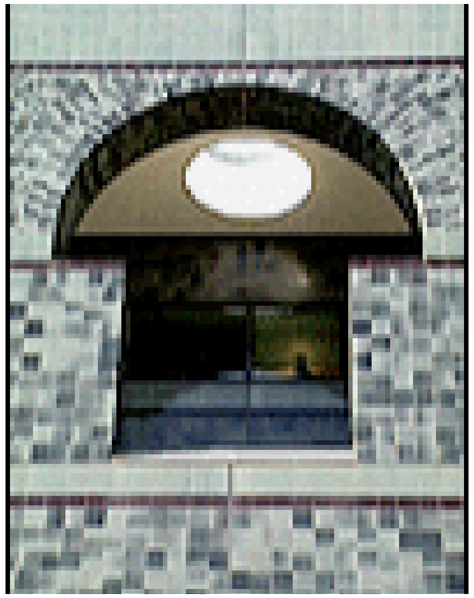
Music 209
Advanced Topics in Computer Music
Lecture 8 – Off-line Concatenation Control

Pre-recorded audio and MIDI performances: we know data for "future" t's.

2006-3-9

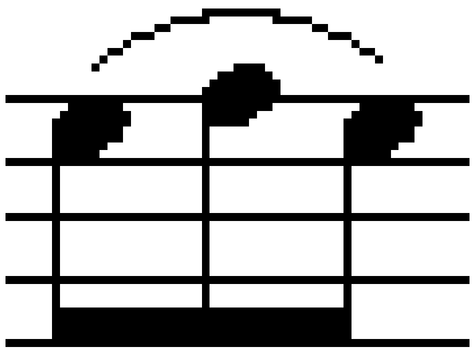


Professor David Wessel (with John Lazzaro)

(cnmat.berkeley.edu/~wessel, www.cs.berkeley.edu/~lazzaro)

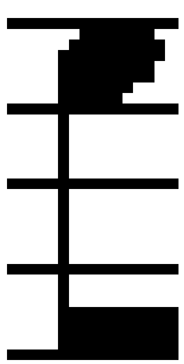
www.cs.berkeley.edu/~lazzaro/class/music209

From Lecture 2: Legato Concatenation

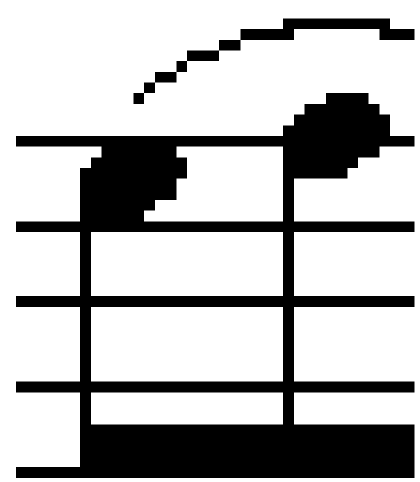


In this lecture, we assume we know all of the notes that follow and precede these three notes, and can use that knowledge to pick the best units to splice.

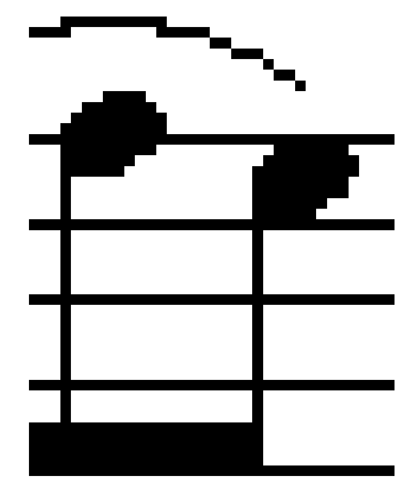
Offline stitching: Use knowledge of all time to pick sample #2 and sample #3.



splice to



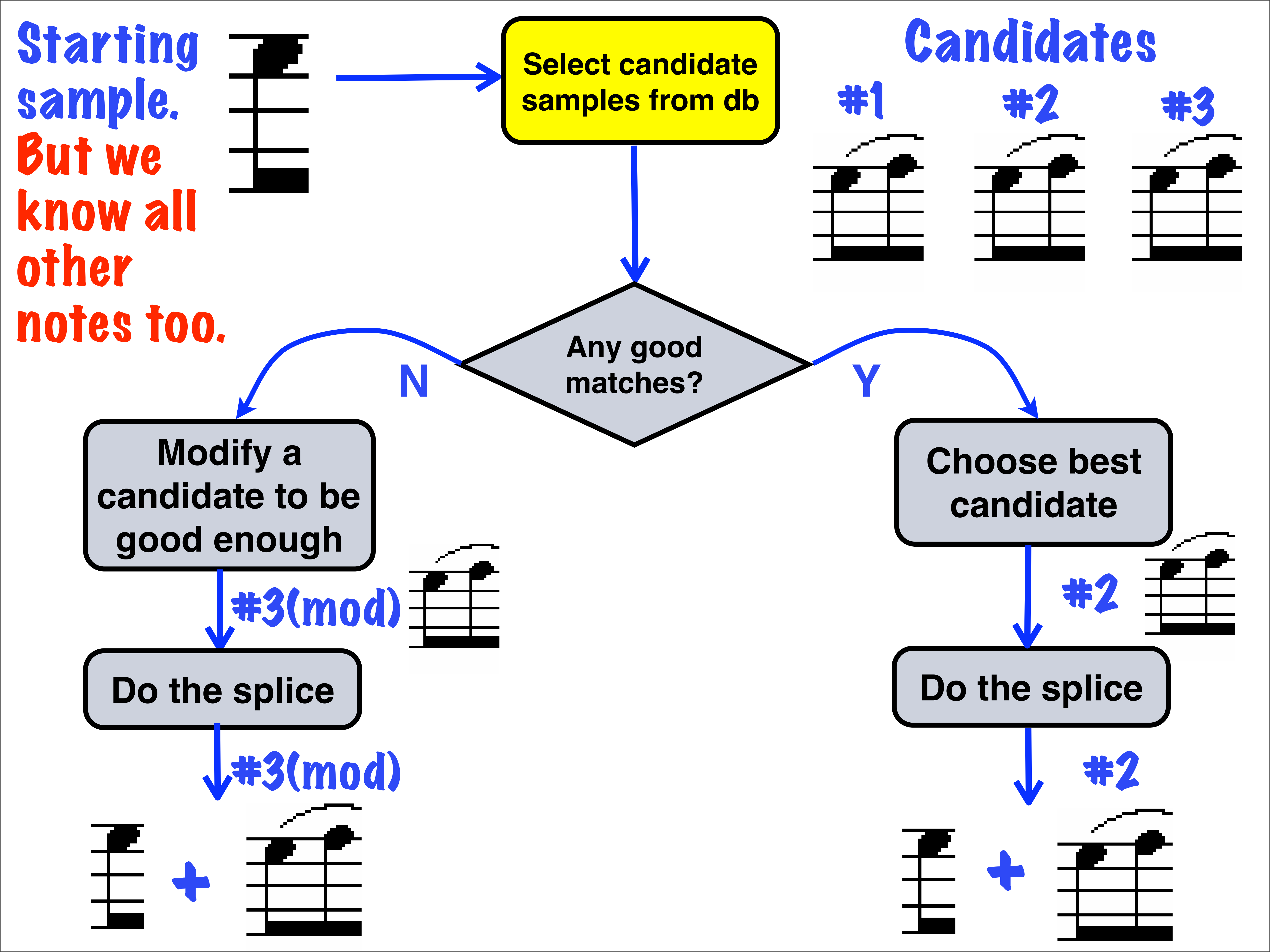
splice to



Sample #1:
isolated E

Sample #2:
E to F interval
played legato

Sample #3:
F to E interval
played legato



Starting sample.
But we know all other notes too.
Select candidate samples from db
Candidates
#1
#2
#3
Any good matches?
N
Y
Modify a candidate to be good enough
#3(mod)
Do the splice
#3(mod)
+
Choose best candidate
#2
Do the splice
#2
+

Topics for today ...

