But diphone synthesis is too restricted

- ☐ Phonetic phonomena go over more than two phones
- ☐ Phone-only systems ignore:
  - prosody, stress, syllable position etc
- ☐ Two directions:
  - Larger DB
  - More natural DB

Larger database

- ☐ triphones:
  – where it matters
- ☐ stress, onset/coda
- ☐ demi-syllables:
  – approx 10K syls in English

Gives larger, more carefully constructed db:
– more difficult to collect

More natural database

- natural speech has natural coverage:
  - lots of examples of common combinations
  - few examples or rare ones
- Should be good for synthesis, if:
  - has basic coverage
  - you can find appropriate units

Why automatic unit selection

- Carefully designed dbs:
  - speaker makes errors
  - speaker doesn't speak intended dialect
  - require db design to be right
- If its automatic:
  - labelled with what was actually said
  - flaps, schwas, coarticulation is natural
- Can better model speaker:
  - want the system to sound like Walter Cronkite
  - picks up ideolect of speaker

Unit selection synthesis systems

Selecting appropriate units from natural speech

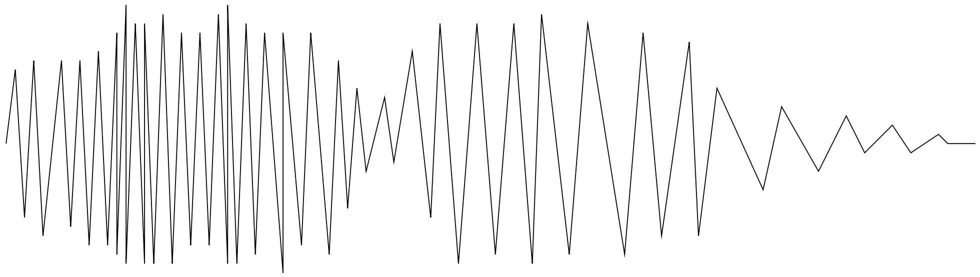
- nuu-talk (non-uniform units):
  - – ATR, Japanese only
  - – 503 sentences "balanced"
  - – acoustic selection only
- CHATR:
  - – Multi-language
  - – Uses prosody (and general features)
- Acuvoice:
  - – first commercial unit selection system
- AT&T's NextGen, SpeechWorks' Speechify:
  - – CHATR/Festival based
- Lernout & Houspie's RealSpeak:
  - – Phonological structure with exception rules
- Others:
  - – Rhetorical, Cepstral, Loquendo.

Unit selection synthesis algorithms

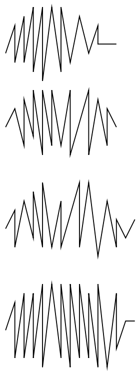
- Hunt and Black 96:
  - CHATR and NextGen
  - estimate target cost of units
- Clustering
  - Donovon and Woodland 95/Black and Taylor 97
  - Microsoft Whisper, Festival/clunits
  - group acoustically similar units
- Phonological Structure Matching
  - Taylor and Black 99
  - Festival/PSM
  - Index through trees
  - BT Laureate (Breen et all 98) similar

Selecting a candidate

Synthesis Target



Database Candidates



Selection criteria

- ☐ Phonetic context (alone):
  - – assumes that phonological information is sufficient
  - – assumes dbs is pronounced properly
- ☐ Automatic acoustic measure:
  - – do these two units sound the same
  - – why context makes them different
  - – how suitable is this acoustic unit for this context

Acoustic cost: measuring good synthesis

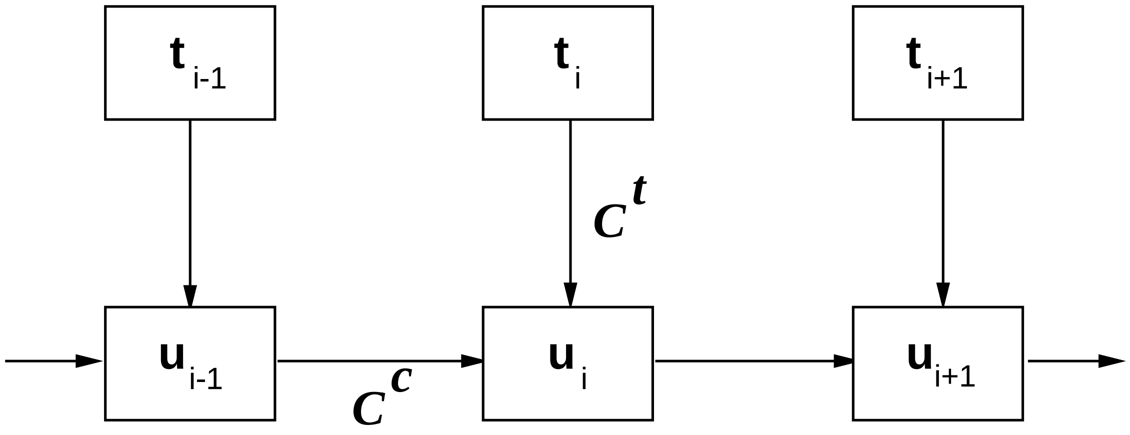
Given a selected set of units how well do they match the original?

