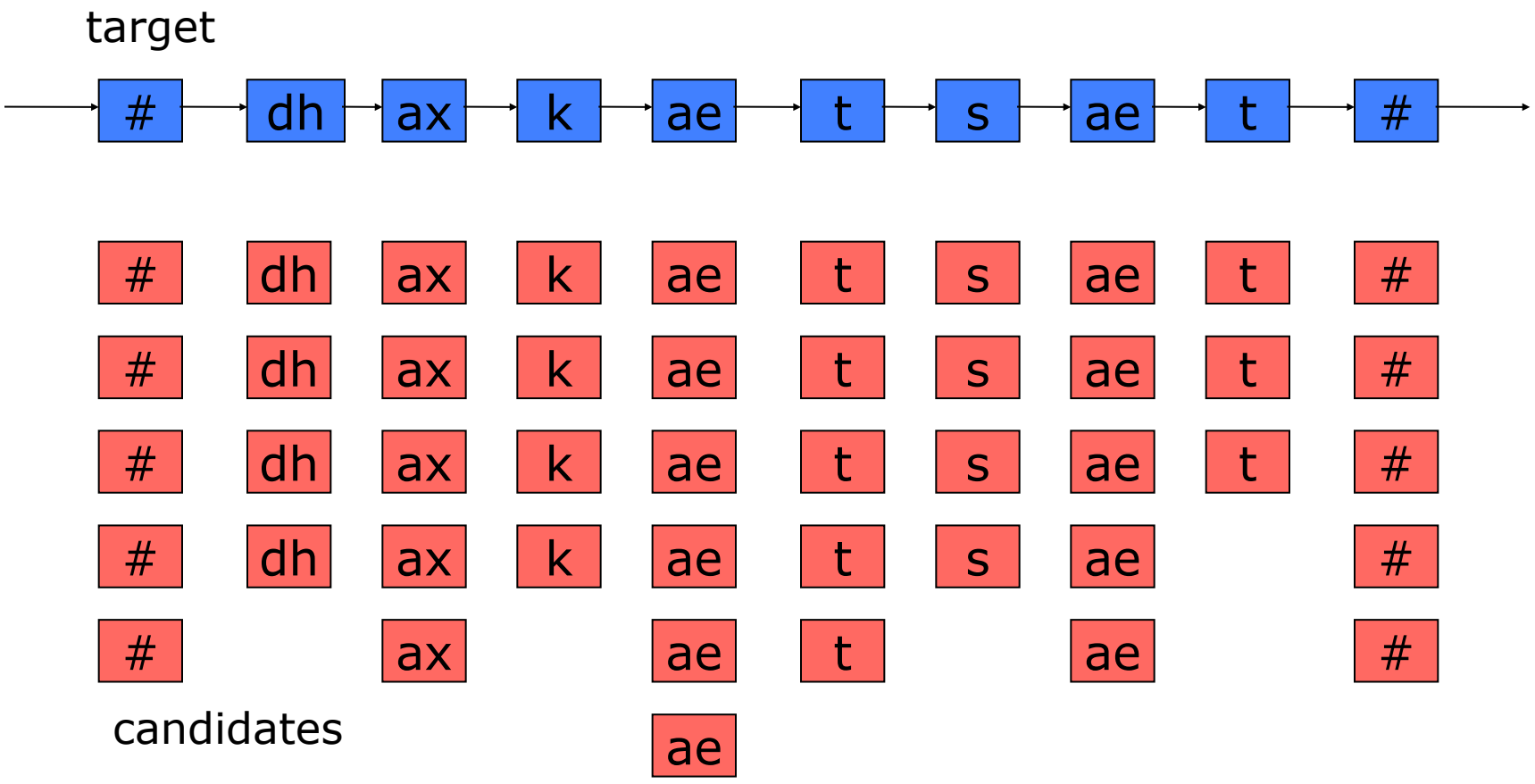
- find the most suitable sequence of speech units from the database
- concatenate them to produce the output speech

First we carry out linguistic analysis

- results in a structured linguistic annotation on the input text, similar to that on the database utterances

31

Target sequence and candidates



32

What are the criteria for selecting units?

- In general, there will be more than one possible unit sequence that could be chosen
- Need to
  - define the criteria for choosing the best sequence
  - find this sequence, from amongst all possible sequences

33

Linguistic criteria

- The ideal unit sequence would comprise units taken from identical linguistic contexts to those in the sentence being synthesised
  - of course, this will not be possible in general, so we must use less-than-ideal units from non-identical (i.e., mismatched) contexts
  - need to quantify how close to ideal they are, so we can choose amongst them
- The mismatch between the linguistic context of a candidate unit and the ideal (i.e., target) context is measured by the **target cost**