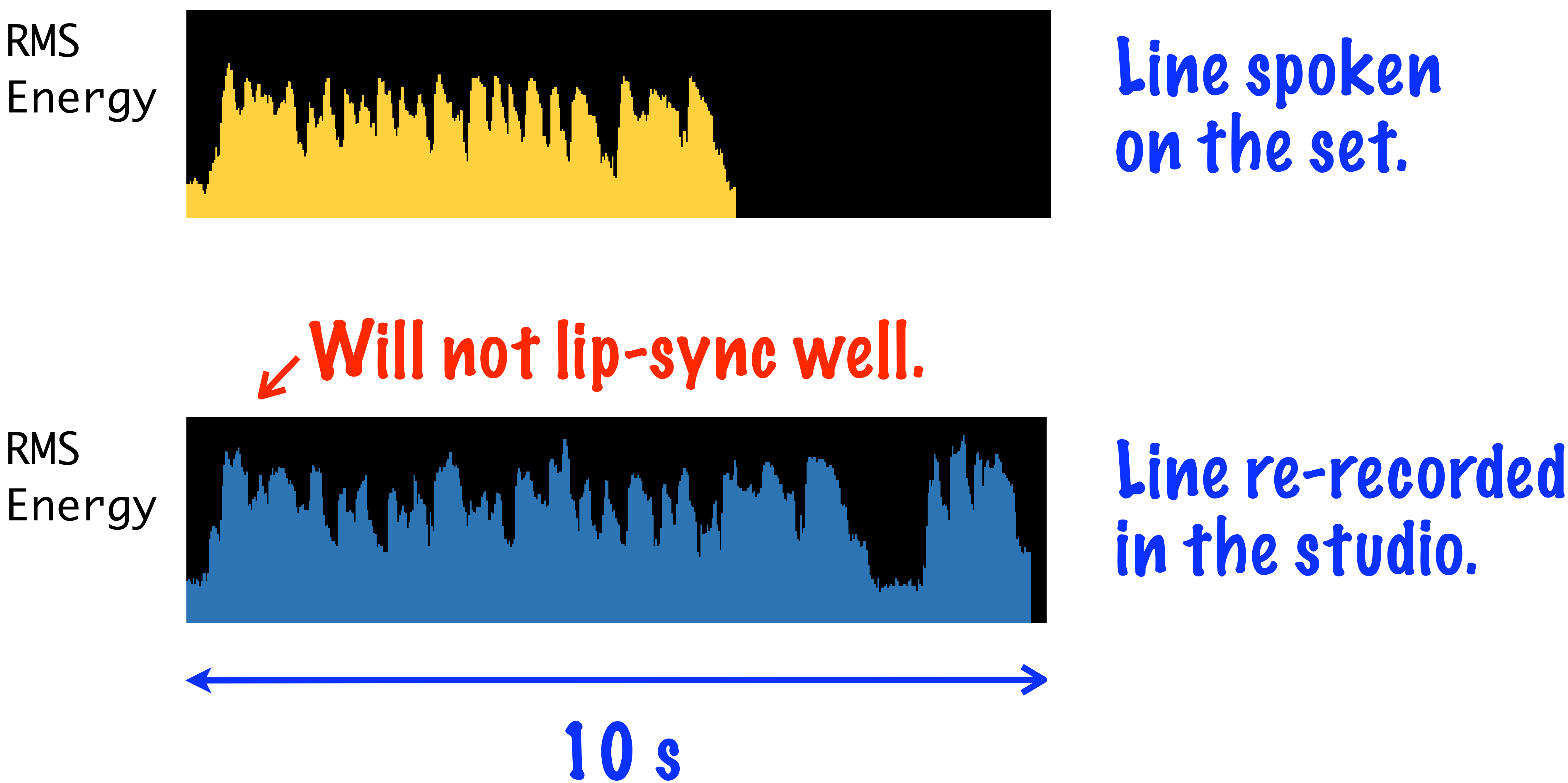
- **The Viterbi Algorithm:** How to choose each unit to get the best fit over an entire piece.
- **Vocalign:** Introductory example to using the algorithm in audio application.
- **Score alignment:** Using the algorithm to populate the database with units.
- **Synthesis:** Using the algorithm to do unit selection for off-line concatenation.

Audio on a movie set ...

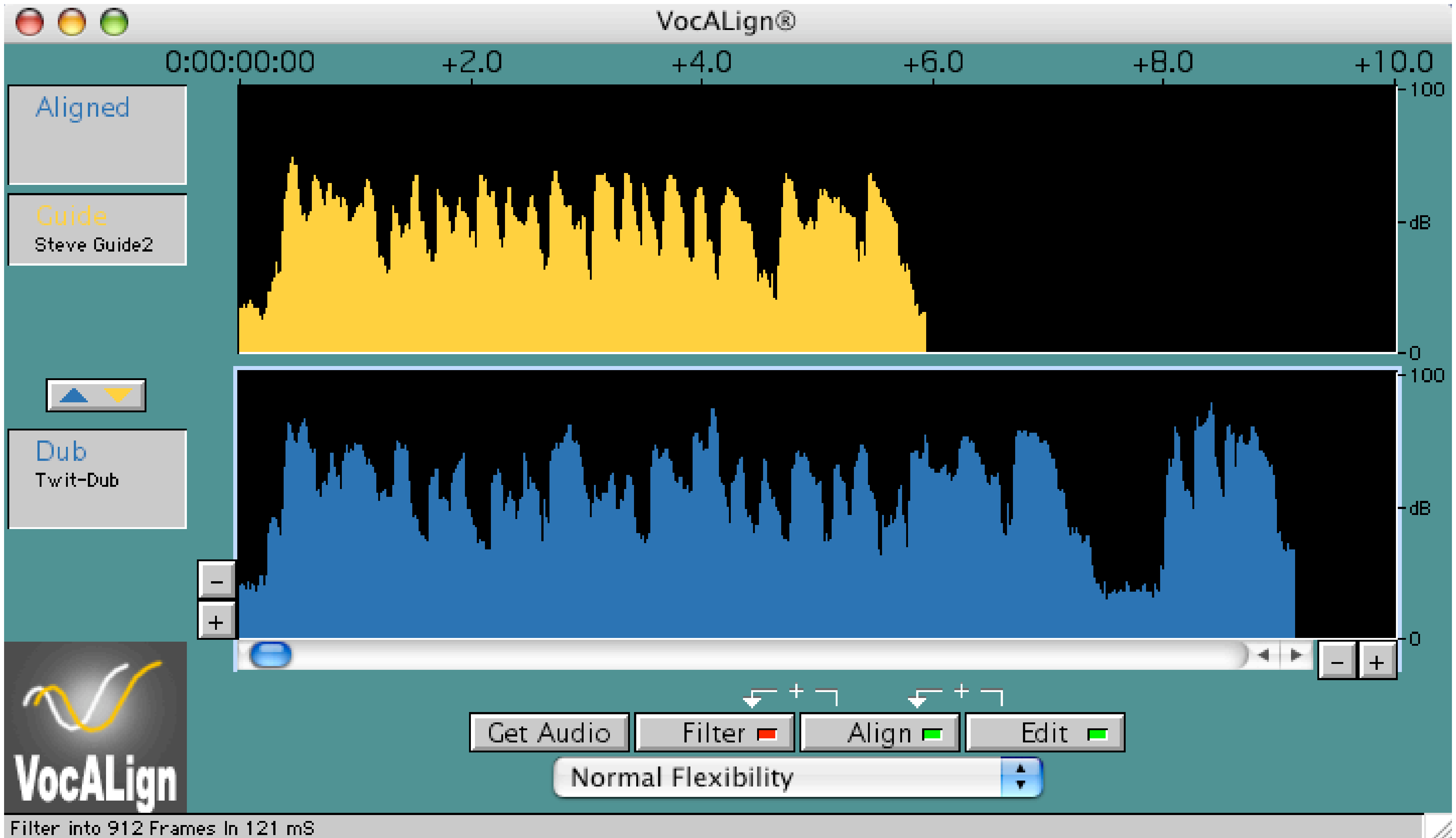
Audio quality may leave something to be desired ...

Redo audio in the studio

Problem: Re-recorded audio must **synchronize** tightly to **visuals** (lip-sync, footsteps, etc).



VocAlign: A plug-in that automatically aligns "dub" audio with "guide" audio.

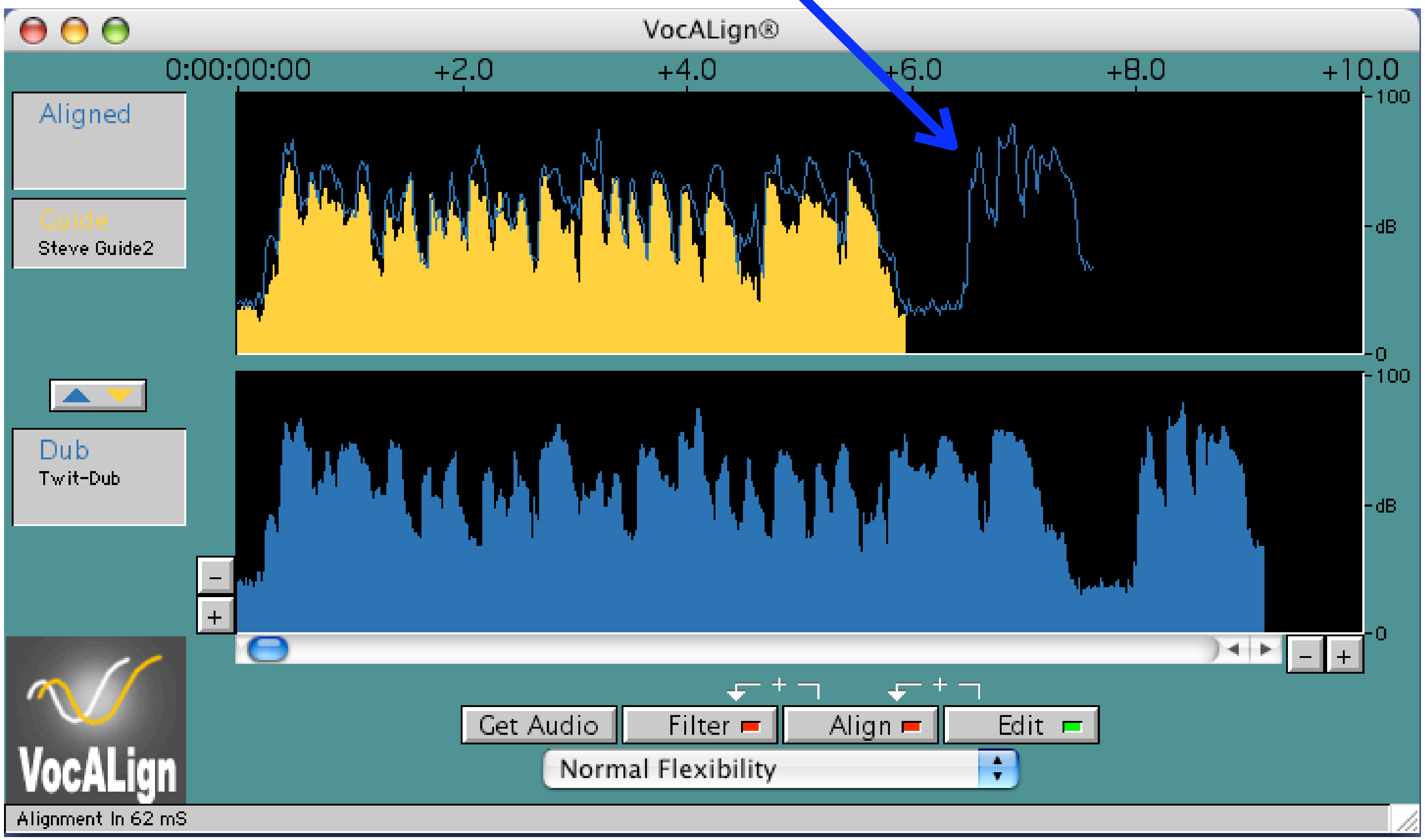


"guide"