Best phonetic context, least F_0 difference?
– NO, these are too indirect
– they assume that phonology defines acoustics
Cepstral distance? (traditionally used)
– we use Mel Frequency cepstrum, F_0, power
– pitch schronous, delta cepstrum
– some other parameterisation
– penalty for duration mismatch

Ideally:
– acoustic measure follows human perception

Basic selection model

Find candidate units
Find best selection through theses options



HB96: acoustic distance

What is the similarity between two pieces of speech:

- ☐ MEL Cepstrum 12 params
- ☐ F0 (normalized)
- ☐ Duration penalty

– $AC^t(t_i, u_i) = \Sigma_{i=1}^{p} w_i^a abs(P_i(u_n) - P_i(u_m))$
– weights are hand defined

HB96: Estimating acoustic distance

Selection features:
– phone context, prosodic context, and others
Database and target units labelled with those features:
– need weighted distance between feature vectors

Target distance is:
– $C^t(t_i, u_i) = \Sigma_{j=1}^{p} w_j^t C_j^t(t_i, u_i)$
For examples *in the database* we can measure
– $AC^t(t_i, u_i)$
Therefore estimate w_{1-j} from all examples of
– $AC^t(t_i, u_i) \approx \Sigma_{j=1}^{p} w_j^t C_j^t(t_i, u_i)$
Use linear regression

HB96: Weight Training

Collect phones in classes of acceptable size
– e.g. stops, nasals, vowel classes etc
Find AC^t between all of same phone type
Find C^t between all of same phone type
Estimate w_{1-j} using linear regression.

Space and time complexity n^2 on units in class.

HB96: Continuity cost

How well does it join:
– $C^c(u_{i-1}, u_i) = \Sigma_{k=1}^{p} w_k^c C_k^c(u_{i-1}, u_i)$
– if $(u_{i-1} == \text{prev}(u_i))$ $C^c = 0$

Used:
– quantised melcep features
– local F0
– local absolute power
– Hand tuned weights

Can vary position of joins too (optimal coupling)

HB96: Using the results

We now have weights (per phone type) for features set between target and db units.

Find best path of units through db that minimise:

$$C(t_1^n, u_1^n) = \Sigma_{i=1}^n C^t(t_i, u_i) + \Sigma_{i=2}^n C^c(u_{i-1}, u_i) + C^c(S, u_1) + C^c(u_n, S)$$

Standard problem solvable with Viterbi search with beam width constraint for pruning.

DW95: Clustering HMM states

- Label databases of speech with HMM
- Use acoustic measure to find distance between states:
  – weighed cepstrum distance
- Use CART to index into clusters:
  – use TTS available features

- DW95 produced only one target candidate

BT97: Acoustic distance

mean weighted Euclidean distance between frames To find most similar units define acoustic distance between two units of the same type U, V

$$Adist(U,V) = \begin{cases} \text{if } |V| > |U| & Adist(V,U) \\ \frac{WD*|U|}{|V|} * \sum_{i=1}^{|U|} \sum_{j=1}^{n} \frac{W_j.(abs(F_{ij}(U) - F_{(i*|V|/|U|)j}(V)))}{SD_j * n * |U|} \end{cases}$$

$|U|$ = number of frames in U
$F_{xy}(U)$ = parameter y of frame x of unit U
SD_j = standard deviation of parameter j
W_j = weight for parameter j
WD = duration penalty

Frames include: F_0, 12 MFCC, Energy, delta MFCC

BT97: Making clusters

Classification and Regression Trees (Breiman84)

Impurity(Cluster) = mean acoustic distance between members

$$Impurity(C) = \frac{1}{|C|^2} * \sum_{i=1}^{|C|} \sum_{j=1}^{|C|} Adist(C_i, C_j)$$

Recursively find best question which splits C such that mean impurity of sub-clusters less than impurity if C.

Questions use:
– phonetic context
– pitch and duration context
– Syllable position, stress, accent
– Position in phrase

i.e. features that exist at synthesis time

```
(w
 ((p.name is #)
  ((duration < 0.0394)
   ((((10 26 31 49 50 55 61 85 89 90 103 233))))
   ((((1 24 86 92 96 124 127 129 131 144 ...)))))
  ((p.name is n)
   ((((2 12 29 59 66 ...))))
   ((n.name is oo)
    ((((5 8 23 30 33 67 ...))))
    ((p.name is @)
     ((n.ph_vheight is 2)
      ((((13 14 106 ...))))
      ...
```

BT97 plus updates

- Acoustic distance:
  - pitch synchronous MFCC
  - include 50% previous phone (i.e. diphones)
  - not use delta cepstrum
- Pruning:
  - remove units farthest fron center
  - makes db smaller
  - can remove “bad” phones
- Further subclassify phones:
  - as diphones
  - as word/class types