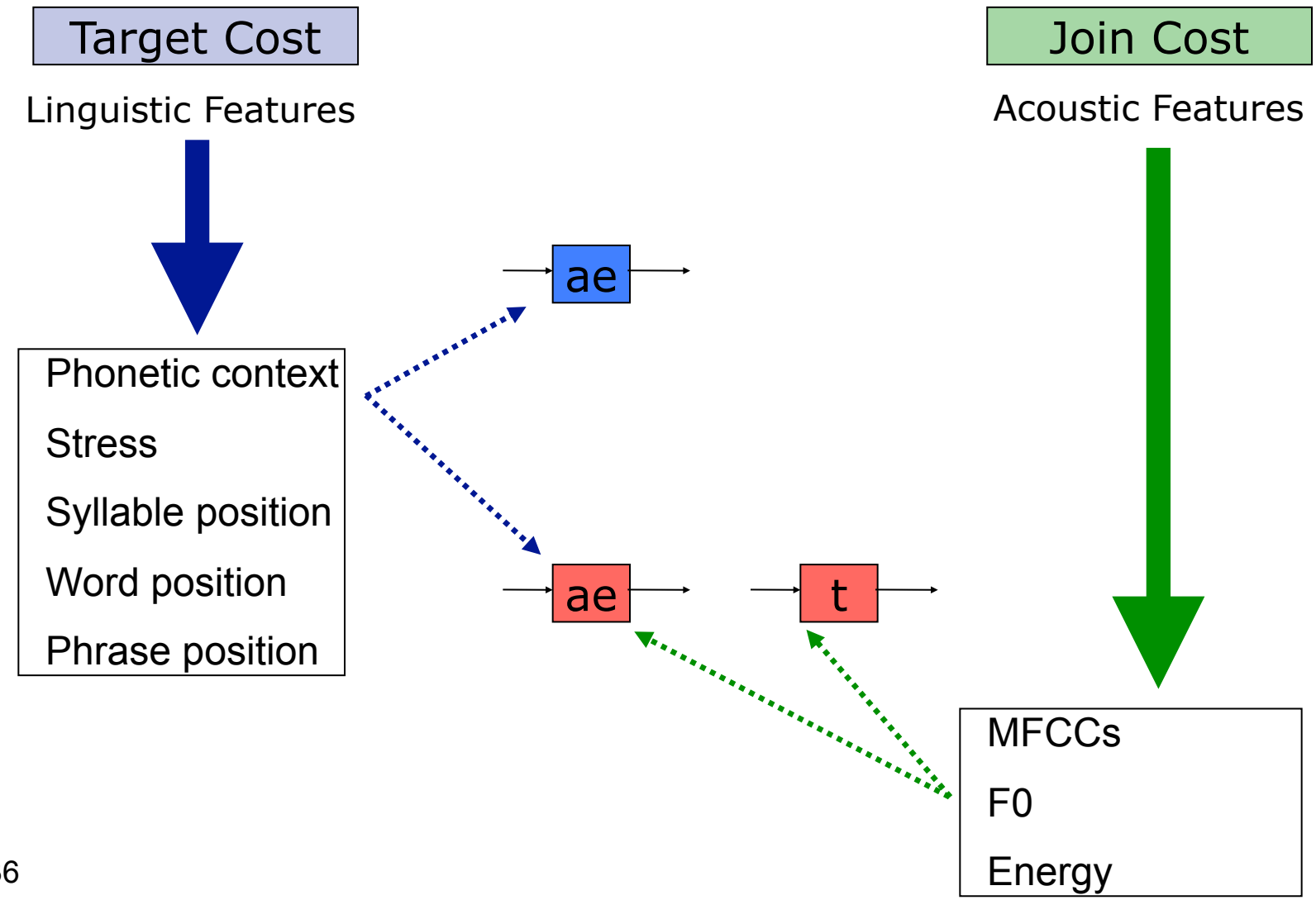
34

Acoustic criteria

- After units are taken from the database, they will be joined (concatenated)
  - Cannot simply join two fragments of speech and hope that it will sound OK because it won't !
- Why? Because of mismatches in acoustic properties around the join point, such as
  - differences in the spectrum, F0, or energy
- The acoustic mismatch between consecutive candidate units is measured by the **join cost**