"dub"

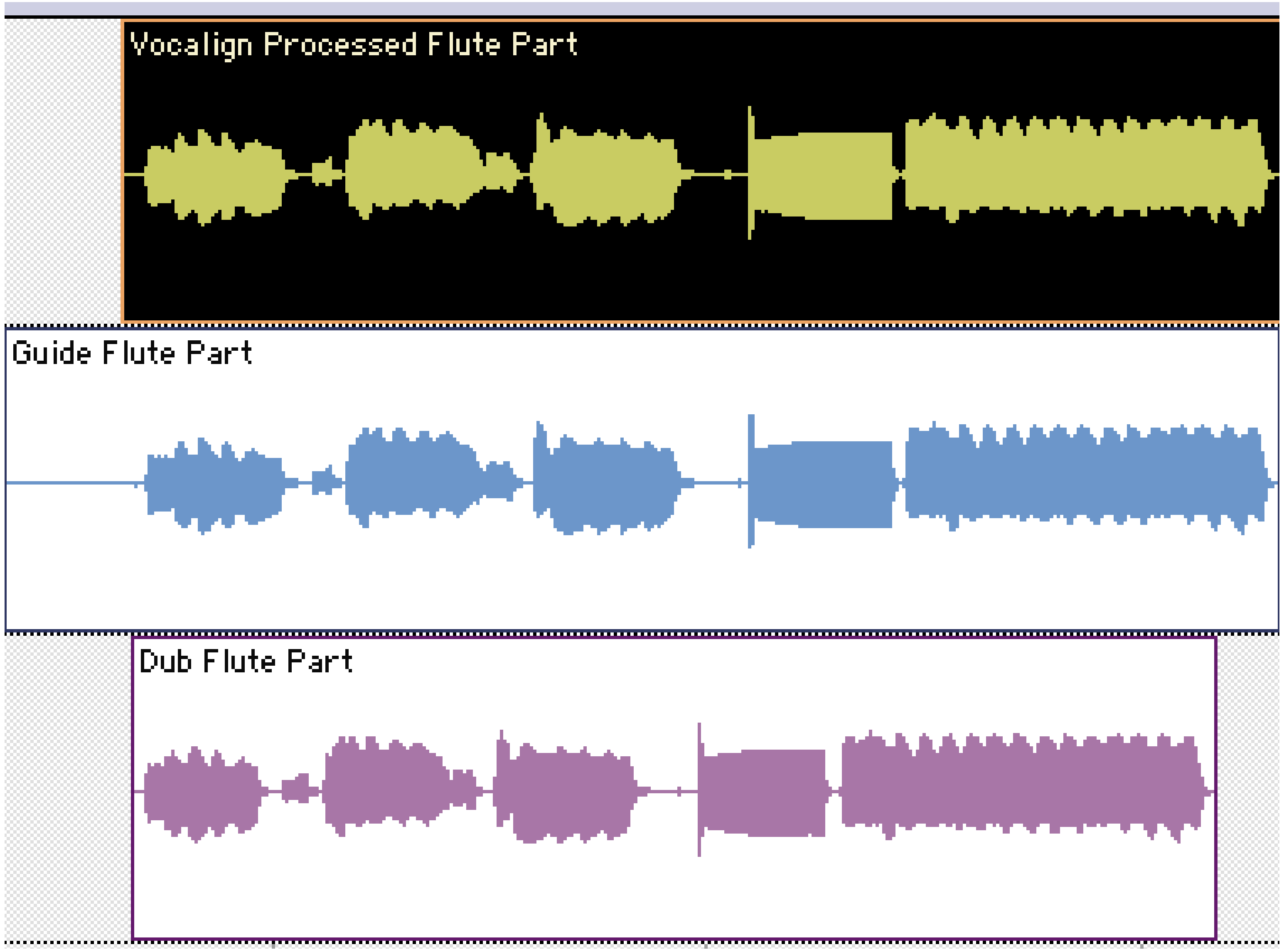
Setup: User selects segments of dub and guide audio tracks for alignment.

Result: Blue line shows envelope of aligned dub audio (user can also listen).



Fine-tuning: User can choose different algorithms to improve fit, then "print" best one.

Tuned for speech, but doesn't seem to use speech-exclusive features ...



"aligned"

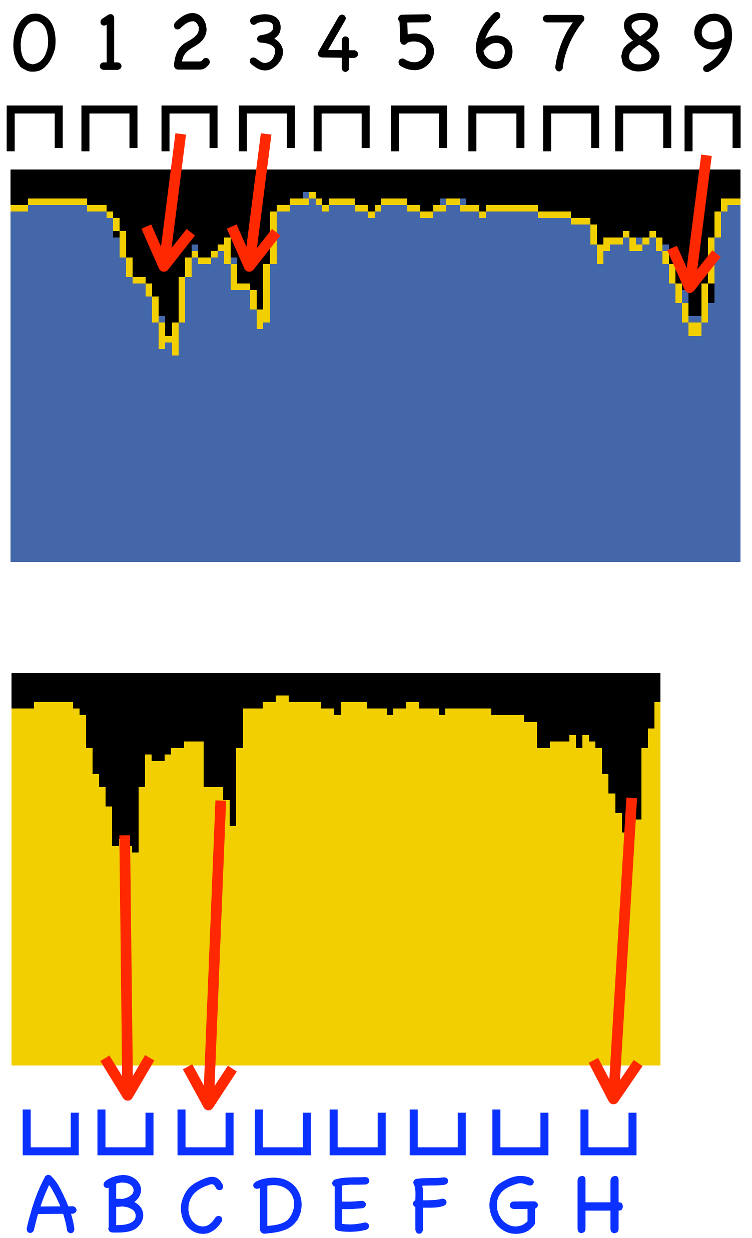
"guide"

"dub"

Double-track rap example: Before. After.

How could it work? Dynamic Time Warping

We are given this ...



"guide"
(longer)

"dub"
(shorter)

We compute this ...

0,1:A
2:B
3:C
4,5:D
6:E
7:F
8:G
9:H

Time stretch "dub" here to equalize lengths ...

Note: **If a good match required time shrinking parts of "dub", #:α,β items would appear in list.**

Local costs: How well does 2 match C ?