TB99: Phonological Structure Matching

- ☐ Label whole DB as trees:
  – Words/phrases, syllables, phones
- ☐ For target utterance:
  – label it as tree
  – top-down, find subtrees that cover target
  – recurse if no subtree found
- ☐ Produces list of target subtrees:
  – explicitly longer units that other techniques
- ☐ Selects on:
  – phonetic/metrical structure
  – only *indirectly* on prosody

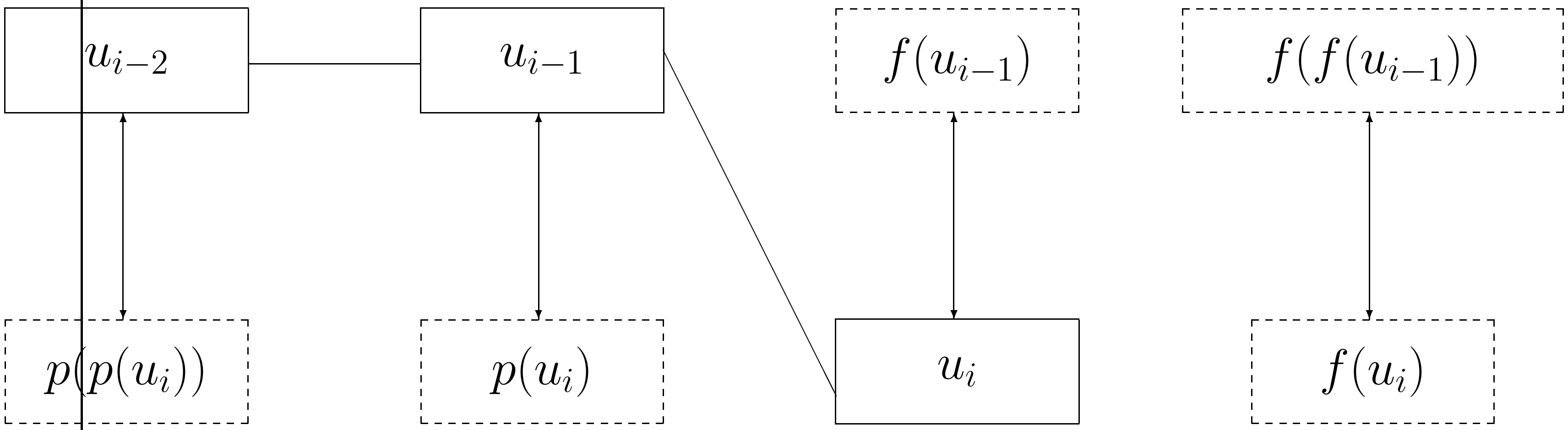
Unit selection comparison

- Hunt and Black 96:
  - acoustic distance estimation
  - expensive target selection
  - easy to hand tune
- Cluster method
  - depends on acoustic distance
  - can overtrain
- Phonological structure matching
  - no acoustic cost
  - selects longer units

All use optimal coupling

Optimal coupling

Where is the best join for two units?
How good is it?



Non-dashed boxes: selected units
Dashed boxes: consecutive units in db
p: a unit's actual previous unit *from the database*
f: a unit's actual following unit

Optimal coupling

How to measure good joins

- F0, power
- Cepstrum (window or single frame)
- Frequency domain
- How does this compare with human views:
  – "randomly" join bunch of units
  – play to subjects and mark "goodness"
  – find automatic measure that corelates with humans

The right type of database

- Synthesized example reflect db type:
  - news data synthesizes as new data
  - news data is bad for dialog
- Natural vs controlled:
  - domain related data
  - phonetically balanced (e.g. timit)
- train prosodic models on database

The right type of speaker

- Professional speakers are always better:
  – consistent style and articulation
  – though these dbs are carefully labelled
- Ideally (and AT&T experiment, Syrdal99)
  – record 20 professional speakers
  – (small amount of data)
  – build simple synthesi examples
  – get many (200?) people to listen and score them
  – take best voices
- Find corelates for human selection:
  – high power in unvoiced speech
  – high power in higher frequencues
  – larger pitch range

The right type of things to synthesis

- Instead of making the db appropriate
- Make the things we synthesize approrpiate
- Domain synthesis:
  – know what is to be said – design the database specifically

Unit selection comments

Advantages

- ☐ Quality is far superior to diphones
- ☐ Even (some) bad joins are better diphone syntheses
- ☐ Natural prosody selection sound better.

Disadvantages

- ☐ Quality can be *very* bad
- ☐ Synthesis is computationally expensive
- ☐ Can't synthesize everything you want:
  - – diphone technique can move emphasis
  - – unit selection gives good (but may incorrect) result

Unit selection vs generation

Selecting actual parts of speech
"Averaging" over examples.

HMM synthesis (Tokuda)

- Use reversible Mel Cepstral Parameterisation
- Cluster HMM states (cf Donovan)
- Use HMM parameters to generate speech
  – *not* to identify particular pieces of speech
- Dynamic parameters smooth F0
- Quality smooth, but buzzy
- Allows voice/style transformation