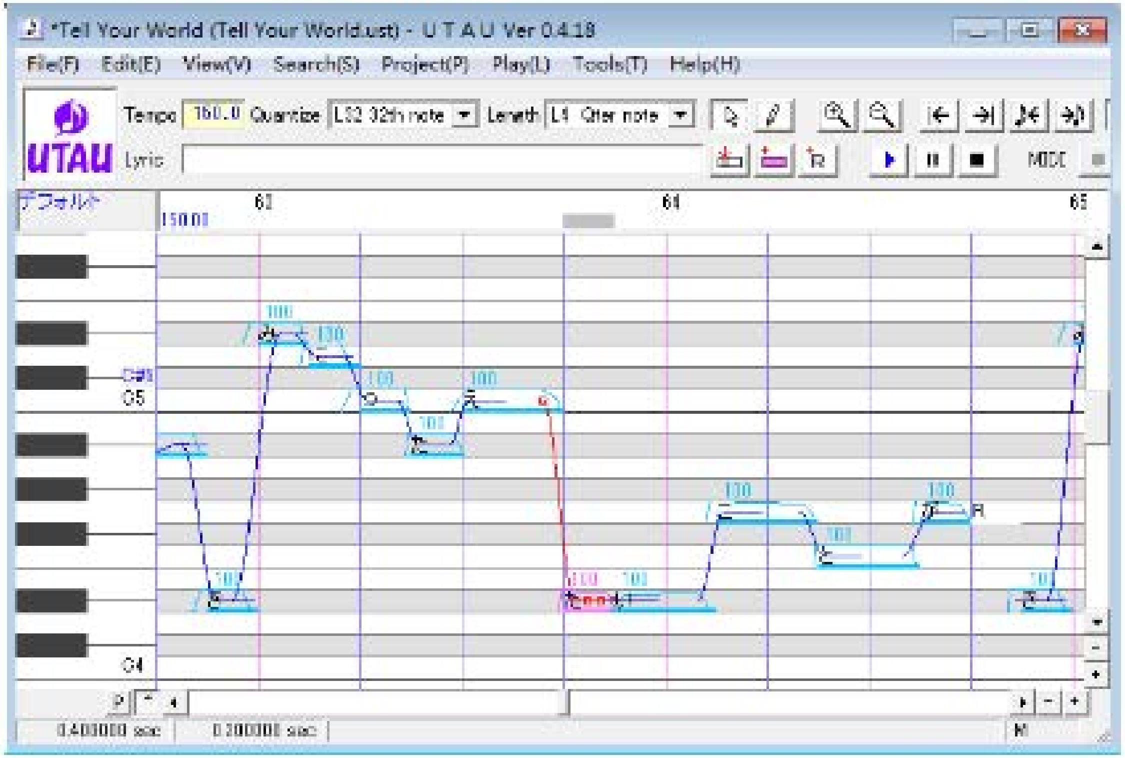
How does Moresampler work?

- That's a great question.
- Answering it in detail requires some tough Digital Signal Processing stuffs. We won't go that far, but these slides will provide you with a conceptual understanding of what's going on inside Moresampler, and how it is different from other resamplers.
- Knowing these will certainly help you make better use of Moresampler.

How does UTAU work?

- To see how Moresampler work, we need to view it in a broader context first.
- What's going on behind this GUI (Graphical User Interface)?



*Tell Your World (Tell Your World.ust) - U T A U Ver 0.4.18

Tempo 160.0

Quantize L32 32th note

Length L4 Qter note

How does UTAU work?

(1)User selects a few notes and presses “play”

(2)The GUI creates a temporary script and writes a bunch of codes into it.

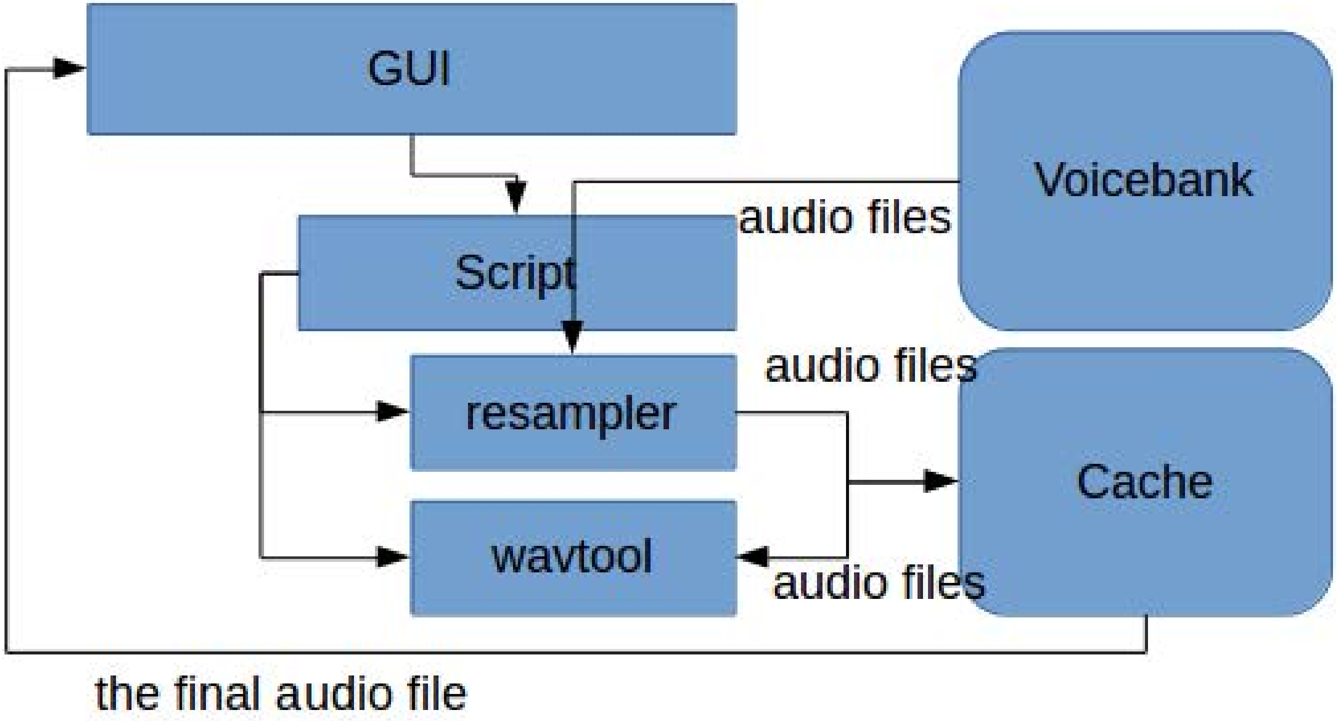
(3)This script calls resampler and wavtool to work on audio files that correspond to the selected nodes.

(4)Resampler/wavtool are called once or multiple times for each note. Each time a short audio segment is outputed to a cache directory.

How does UTAU work?

(5)Finally, a chain of wavtool call is performed to concatenate all cached segments together.

(6)Now go back to the GUI. The GUI loads the concatenated audio file and play it.



How does UTAU work?

- The role of resampler:

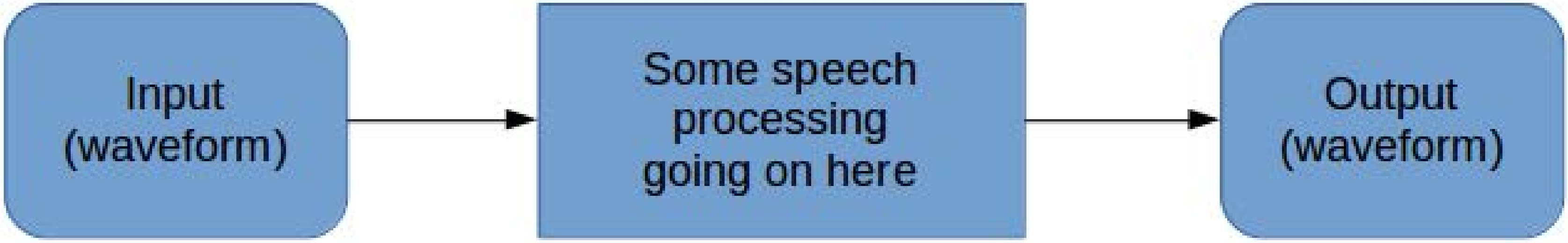
  Shifting the pitch, stretching/compressing the duration, applying flags (e.g. 'g' changes gender).