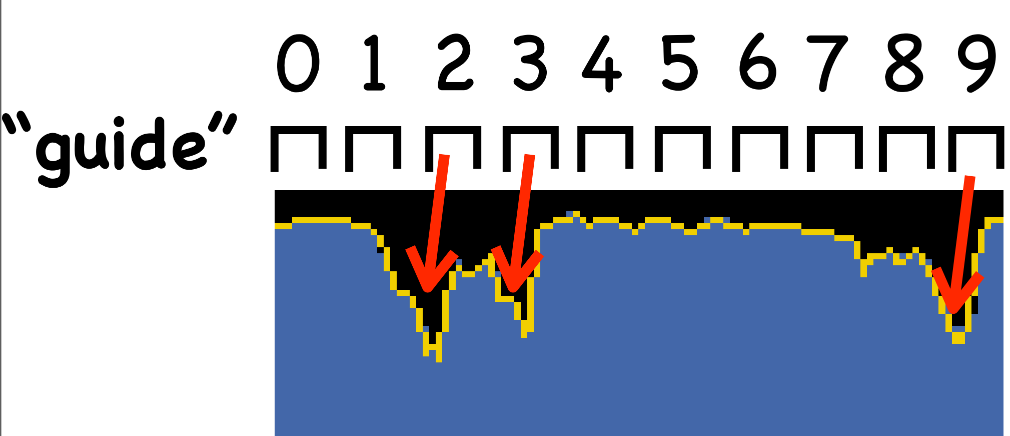
Euclidian distance of energy between "guide" and "dub"

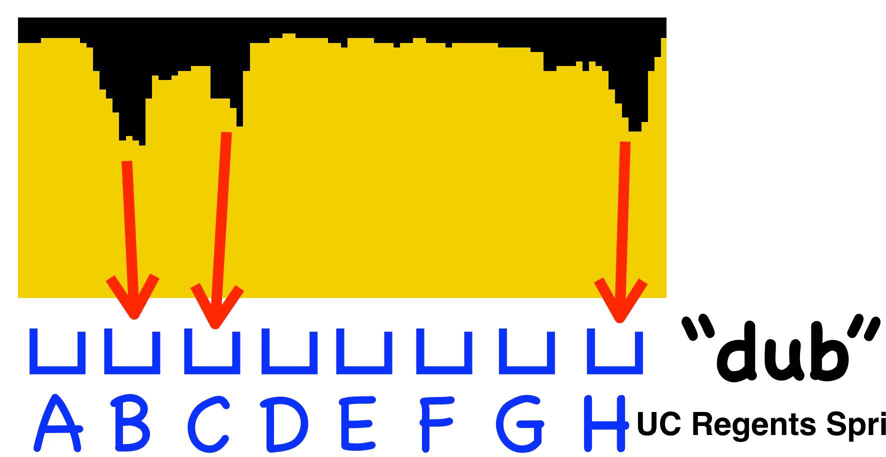
	0	1	2	3	4	5	6	7	8	9
A										
B										
C										
D										
E										
F										
G										
H										

Good Fit

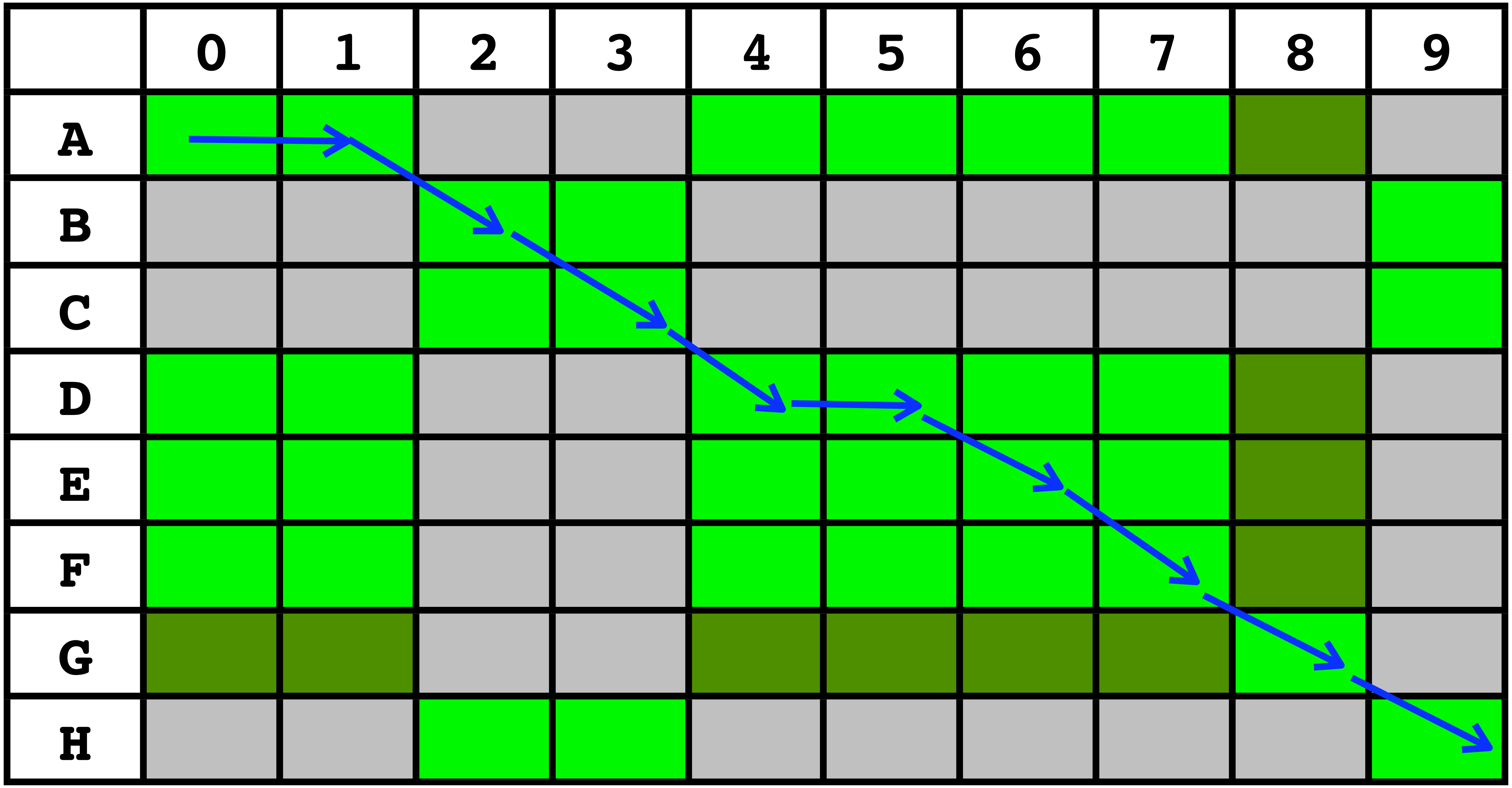
OK Fit

Bad Fit





Goal: Find "path" with highest "global fit"



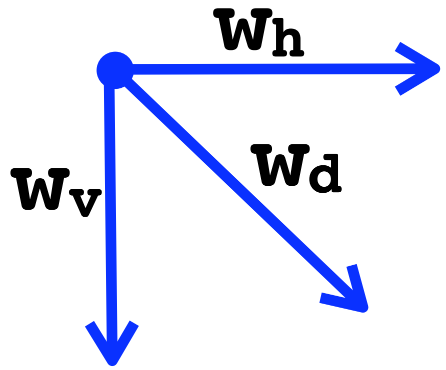
Recall:

0,1:A
2:B
3:C
4,5:D
6:E
7:F
8:G
9:H

We determine the global fit of a path by summing all of the local fits in the path.

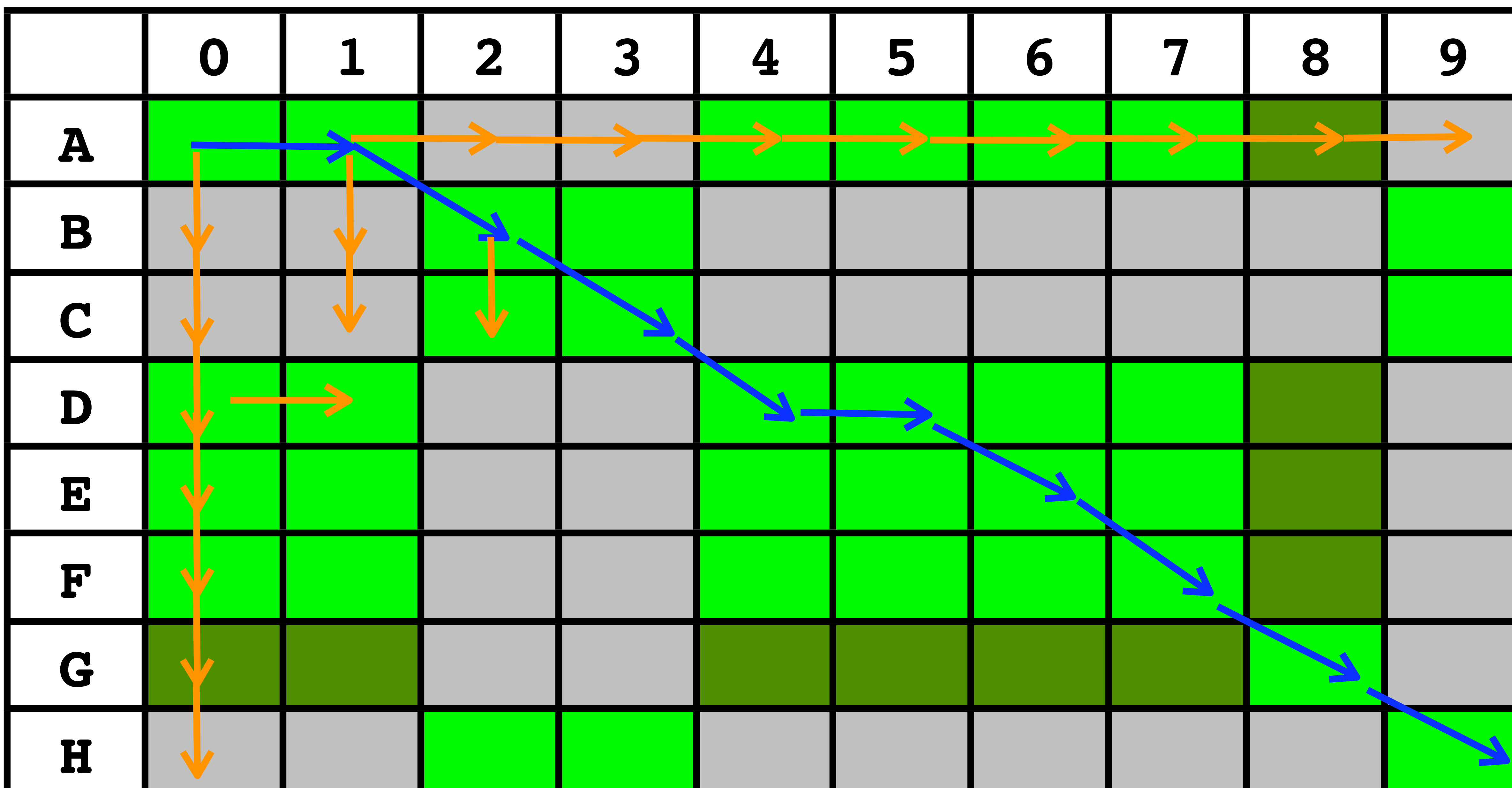
Algorithm: Trace all paths to (9, H) ...