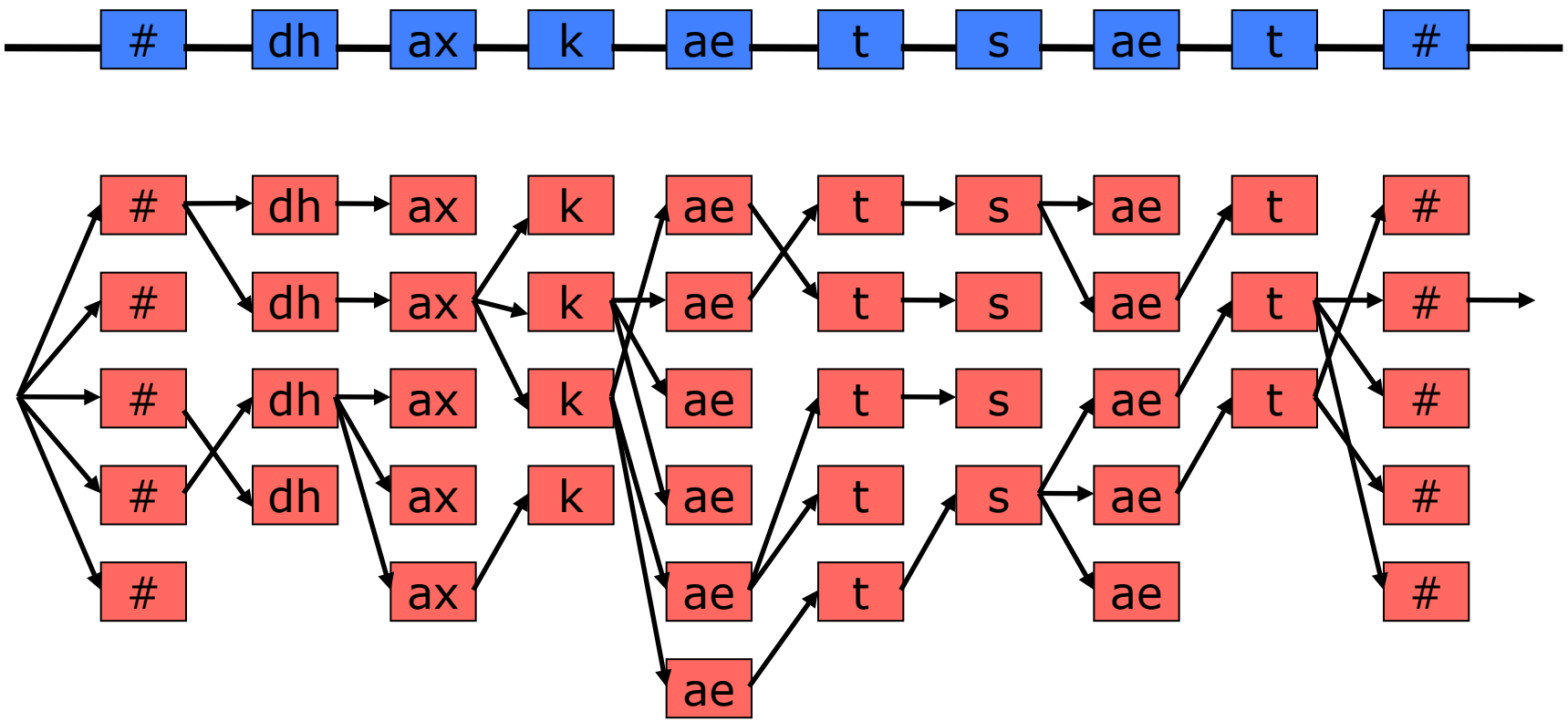
35

Target cost and join cost



36

Search



37

Some examples

- diphone
- clunits (domain specific database)
- unit selection

Optional reading: Alan W Black and Paul Taylor. AUTOMATICALLY CLUSTERING SIMILAR UNITS FOR UNIT SELECTION IN SPEECH SYNTHESIS. Proc. Eurospeech 1997

38

Speech Synthesis

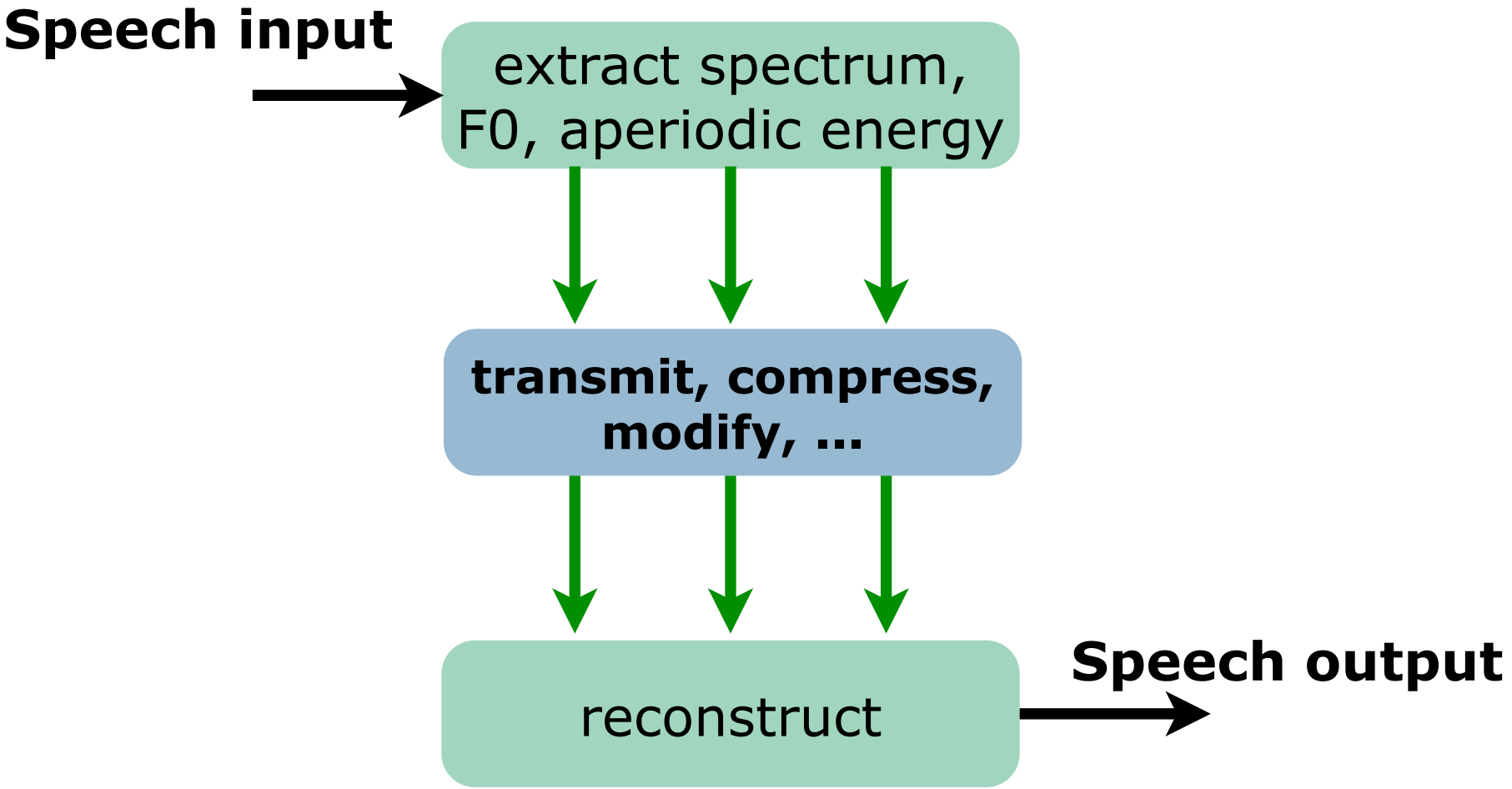
Speech coding and parameter driven speech synthesis