- The role of wavtool:

  Amplitude enveloping the audio segment, concatenating the segments with or without overlaping.

Parametric/Non-parametric Method

- We've seen the role of resampler. In a more general sense, it's all about speech processing.
- There are two broad types of speech processing methods:
- First, you feed some speech wave in; it goes through some processing, and modified speech wave comes out.

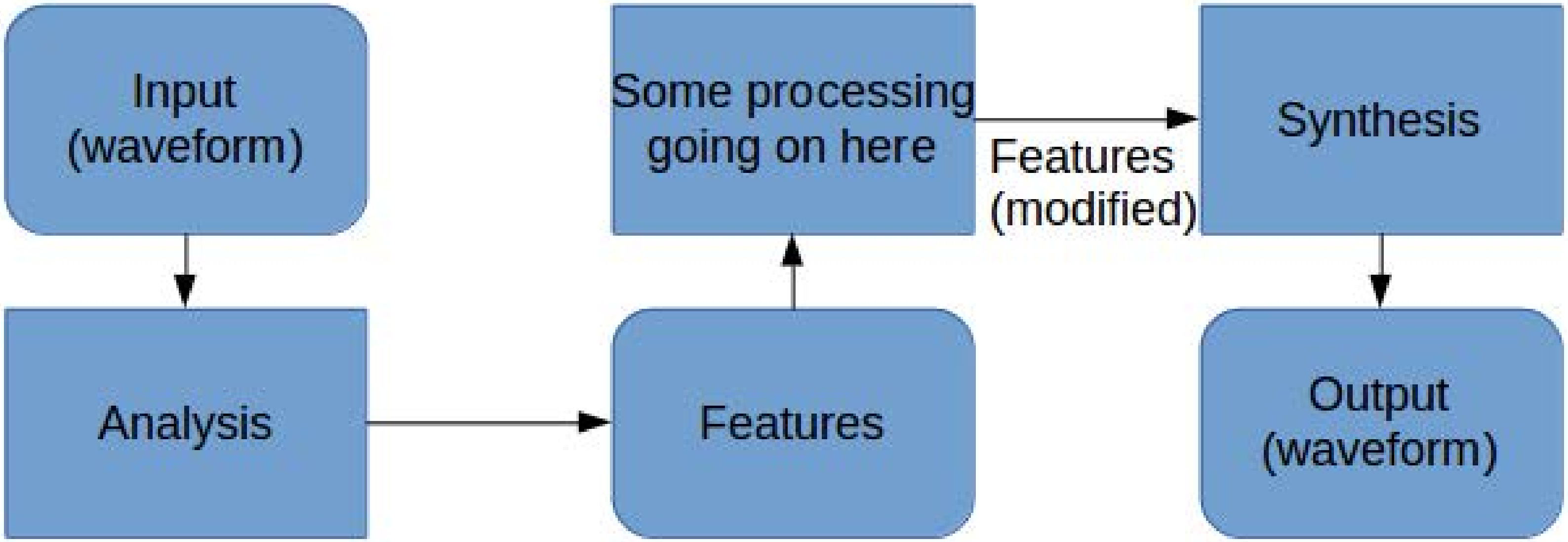

Input (waveform)

Some speech processing going on here

Output (waveform)

Parametric/Non-parametric Method

- Second, you feed some speech wave in; the speech wave is analyzed; features are extracted from the input; features go through some processing; output speech wave is synthesized (i.e. reconstructed) from the modified features.



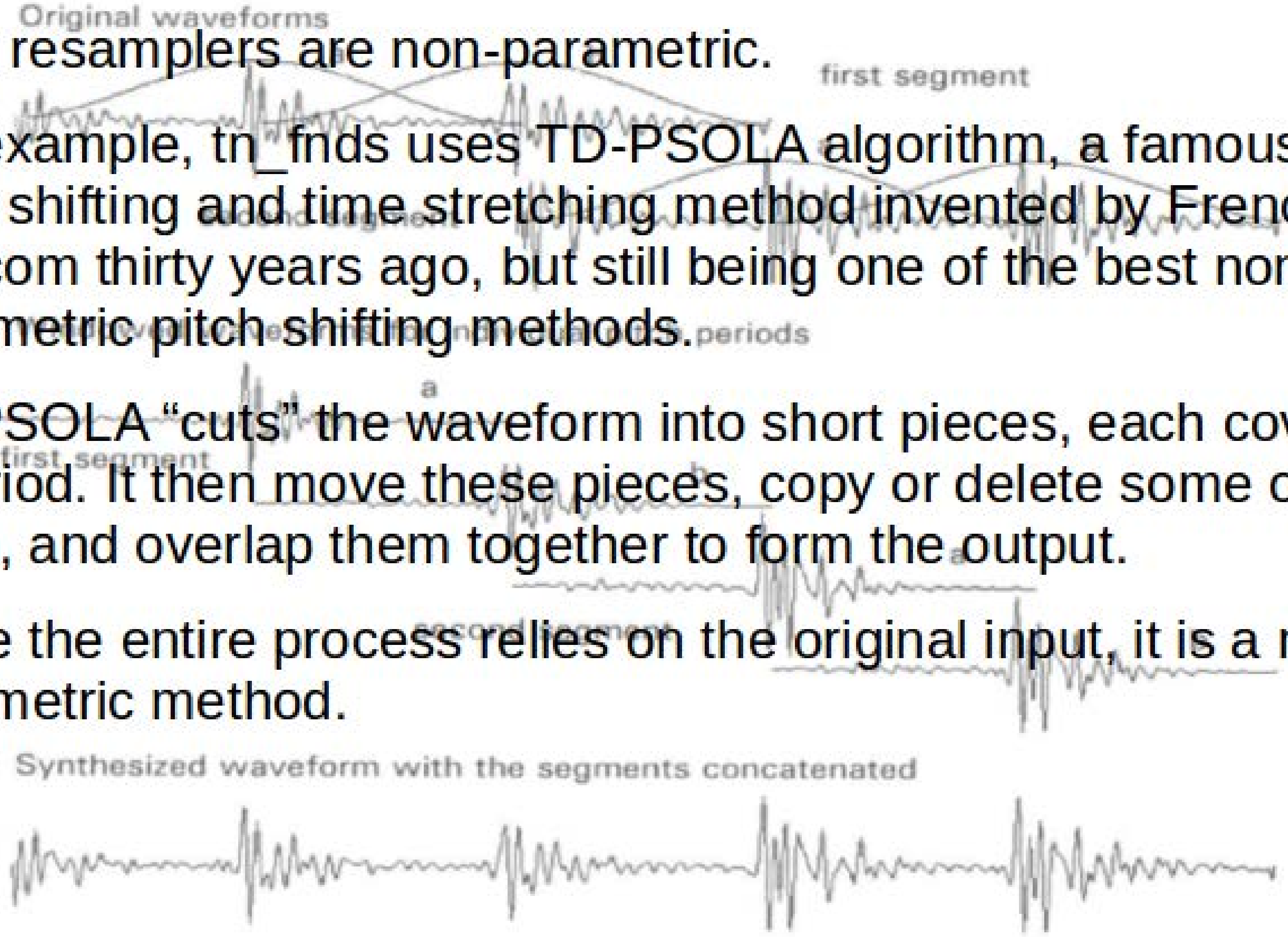
Parametric/Non-parametric Method

- The features, for example, can be the fundamental frequency (what we call “pitch”), the spectrogram, the energy, etc.
- They must form a complete description of speech that the synthesis no longer requires the original input waveform.
- Why on earth we need these seemingly redundant procedures (analysis and synthesis)? Why don't we directly modify it like what we did in the first method? Wouldn't that be simpler?