For each square, pick **best fit** legal path into square.



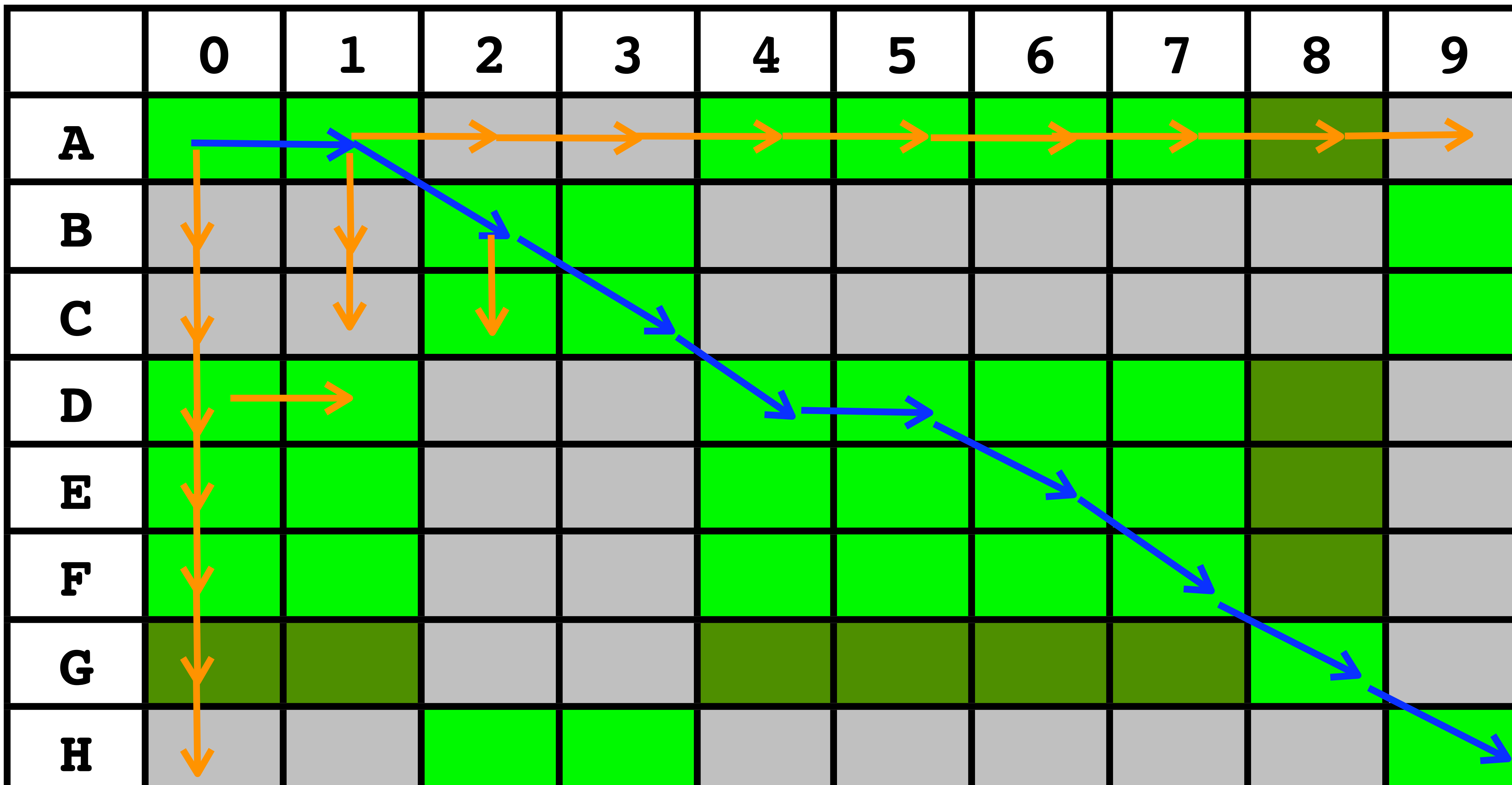
<<< Only legal transitions in this example.

Weights multiply local fitnesses in global path fitness calculation. Prevents **Manhattan** paths.



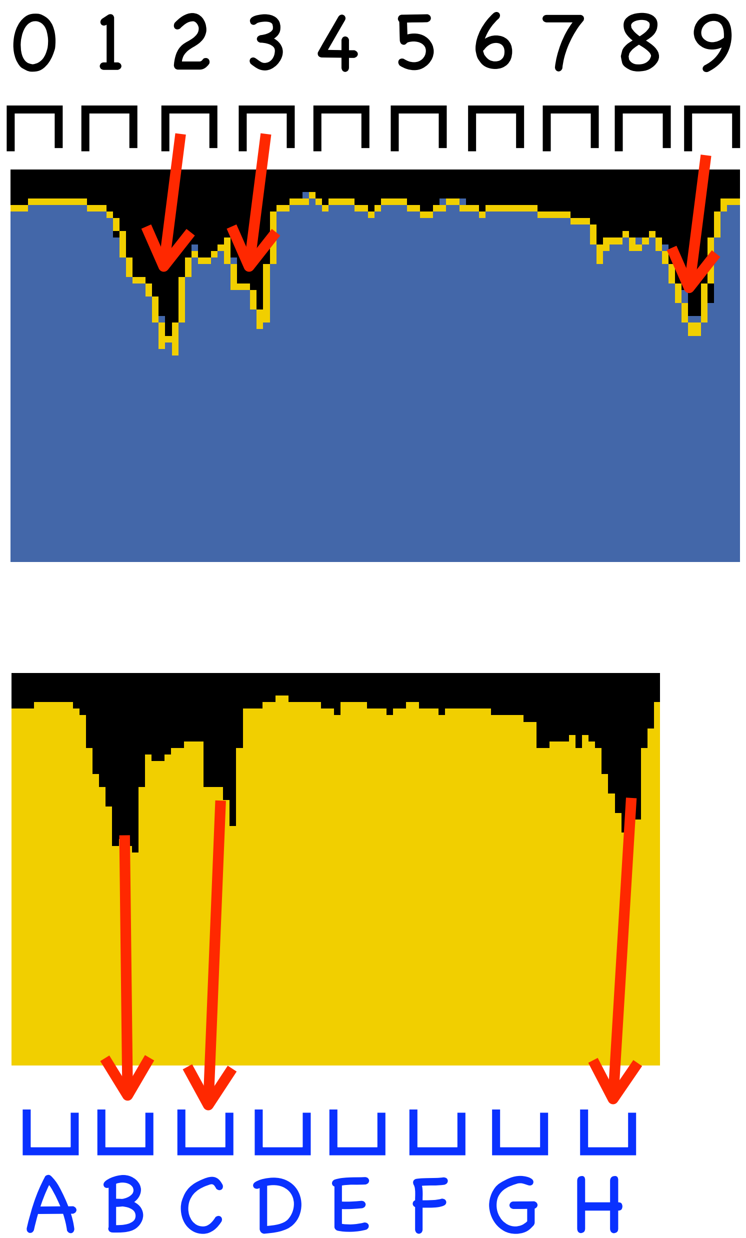
Isn't this an "N-squared" problem?

Common trick to reduce computation: **Path pruning.**
Stop tracing paths when fitness is "too low" ...



How could it work? Dynamic Time Warping

We are given this ...



"guide"
(longer)

"dub"
(shorter)

We compute this ...

0,1:A
2:B
3:C
4,5:D
6:E
7:F
8:G
9:H

Time stretch "dub" here to equalize lengths ...