Simon King

Simon.King@ed.ac.uk

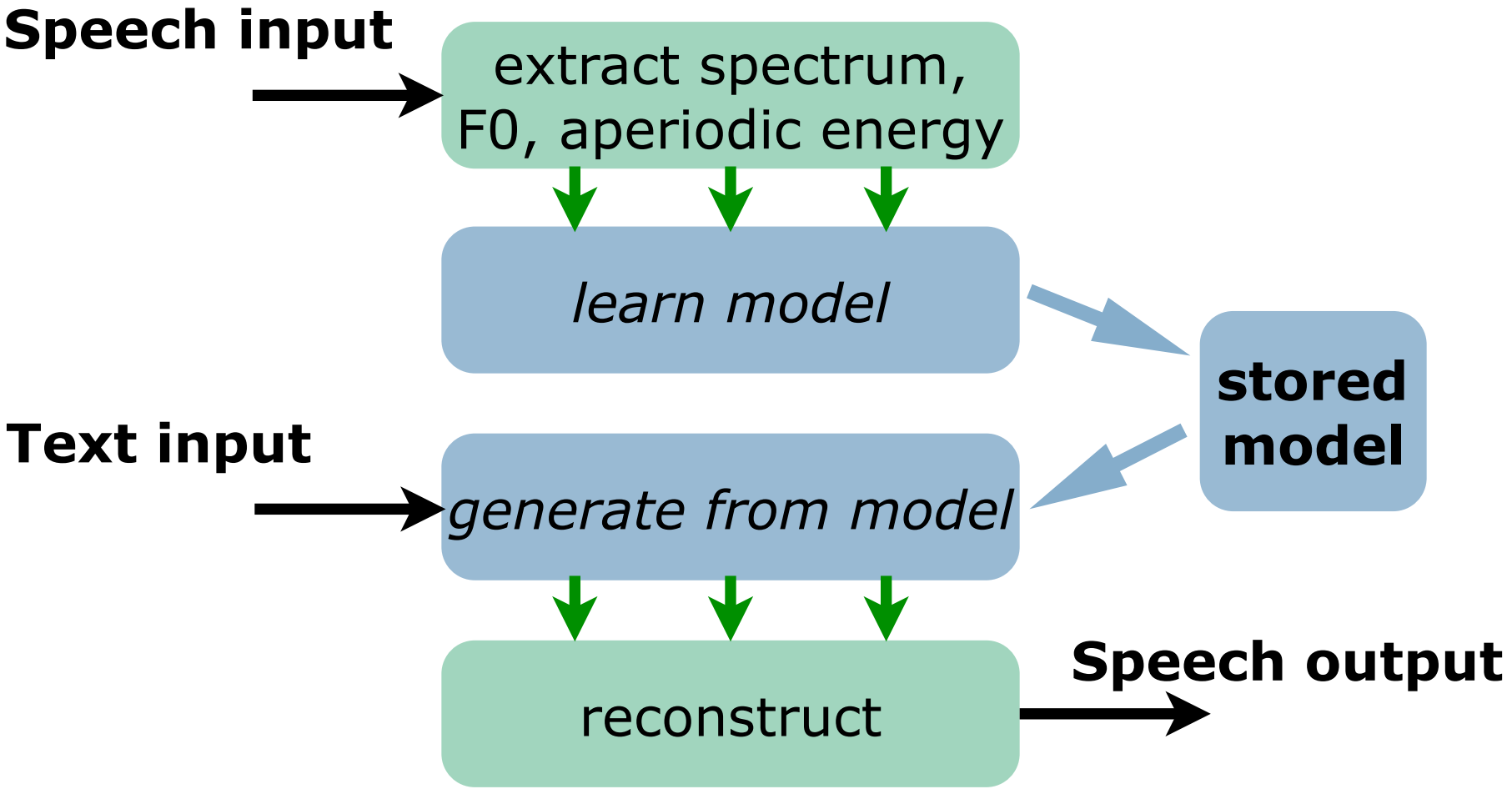
Centre for Speech Technology Research, Informatics Forum

From speech coding to speech synthesis



40

From speech coding to speech synthesis



41

The STRAIGHT vocoder
(Kawahara)