Parametric/Non-parametric Method

- Well, that's another great question which I will explain later. In general, the second method is more flexible because we can do almost anything we want to these features. On other hand, waveform is hard to deal with.
- We call the first class of methods “non-parametric”;
- The second class is called “parametric”.

Parametric/Non-parametric Resamplers



- Most resamplers are non-parametric.
- For example, tn_fnds uses TD-PSOLA algorithm, a famous pitch shifting and time stretching method invented by French Telecom thirty years ago, but still being one of the best non-parametric pitch shifting methods.

  TD-PSOLA “cuts” the waveform into short pieces, each covering a period. It then move these pieces, copy or delete some of them, and overlap them together to form the output.

  Since the entire process relies on the original input, it is a non-parametric method.

[Holmes, 2001]

Parametric/Non-parametric Resamplers

- Non-parametric does not mean it has to be completely in time domain, i.e., dealing with waveform.
- Another example is phavoco, abbreviated from Phase Vocoder. It uses a frequency-domain non-parametric method that basically modifies the spectrum frame-by-frame, from input to output.

  Again, the output has direct dependency on the input, so it's non-parametric.

Parametric/Non-parametric Resamplers

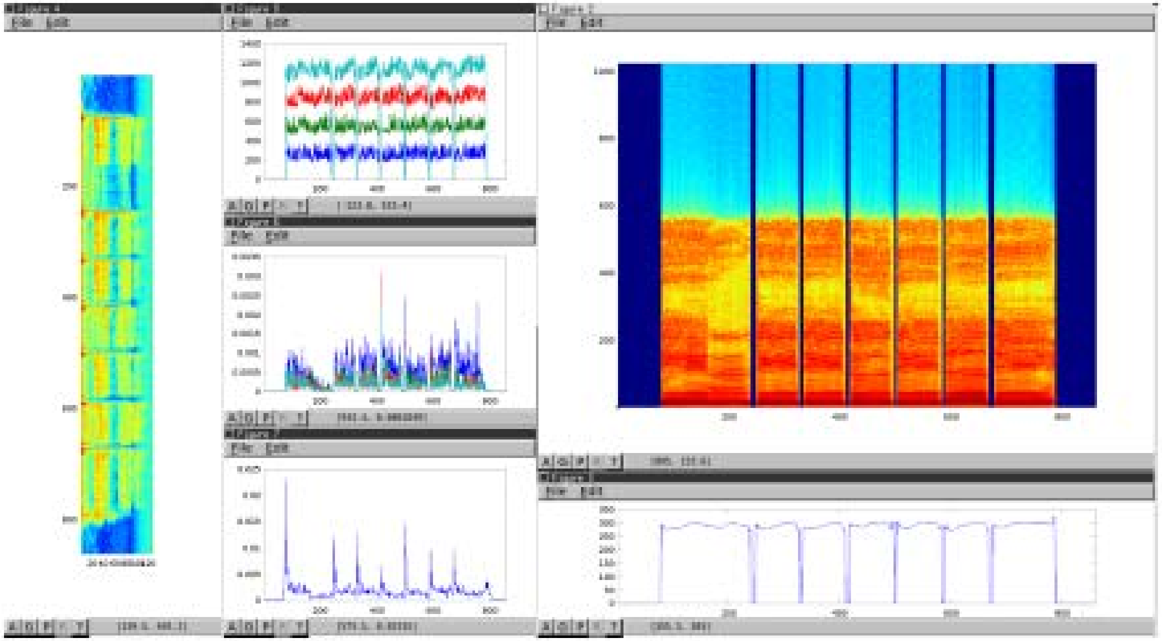
- Resamplers based on Dr. Morise's World speech synthesis system are mixed; the older version of World is non-parametric (which operates in a way similar to FD-PSOLA, a frequency domain variant of TD-PSOLA) while the newer one is parametric.
- The features used by the newer World are,
  - fundamental frequency
  - spectral evenlope: the shape, or contour of spectrum
  - aperiodicity: the amount of noisy sound the speech contains
- All of these features are time-varying. They are stored in hundreds or even thousands of very short frames.

Parametric/Non-parametric Resamplers

- Given these features, World synthesizer is able to reconstruct speech, without refering to the original recording. Hence it is parametric.
- In most cases the feature still come from the (analysis results of) original recording, even though it is parametric.
- But it's possible to generate these features “from nothing” - build a HMM-based speech synthesizer, feed the feature output from HMM to World, like what CeVIO or Sinsy did. But using HMM to drive PSOLA is not so possible.
- This example shows how flexible a parametric method can be. We're a bit off-topic now. Let's go back to Moresampler!

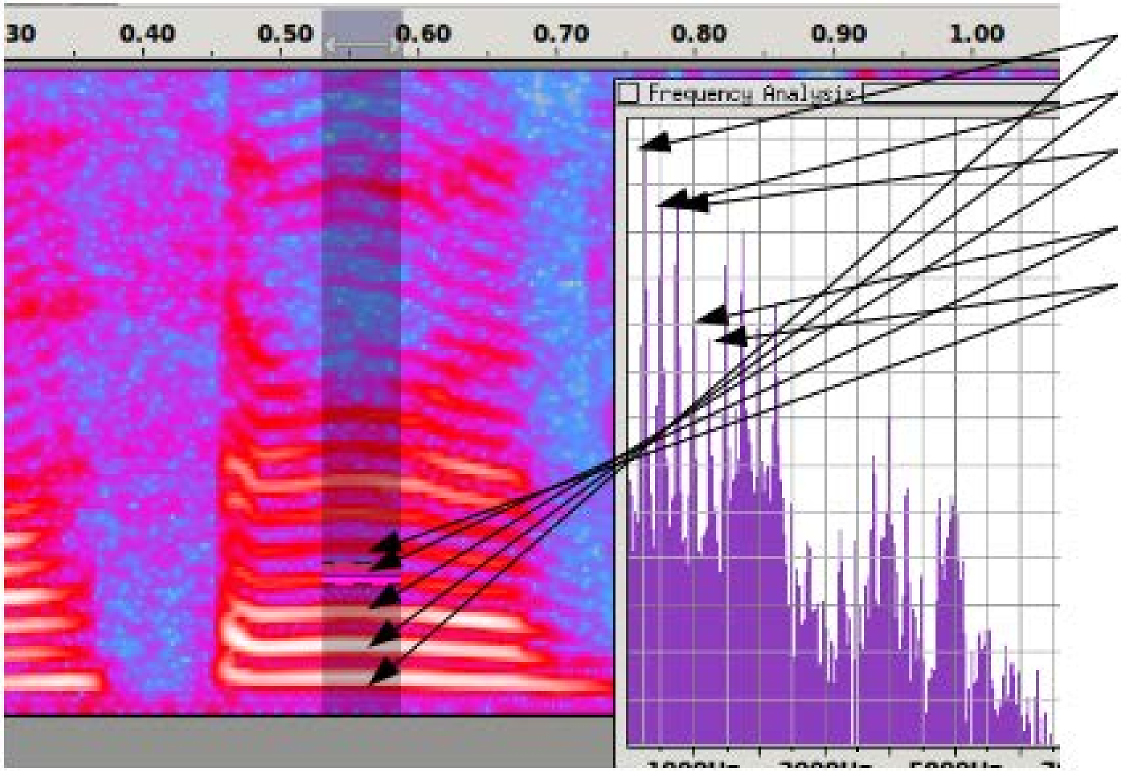
LLSM – foundation of Moresampler

- The parametric speech model used by Moresampler is LLSM (Low Level Speech Model)
- LLSM uses a variety of features to represent speech so the quality loss in analysis/synthesis stage is hardly perceptible.