Note: **If a good match required time shrinking parts of "dub", #:α,β items would appear in list.**

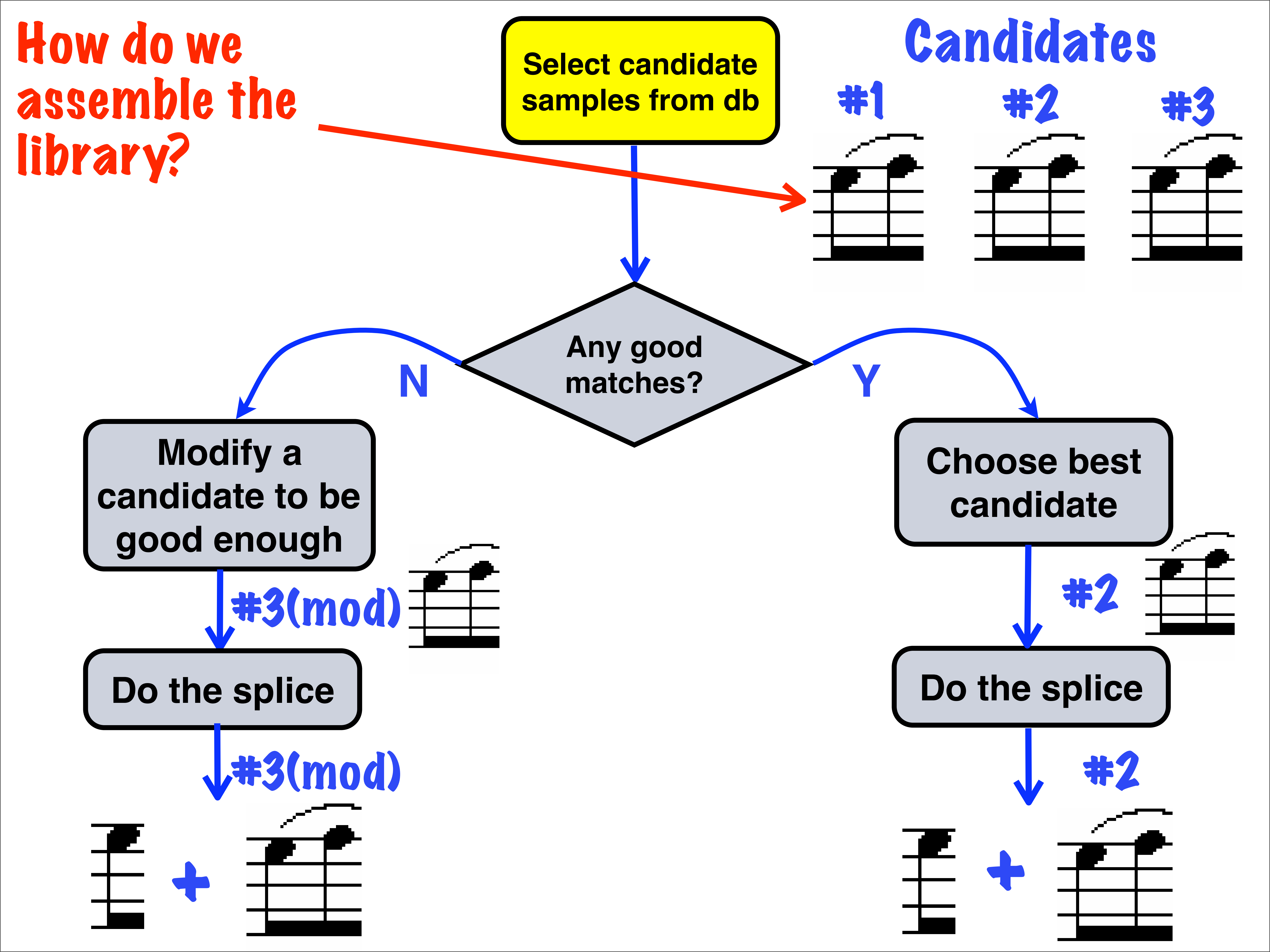
Admin: Progress Report Presentations

Progress Report Presentation	March 23 in class	A 10-15 minute presentation to the class, describing the current status of the project. Group projects should share presentation duties between all members. Audio demos of work in progress is encouraged. Primary purpose of presentation is to solicit feedback from the audience.	15 percent

Cynthia Psyche Jeremy

Bradley Carlos Eric

If you are enrolled (or are auditing and doing a project) and not on the list, let us know!


How do we assemble the library?
Select candidate samples from db
Candidates
#1
#2
#3
Any good matches?
N
Y
Modify a candidate to be good enough
#3(mod)
Do the splice
#3(mod)
Choose best candidate
#2
Do the splice
#2

Answer: "Guide" track is the score

In many ways, the architecture we saw for Vocalign directly maps to score alignment.

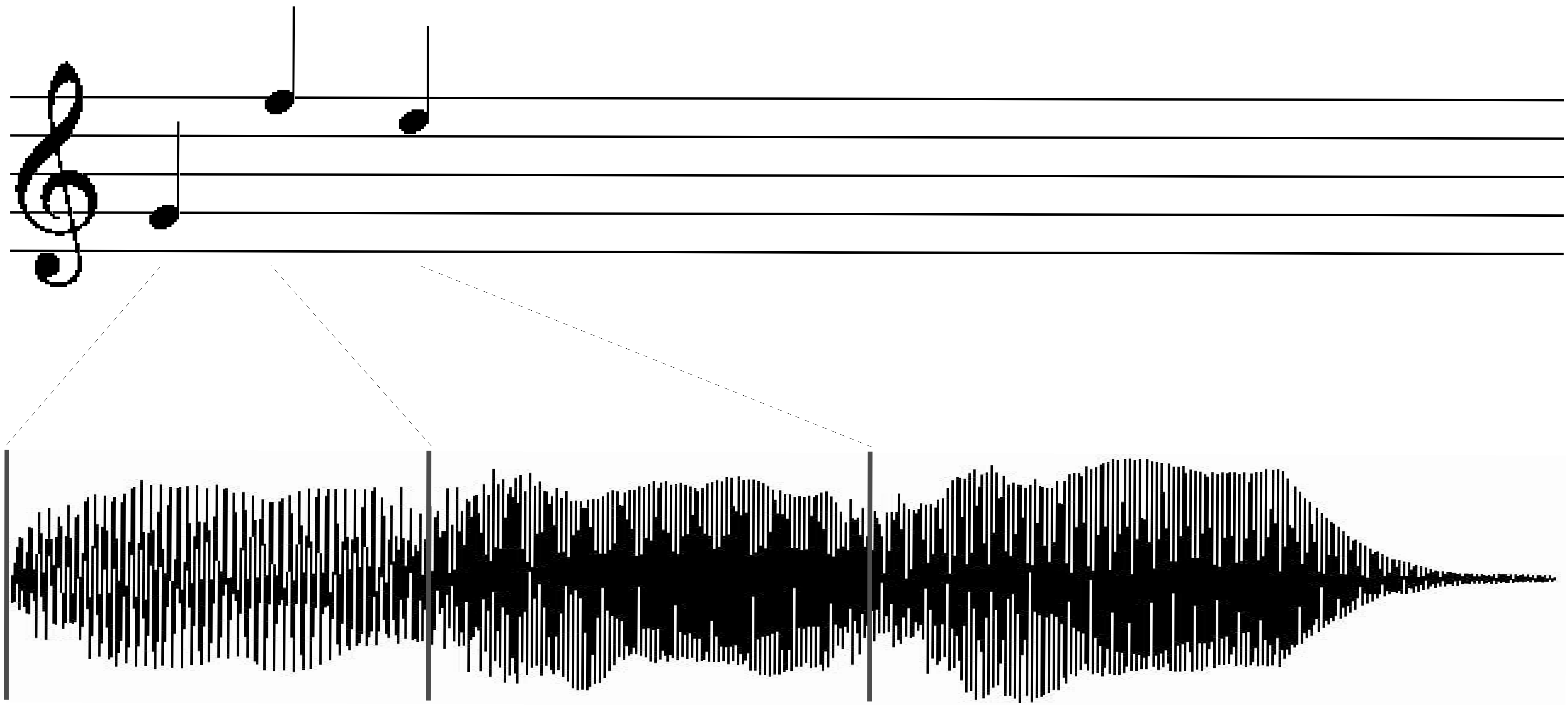


Figure 5.1: The principle of music alignment

If “score” is recorded MIDI, caveats ...