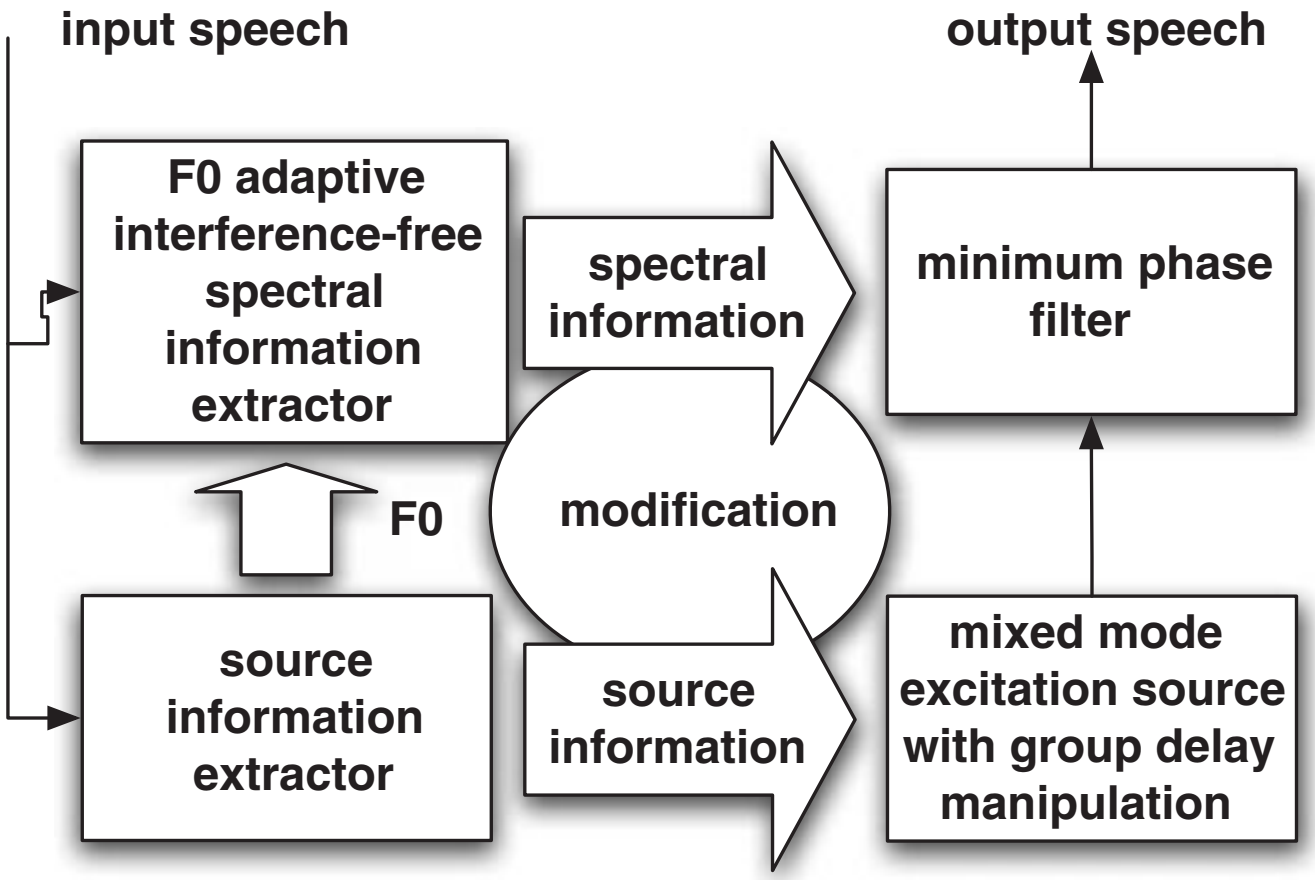


Figure taken from Banno et al, "Implementation of Realtime STRAIGHT", Acoust. Sci. & Tech. 28, 3 (2007)

42

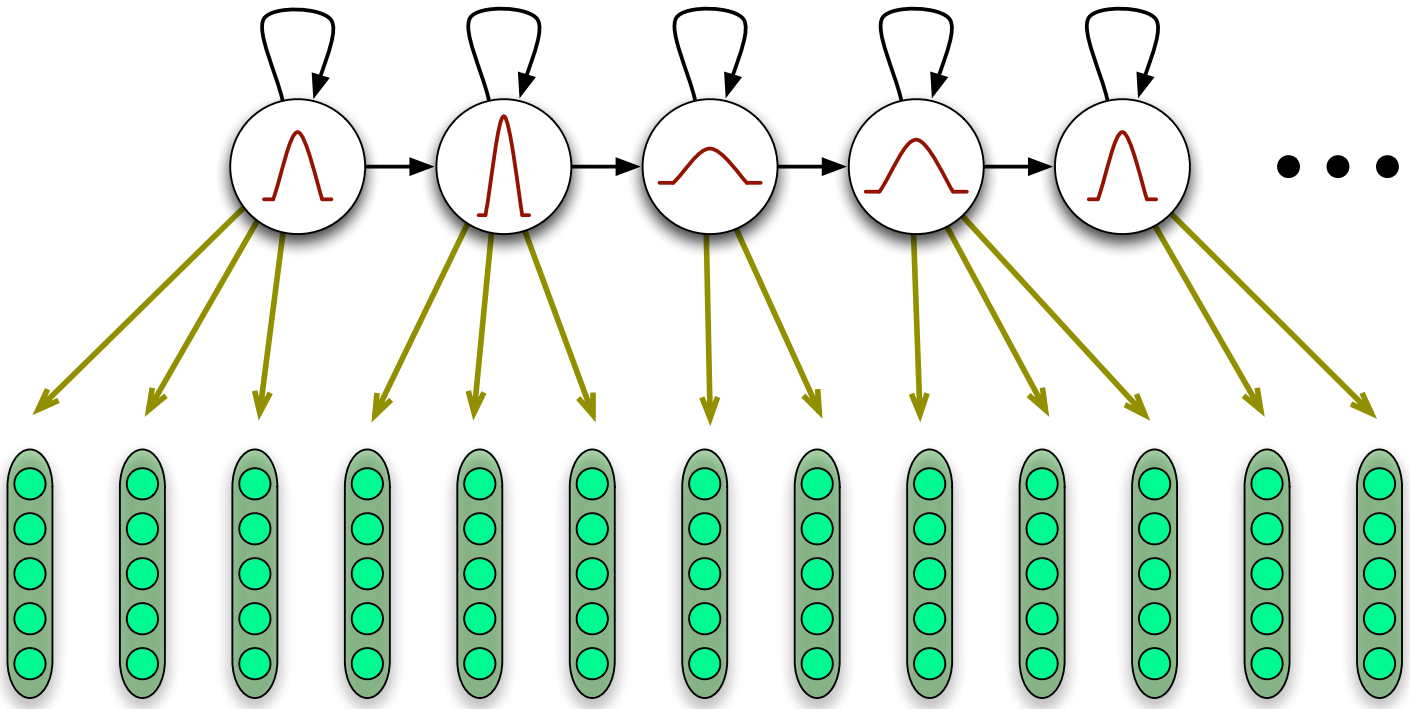
Speech synthesis using HMMs

- Text to speech
  - *input:* text
  - *output:* a waveform that can be listened to

- Two main components
  - *front end:* analyses text and converts to linguistic specification
  - *waveform generation:* converts linguistic specification to speech

43

HMMs are generative models



44

Generating speech from a HMM

- HMMs are used to generate sequences of speech (in a **parameterised form** that we call ‘speech features’)
- From the **parameterised form**, we can generate a waveform
- The **parameterised form** contains sufficient information to generate speech:
  - spectral envelope
  - fundamental frequency (F0) - sometimes called ‘pitch’
  - aperiodic (noise-like) components (e.g. for sounds like ‘sh’ and ‘f’)