LLSM – foundation of Moresampler

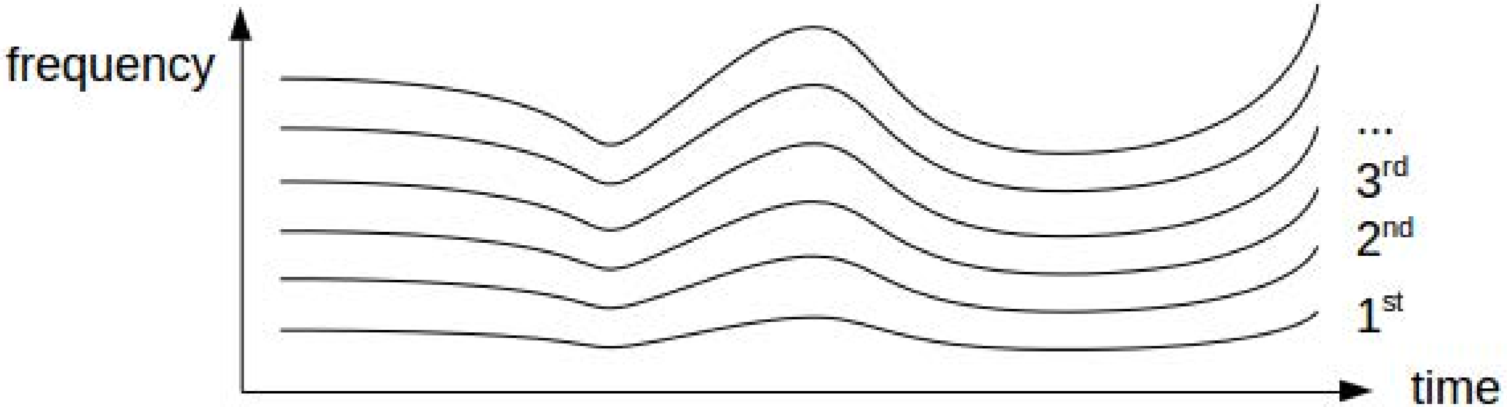
- The most important feature in LLSM is the harmonics.
- A harmonic of a wave is a component frequency of the signal that is an integer multiple of the fundamental frequency. [Wikipedia]



1st Harmonic (fundamental)
2nd Harmonic
3rd Harmonic
4th Harmonic
5th Harmonic
...

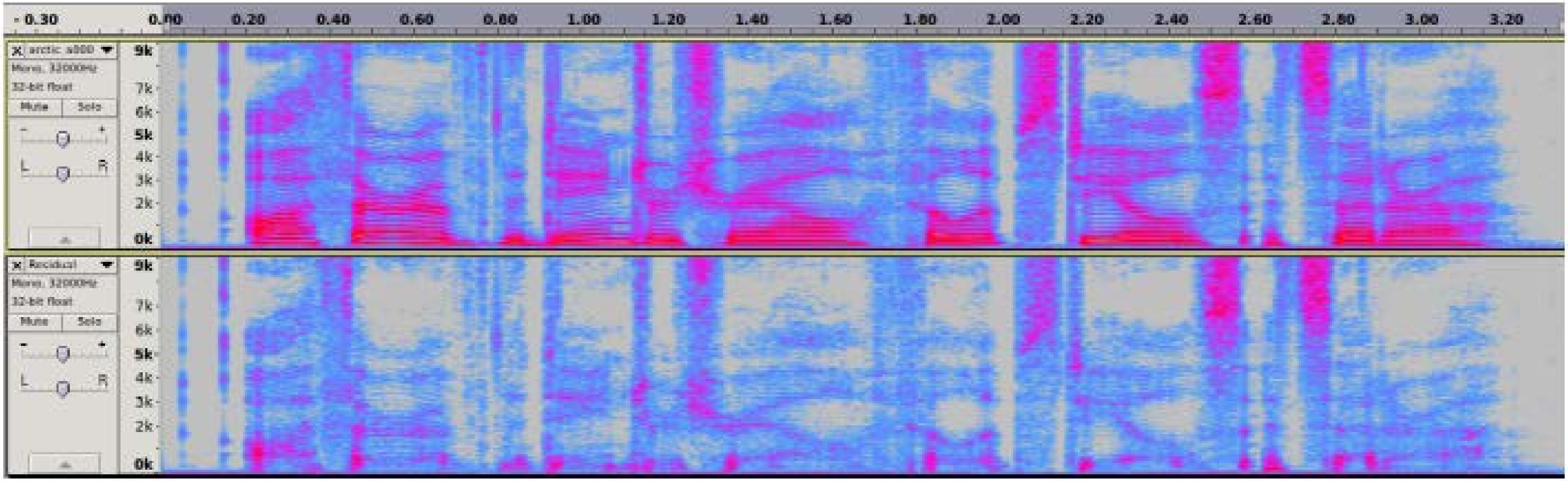
LLSM – foundation of Moresampler

- LLSM keeps track of the frequency, amplitude and phase variation of each harmonic.



- Harmonics are not enough to represent speech, which also contains non-harmonic components, for example, unvoiced consonants, aspiration noise...
- We can subtract the harmonics from the input to obtain these non-harmonic components. We call them residual.

Residual Modelling



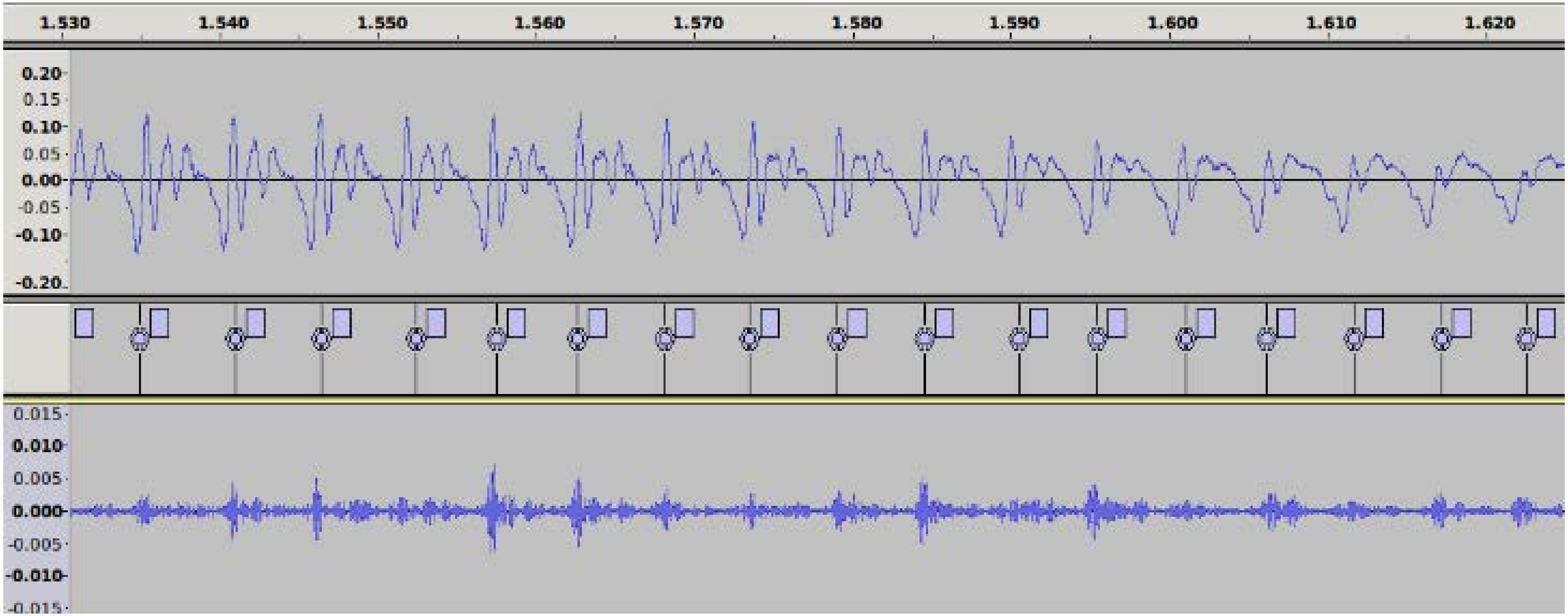
- Residuals are generally noise-like signals. They sound like whisper.
- We can generate a white noise, and then filter it to give it the same spectrogram as the residual.
- The filter parameters are the main features for noise.