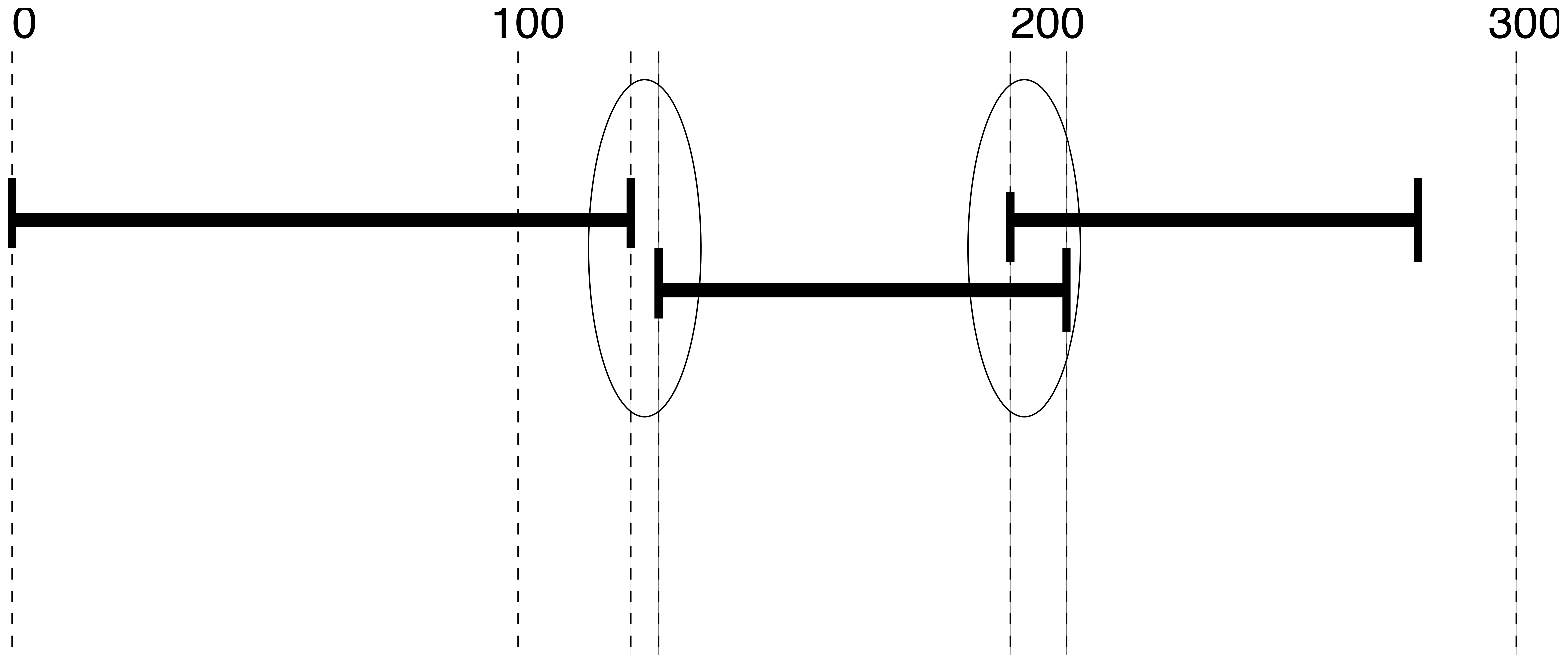


Figure 5.5: Desynchronised legato notes.

If "score" is recorded MIDI, caveats ...

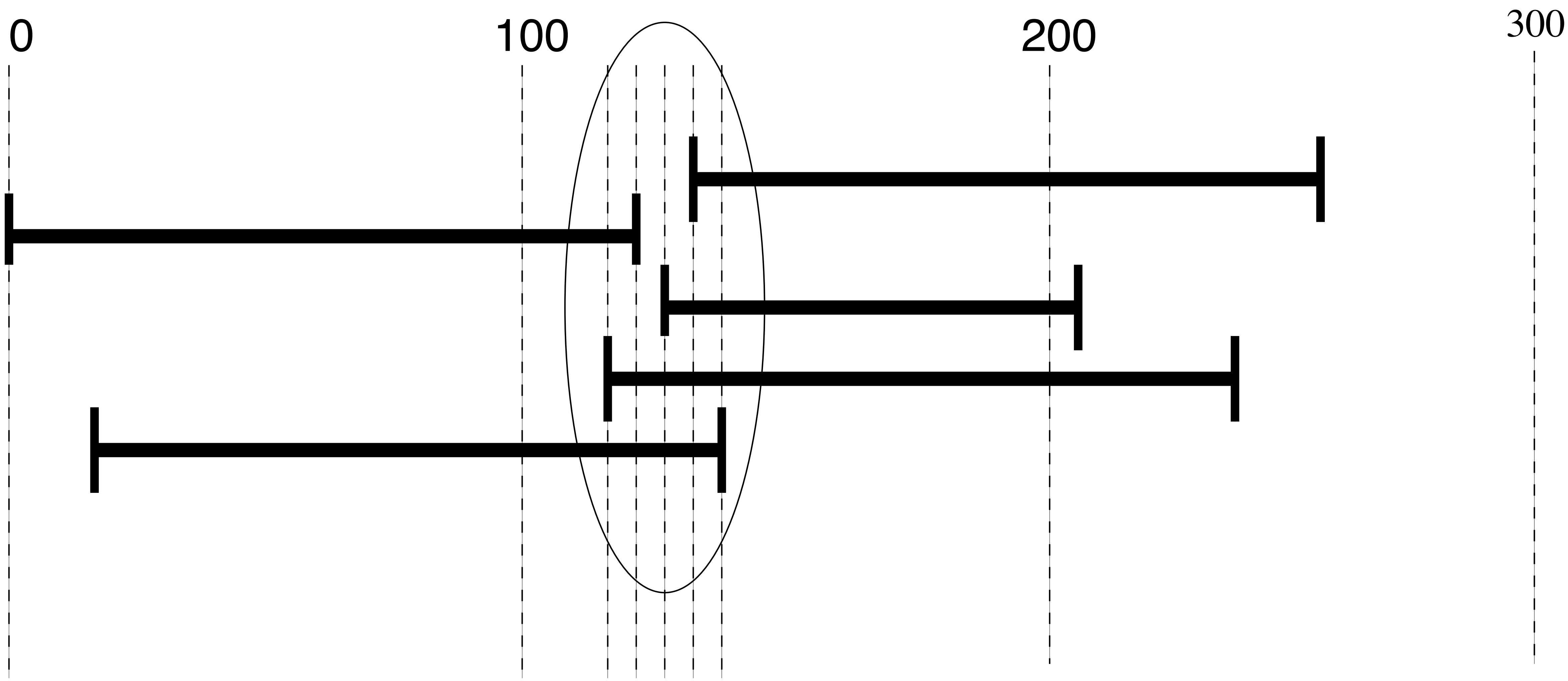
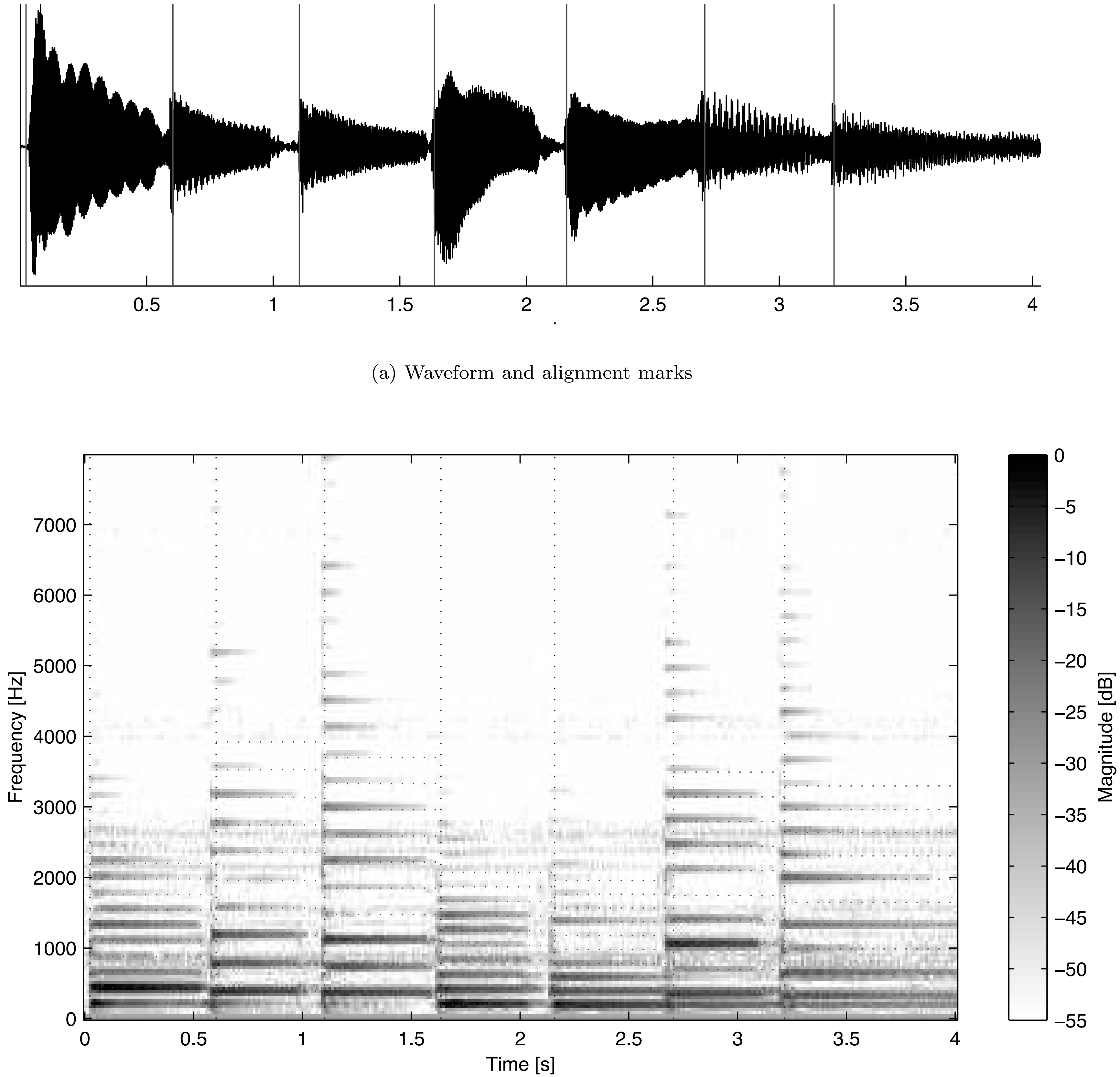