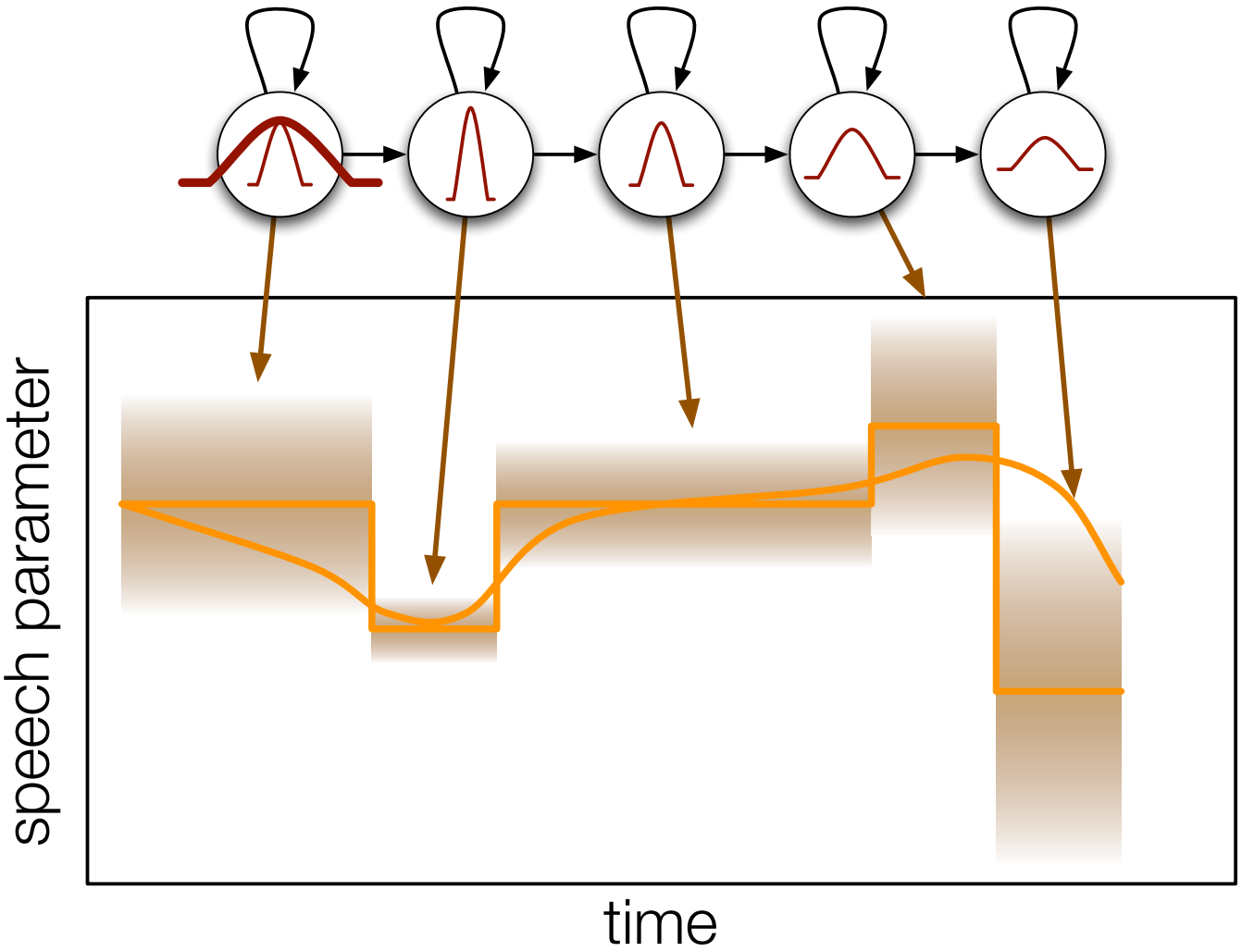
45

Trajectory generation from HMMs

- Using an HMM to generate speech parameters
  - because of the Markov assumption, the most likely output is the sequence of the means of the Gaussians in the states visited
  - this is piecewise constant, and ignores important dynamic properties of speech
- Maximum Likelihood Parameter Generation (MLPG) algorithm
  - solves this problem, by correctly using statistics of the dynamic properties during the generation process