Residual Modelling

- However by doing so, we lose many temporal details of the original residual.
- During phonation, your vocal folds rapidly vibrate and modulate the air flow passing through the vocal tract.
- Though the residual is not periodic, its energy is somewhat periodic. That means we can see the “volume” of residual oscillating!

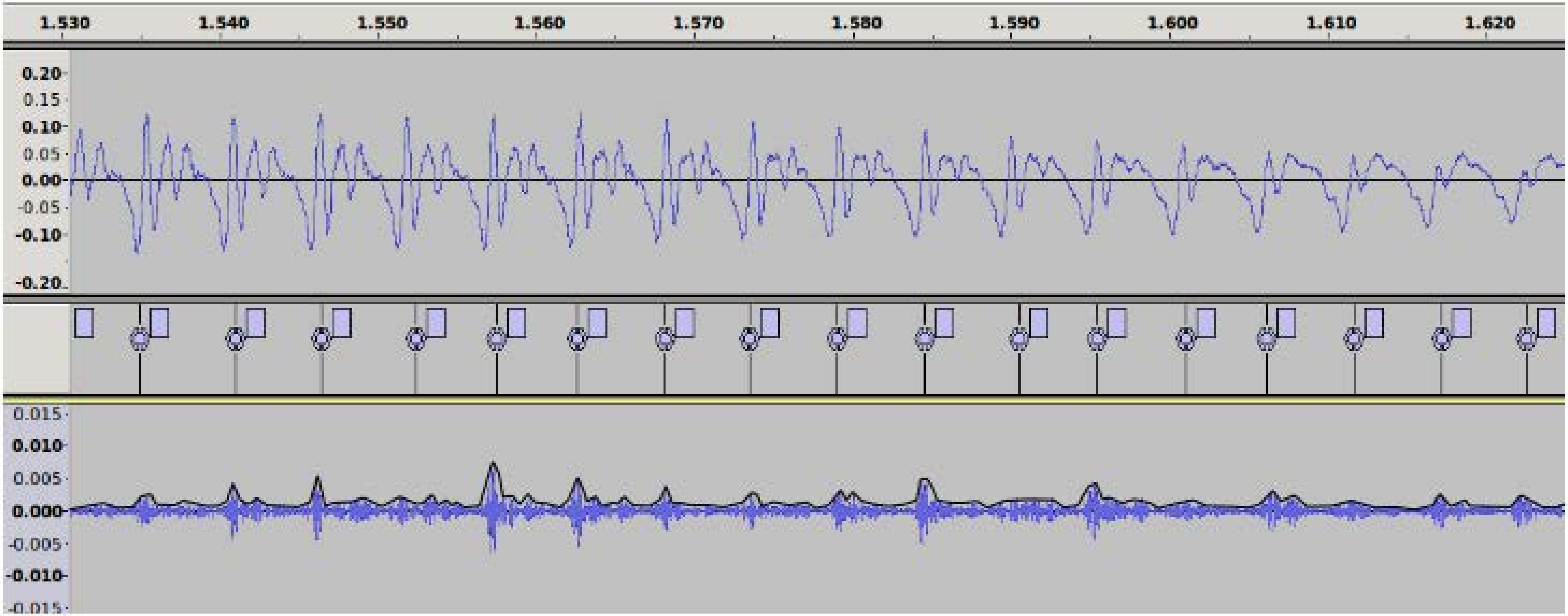
Residual Modelling

- We can see the weakly periodic residual energy:



Residual Modelling

- To model this, we put an envelope on it,



- We treat this envelope as a “speech wave”, and use another harmonic model to describe its amplitude variation!
- Very few harmonics (3 to 4) are enough because energy is concentrated in low frequency.

Residual Modelling

- We can even go a step further – to do this residual energy modelling for multiple frequency bands, for example:
  - 0 – 3KHz
  - 3 – 6KHz
  - 6 – 12KHz
  - 12KHz – sampling rate / 2
- Then we have 1 main harmonic model + 4 smaller harmonic models + 1 noise spectorgram which are basically what constitutes LLSM.

Works Cited

- LLSM is a blend of several sinusoidal model-related techniques. I must give credit to their authors.

Pantazis, Yannis, and Yannis Stylianou. "Improving the modeling of the noise part in the harmonic plus noise model of speech." Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on. IEEE, 2008.

Quatieri, Thomas F., and Robert J. McAulay. "Shape invariant time-scale and pitch modification of speech." Signal Processing, IEEE Transactions on 40.3 (1992): 497-510. 1992.

Serra, Xavier. "A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition." Diss. Stanford University. 1989.

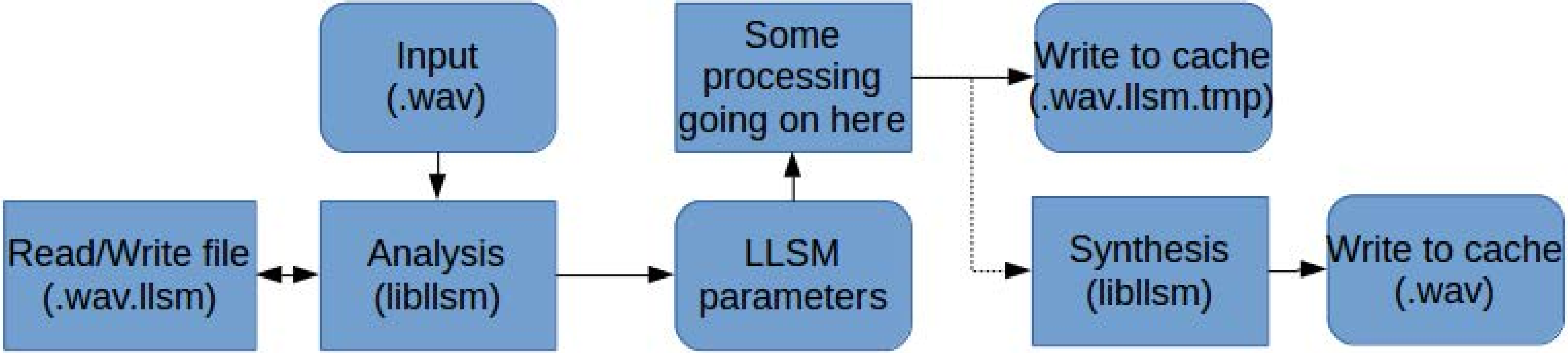
Stylianou, Yannis. "Harmonic plus noise models for speech, combined with statistical methods, for speech and speaker modification." Diss. Ecole Nationale Supérieure des Télécommunications. 1996.

Moresampler (resampler mode)

- When Moresampler is called by the script, it first figures out whether it is called as a resampler or as a wavtool.
- We now talk about the resampler mode.
- For the first time Moresampler, as a resampler, runs on a .wav file, it calls *libllsm* (an implementation of LLSM) to analyze the .wav.
- The analysis result is written to a .wav.llsm file. So next time Moresampler will skip the .wav and directly loads .wav.llsm. This even works if the .wav is deleted (though it's definitely not suggested to do so).
- Then it modifies the LLSM parameters (features), faithfully following the instructions given by the script (which tells it the pitch, duration, starting/ending time, flags, pitchbend, etc.).