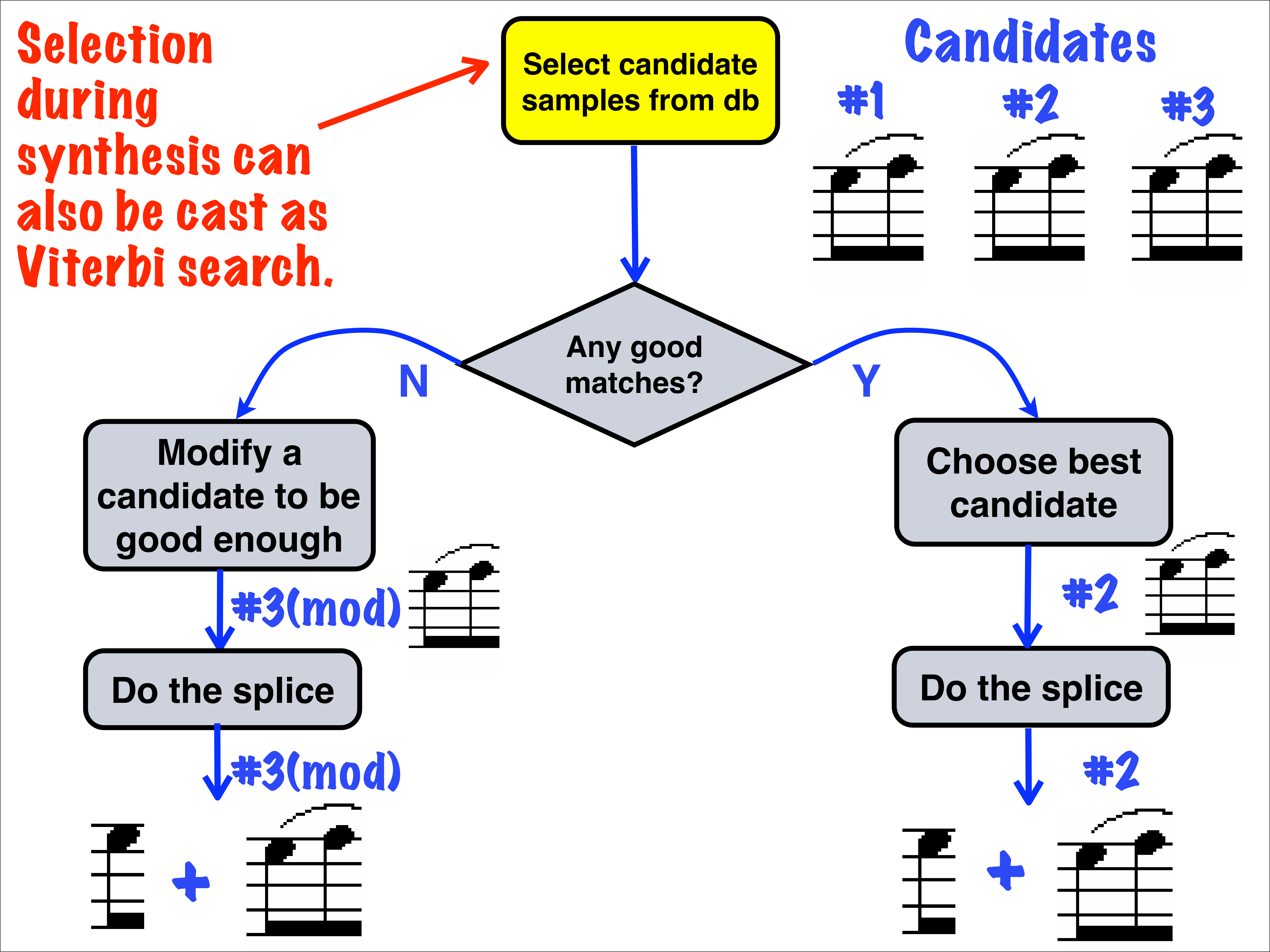
Figure 5.4: Desynchronised chord.



(a) Waveform and alignment marks

(b) Spectrogram and alignment marks

Figure 5.7: Alignment result example for an easy Guitar melody


Selection during synthesis can also be cast as Viterbi search.
Select candidate samples from db
Candidates
#1
#2
#3
Any good matches?
N
Y
Modify a candidate to be good enough
#3(mod)
Do the splice
#3(mod)
+
Choose best candidate
#2
Do the splice
#2
+

16.3.1 Target Cost

The *target cost* C^t corresponds to the perceptual similarity of the database unit u_i to the target unit t_τ. It is given as a sum of p weighted individual feature distance functions C^t_k as:

$$C^t(u_i, t_\tau) = \sum_{k=1}^{p} w^t_k \, C^t_k(u_i, t_\tau) \tag{16.3}$$

To favour the selection of units out of the same context in the database as in the target, the *context cost* C^x or *extended target cost*, for the sake of the mnemonic, considers a sliding context in a range of r units around the current unit with weights w_j decreasing with distance j.

$$C^x(u_i, t_\tau) = \sum_{j=-r}^{r} w^x_j \, C^t(u_{i+j}, t_{\tau+j}) \tag{16.4}$$

16.3.2 Concatenation Cost

The *concatenation cost* C^c expresses the discontinuity introduced by concatenating the units u_i and u_j from the database. It is given by a weighted sum of q feature concatenation cost functions C^c_k:

$$C^c(u_i, u_j) = \sum_{k=1}^{q} w^c_k \, C^c_k \, (u_i, u_j) \tag{16.5}$$

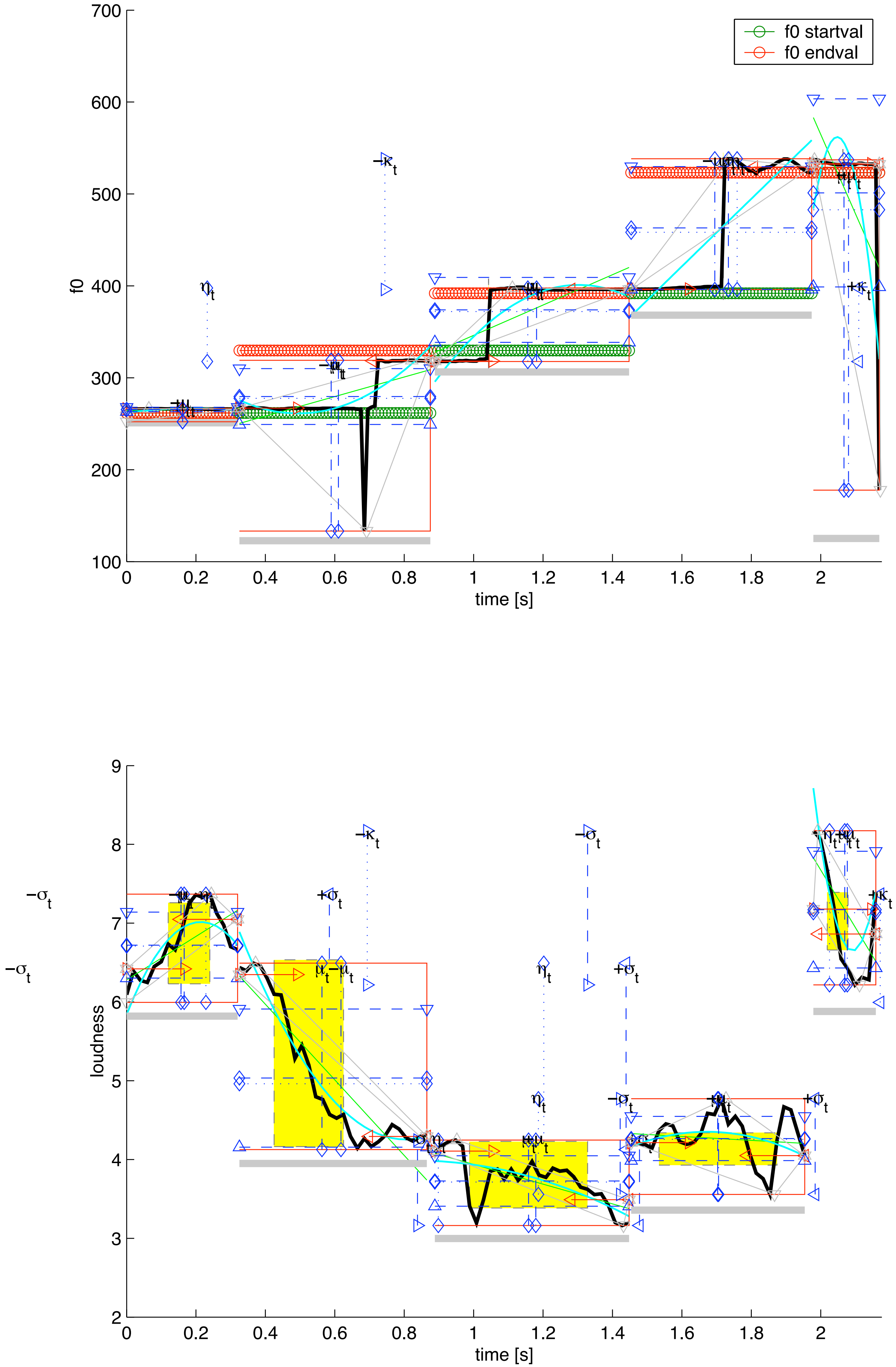


Figure 17.4: Selected units and characteristic values of pitch (f0) and loudness. The selection