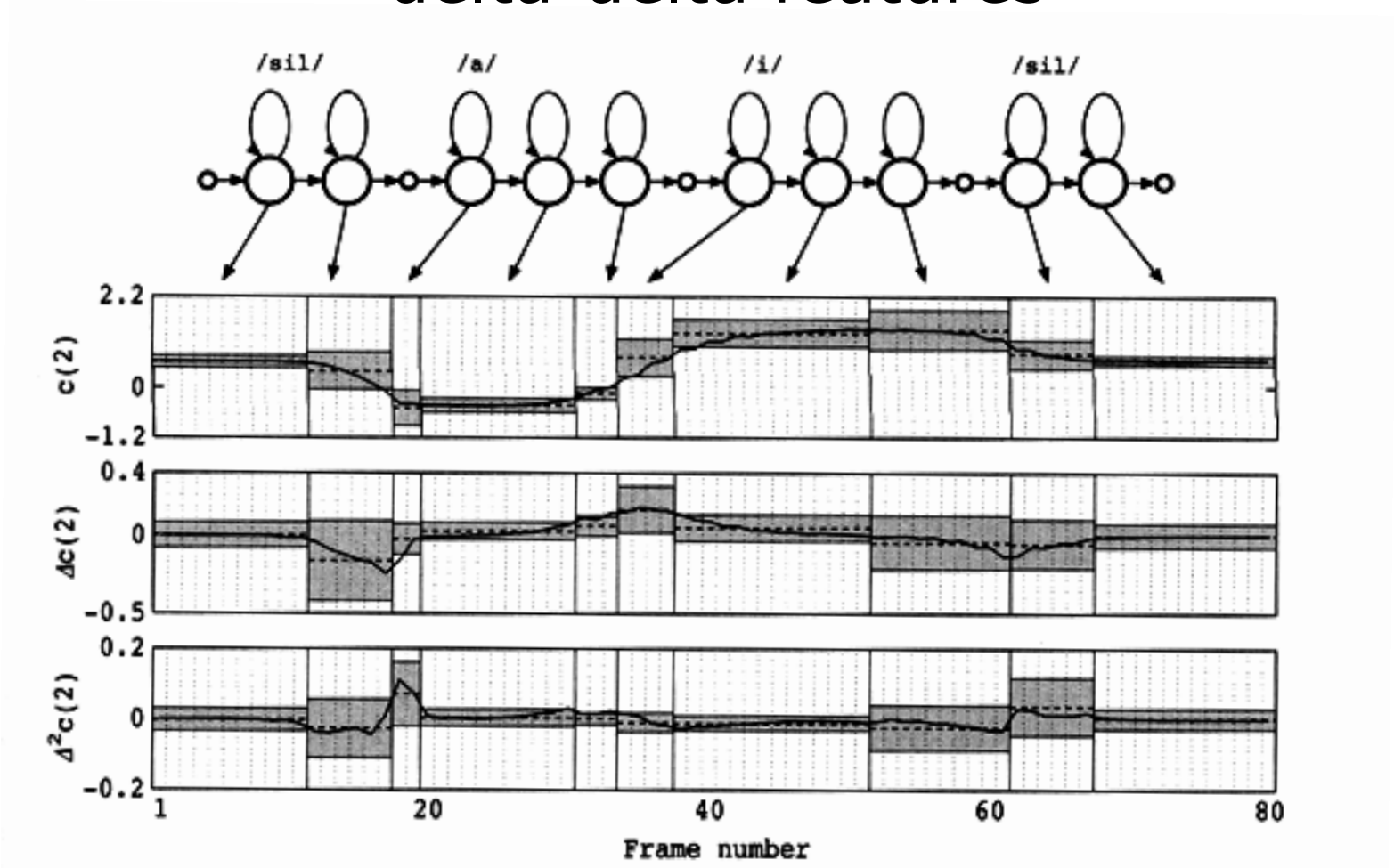
46

Maximum Likelihood Parameter Generation



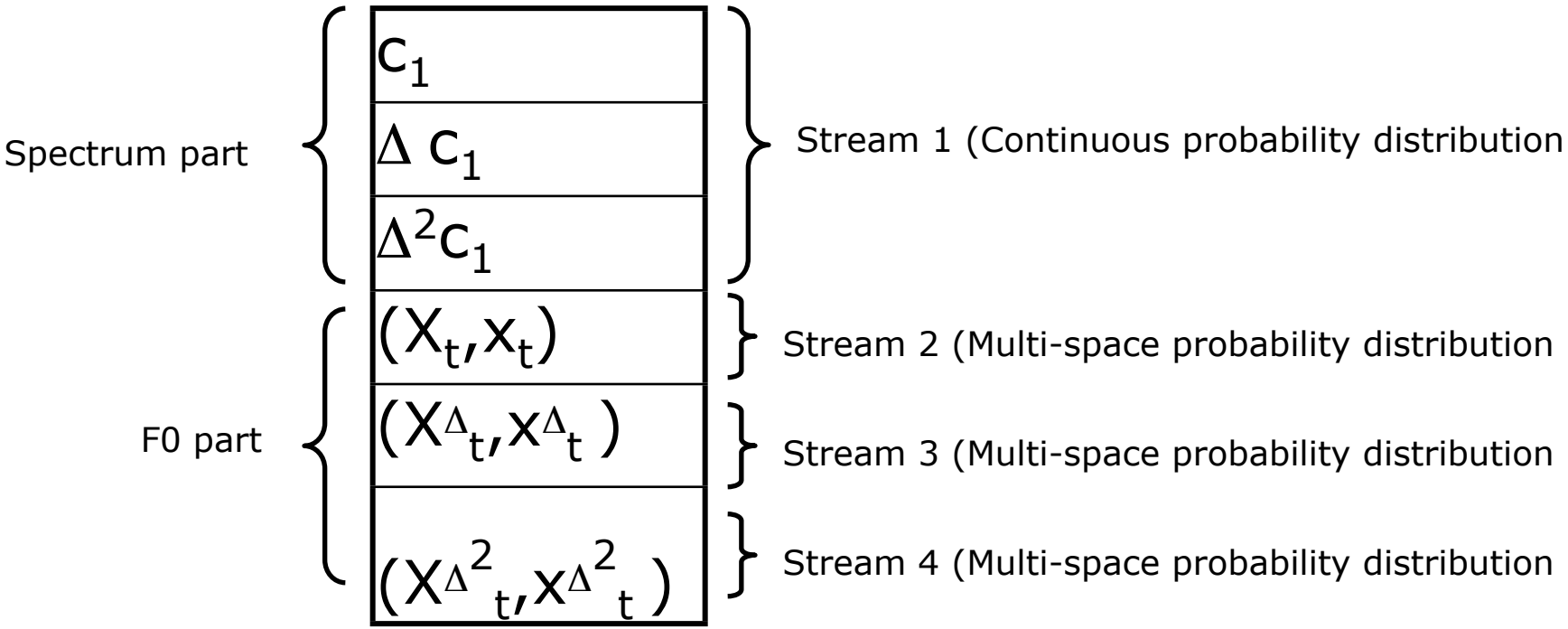
47

Using the statistics of the delta and delta-delta features



48

HMM output vectors



Models additionally have state duration densities to model duration

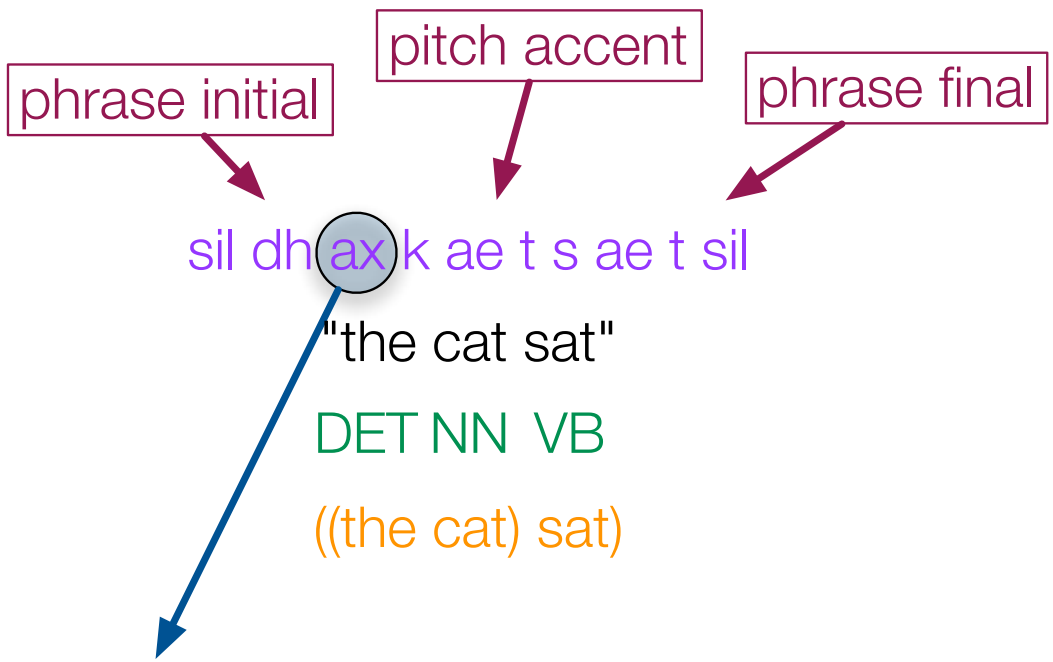
49

Constructing the HMM

- Linguistic specification (from the front end) is a sequence of phonemes, annotated with contextual information
- There is one 5-state HMM for each phoneme, in **every required context**
- To synthesise a given sentence,
  - use front end to predict the linguistic specification
  - concatenate the corresponding HMMs
  - generate from the HMM

50

From words to linguistic specification



51

Example linguistic specification

```
pau^pau-pau+ao=th@x_x/A:0_0_0/B:x-x-x@x-x&x-x#x-x$.....
pau^pau-ao+th=er@1_2/A:0_0_0/B:1-1-2@1-2&1-7#1-4$.....
pau^ao-th+er=ah@2_1/A:0_0_0/B:1-1-2@1-2&1-7#1-4$.....