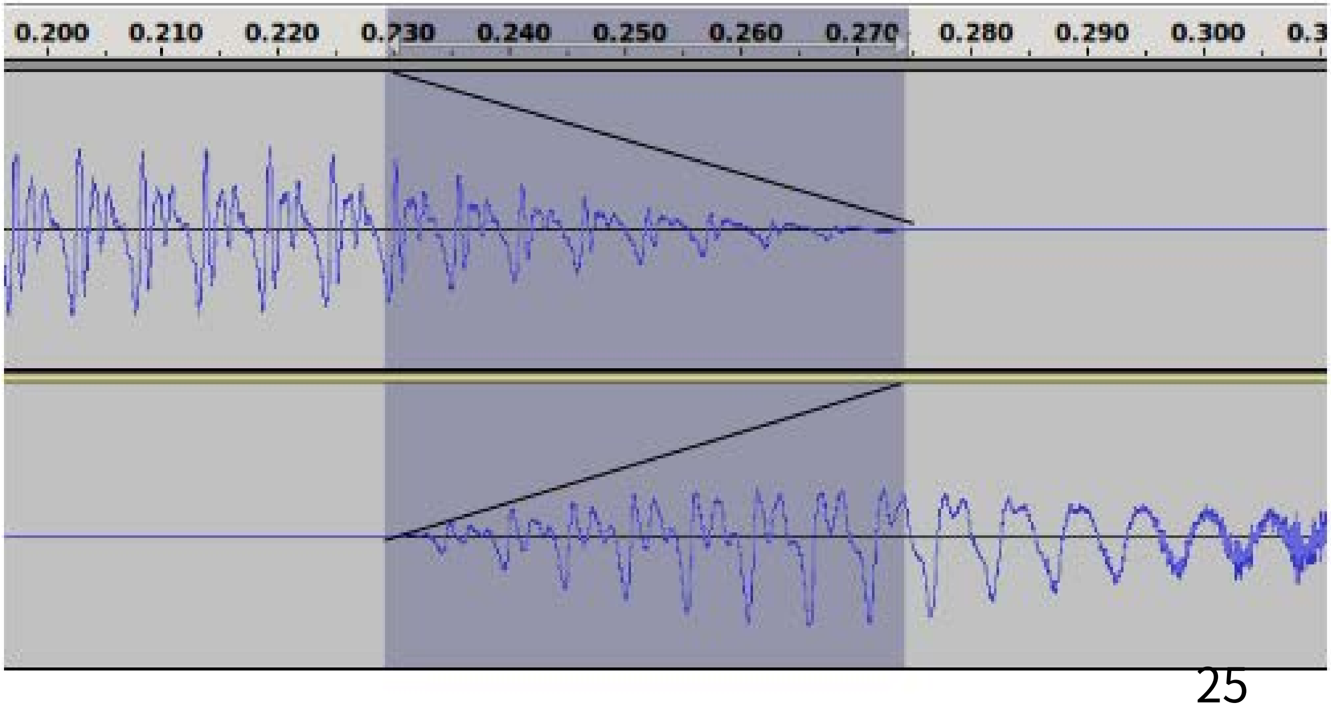
Moresampler (resampler mode)

- What shall it do next step?
- The script always gives Moresampler a output file path. Say it's C:\...\123_abc.wav
- Moresampler then outputs the modified LLSM parameters to C:\...\123_abc.wav.llsm.tmp
- It actually won't synthesize and give the C:\...\123_abc.wav in resampler mode, unless you turn on `resampler-compatiblity`.



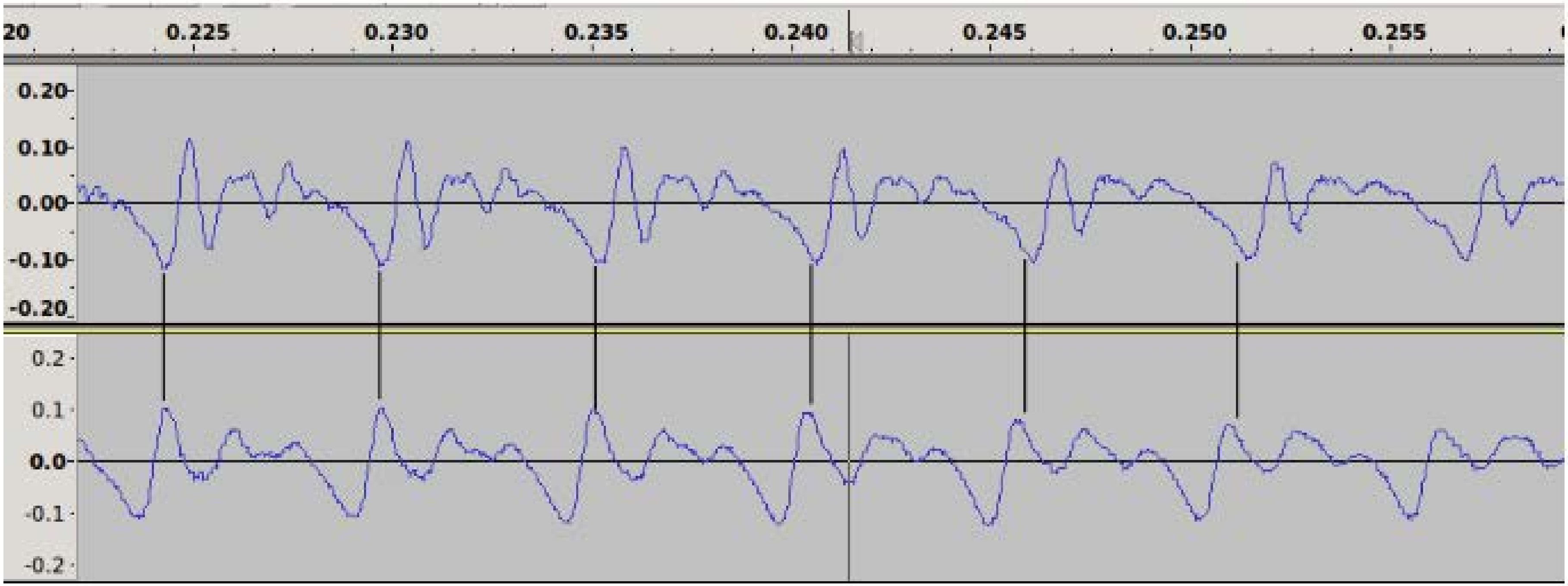
Concatenation of Segments

- Why not carrying out synthesis in resampler mode?
- We first look at how to concatenate waveforms in wavtool mode.
- To avoid glitches at boundaries, we need cross-fading.



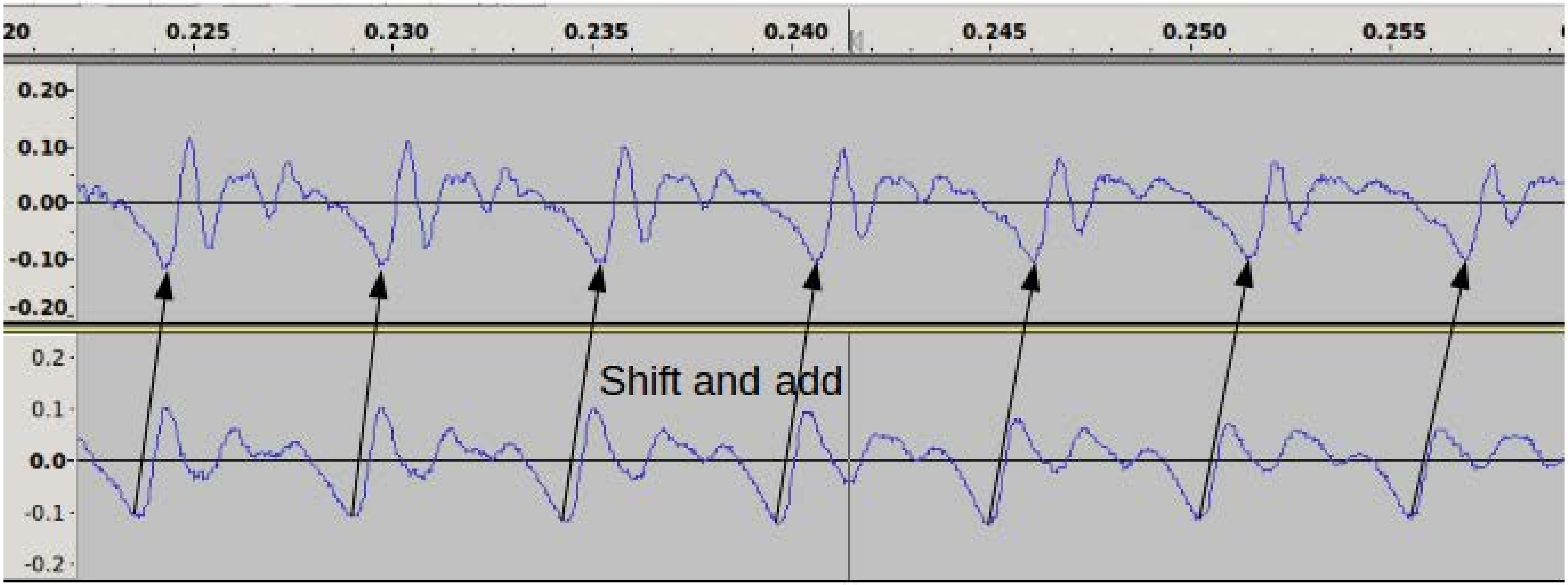
Concatenation of Segments

- Problem: peaks and valleys cancell out.
- We get interference!



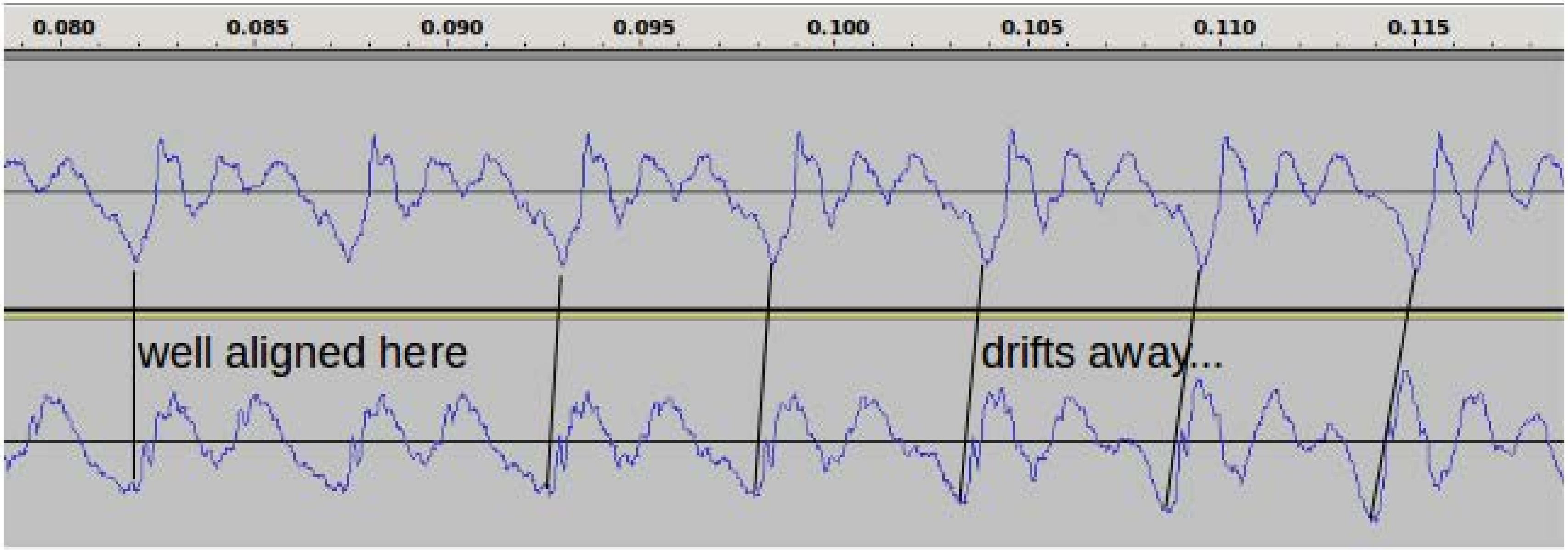
Concatenation of Segments

- One way to reduce interference is called maximum cross-correlation, which aligns/synchronizes the two pieces so that peaks match with peaks; valleys match with valleys.



Concatenation of Segments

- However this doesn't always work. When frequency slightly changes or phase slightly shifts, the periods lose alignment again.



Concatenation of Segments

- What's worse – if you shift the wave by t second, the n-th harmonic will have its phase shifted by $2\pi nft$ radians, where f is the fundamental frequency. (recall a sine wave at fHz is $y(t) = \sin(2\pi ft)$)
- This means if the alignment has some error, then the error will get larger as frequency goes up, which implies the higher harmonics may not be aligned even if the first few harmonics are well aligned.

A Parametric Solution to Cross-fading

- A better way is to cross-fade the LLSM parameters, instead of cross-fading the waveform.
- That is to say, cross-fade the harmonics, noise spectrums and fundamental frequency. Since these features are already aligned to frames*, we don't need to worry about interference.
- We call this *interpolation*.

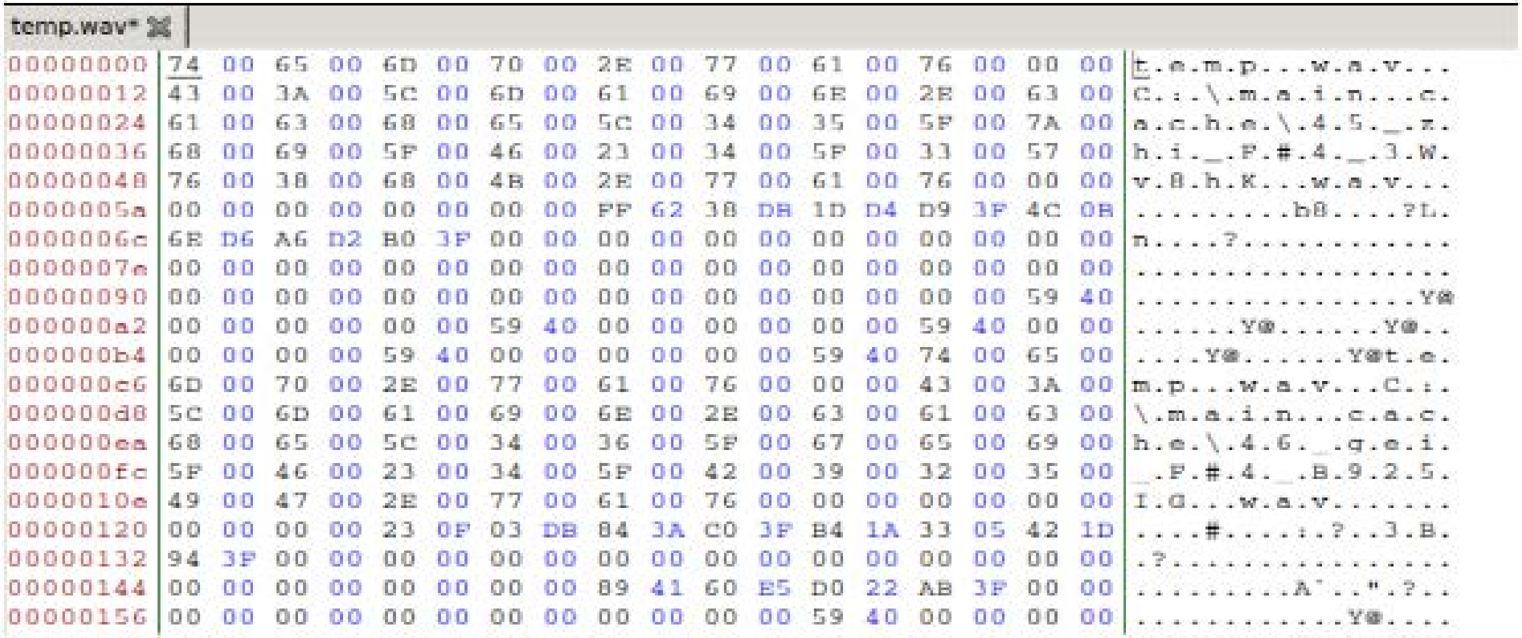
* This is an oversimple explanation. Being parametric doesn't guarantee interpolatability. Interpolation of harmonic phases has a great deal of technical stuffs to work on. Fortunately, Dr. Quatieri and Dr. McAulay had worked this out in 1992. LLSM did some further improvements on that.