ao^th-er+ah=v@1_1/A:1_1_2/B:0-0-1@2-1&2-6#1-4$.....
th^er-ah+v=dh@1_2/A:0_0_1/B:1-0-2@1-1&3-5#1-3$.....
er^ah-v+dh=ax@2_1/A:0_0_1/B:1-0-2@1-1&3-5#1-3$.....
ah^v-dh+ax=d@1_2/A:1_0_2/B:0-0-2@1-1&4-4#2-3$.....
v^dh-ax+d=ey@2_1/A:1_0_2/B:0-0-2@1-1&4-4#2-3$.....
```

“Author of the ...”

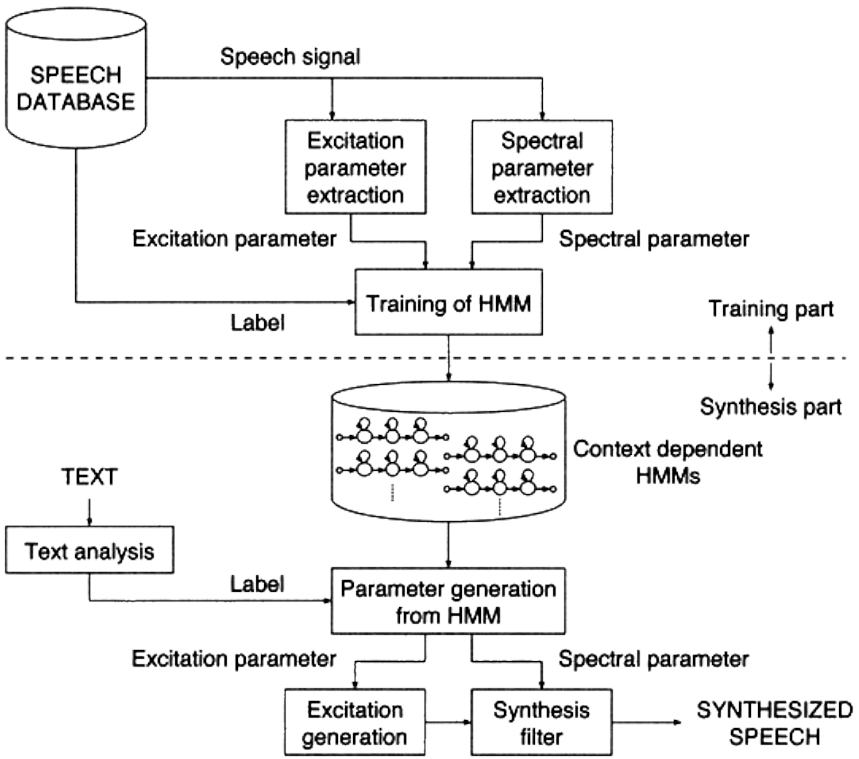
52

Comparison with ASR

- Differences from automatic speech recognition include
  - Synthesis uses a much richer model set, with a lot more context
    - For speech recognition: triphone models
    - For speech synthesis: “full context” models
  - “Full context” = both phonetic and prosodic factors
  - Observation vector for HMMs contains the necessary parameters to generate speech, such as spectral envelope + F0 + multi-band noise amplitudes

53

HMM-based speech synthesis



from: *An HMM-based approach to multilingual speech synthesis*, Tokuda, Zen & Black, in Text to speech synthesis: New paradigms and advances; Prentice Hall: New Jersey, 2004

54

End of this part

55