Moresampler (wavtool mode)

- To implement this LLSM parameter-level cross-fading, we have to write a wavtool that accepts LLSM model file.
- This is why I made Moresampler both a resampler and a wavtool, so I don't need to maintain two projects.
- Taking account of the bulky size of LLSM files, it's better to modify the workflow a bit so our hard drive won't suffer too much.

Moresampler (wavtool mode)

- For the first time Moresampler being called as wavtool, it creates the target .wav file, usually temp.wav unless you're rendering the whole project to some specified path.
- Moresampler doesn't write waveform into it. On subsequent calls it does nothing except appending its command line arguments to temp.wav.

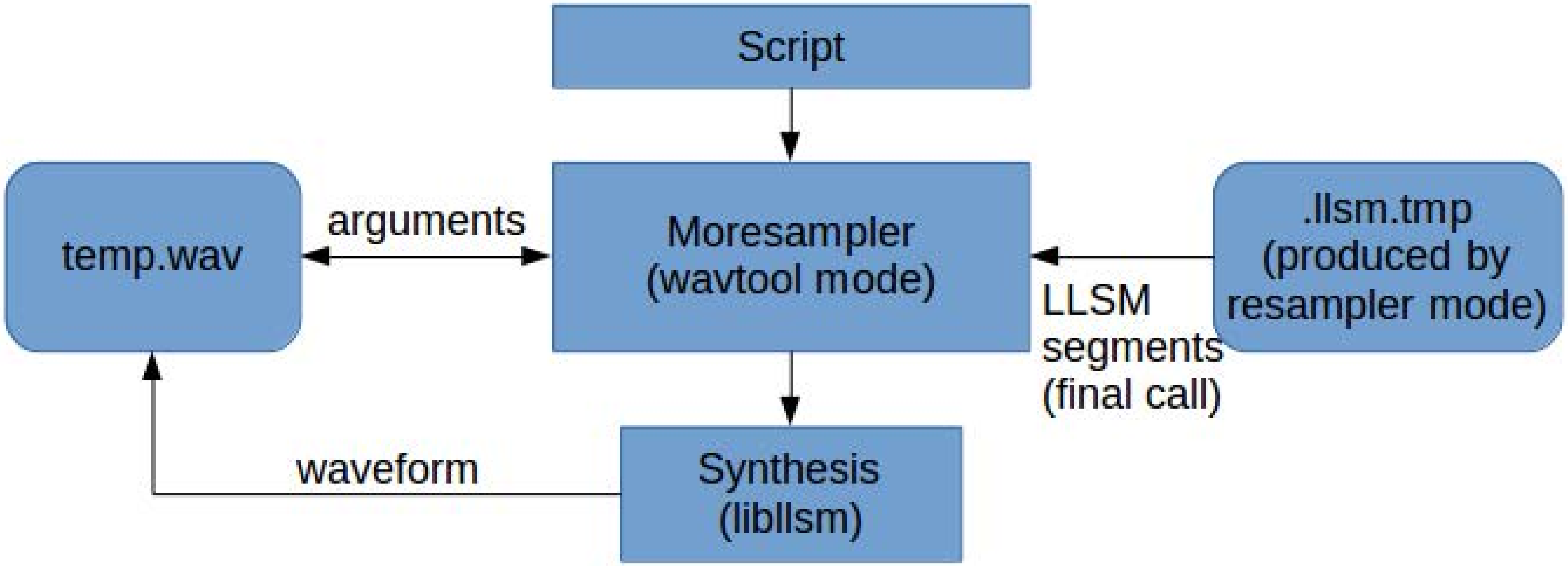


```
00000000 74 00 65 00 6D 00 70 00 2E 00 77 00 61 00 76 00 00 00 t.e.m.p...w.a.v...
00000012 43 00 3A 00 5C 00 6D 00 61 00 69 00 6E 00 2E 00 63 00 C.:.\.m.a.i.n...c.
00000024 61 00 63 00 68 00 65 00 5C 00 34 00 35 00 5F 00 7A 00 a.c.h.e.\.4.5._.z.
00000036 68 00 69 00 5F 00 46 00 23 00 34 00 5F 00 33 00 57 00 h.i._.F.#.4._.3.W.
00000048 76 00 38 00 68 00 4B 00 2E 00 77 00 61 00 76 00 00 00 v.8.h.K...w.a.v...
0000005a 00 00 00 00 00 00 00 00 FF 62 38 DB 1D D4 D9 3F 4C 0B .........b8....?L.
0000006c 6E D6 A6 D2 B0 3F 00 00 00 00 00 00 00 00 00 00 00 00 n....?............
0000007e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..................
00000090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 59 40 ................Y@
000000a2 00 00 00 00 00 00 59 40 00 00 00 00 00 00 59 40 00 00 ......Y@......Y@..
000000b4 00 00 00 00 59 40 00 00 00 00 00 00 59 40 74 00 65 00 ....Y@......Y@t.e.
000000c6 6D 00 70 00 2E 00 77 00 61 00 76 00 00 00 43 00 3A 00 m.p...w.a.v...C.:.
000000d8 5C 00 6D 00 61 00 69 00 6E 00 2E 00 63 00 61 00 63 00 \.m.a.i.n...c.a.c.
000000ea 68 00 65 00 5C 00 34 00 36 00 5F 00 67 00 65 00 69 00 h.e.\.4.6._.g.e.i.
000000fc 5F 00 46 00 23 00 34 00 5F 00 42 00 39 00 32 00 35 00 _.F.#.4._.B.9.2.5.
0000010e 49 00 47 00 2E 00 77 00 61 00 76 00 00 00 00 00 00 00 I.G...w.a.v.......
00000120 00 00 00 00 23 0F 03 DB 84 3A C0 3F B4 1A 33 05 42 1D ....#....:.?..3.B.
00000132 94 3F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .?................
00000144 00 00 00 00 00 00 00 00 89 41 60 E5 D0 22 AB 3F 00 00 .........A`..".?..
00000156 00 00 00 00 00 00 00 00 00 00 00 00 59 40 00 00 00 00 ............Y@....
```

Moresampler (wavtool mode)

- Finally, when Moresampler detects its final call by the script (it opens the script and checks if file name matches),

  It loads all arguments stored in temp.wav; then it loads all temporary LLSM files, joins them together, and calls libllsm to synthesize. temp.wav is then replaced by the output.


Script
temp.wav
arguments
Moresampler (wavtool mode)
.llsm.tmp (produced by resampler mode)
LLSM segments (final call)
waveform
Synthesis (libllsm)