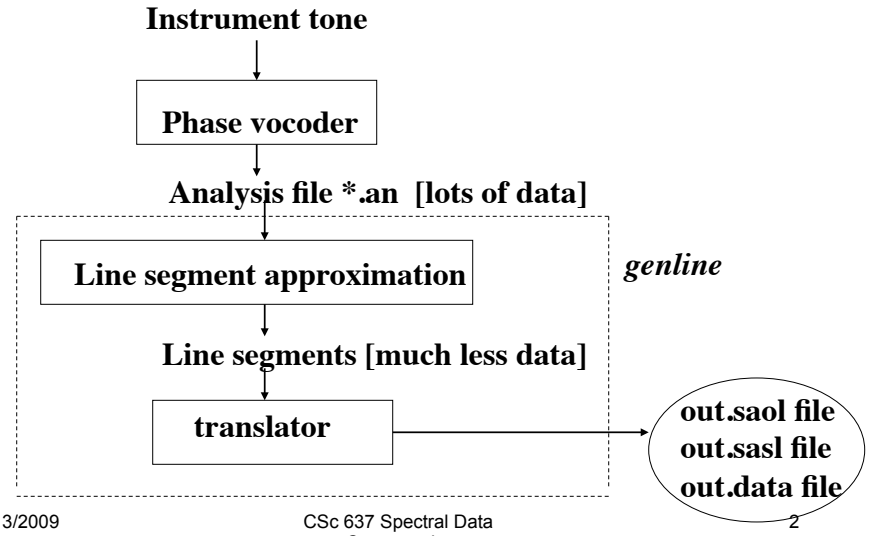
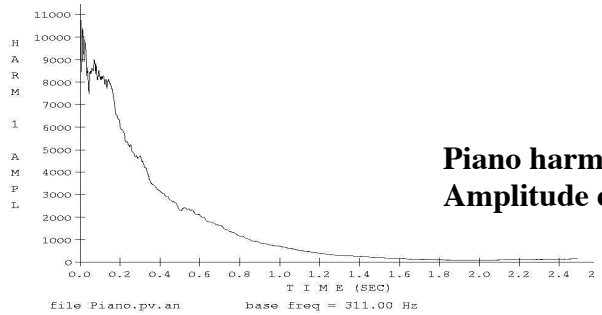
Project 2: Spectral Data Compression

Spring 2009

Working with approximations of analysis data



Example of line segment approximation



Piano harmonic 1
Amplitude envelope

Approximation with 7 data points:
(0.00 0.00) (0.002 0.308) (0.01 0.317) (0.0418 0.228)
(0.068 0.274) (0.500 0.07) (2.49 0.000)

Out.saol file excerpts

```
global {
  table cyc(harm,
  128,
  1);
srate 44100;
}
instr vtone (fr) {
imports exports table cyc;
asig y;
ksig frenv,aenv1,aenv2,aenv3,aenv4,aenv5,aenv6,aenv7;
ksig aenv8,aenv9,aenv10,aenv11,aenv12,aenv13,aenv14;
ksig aenv15,aenv16,aenv17,aenv18,aenv19,aenv20;
aenv1 = kline(0.000000,
0.001608,0.308058,
0.008039,0.316632,
0.032154,0.227911,
0.025723,0.274252,
0.432476,0.069834,
1.991961,0.000000);
```

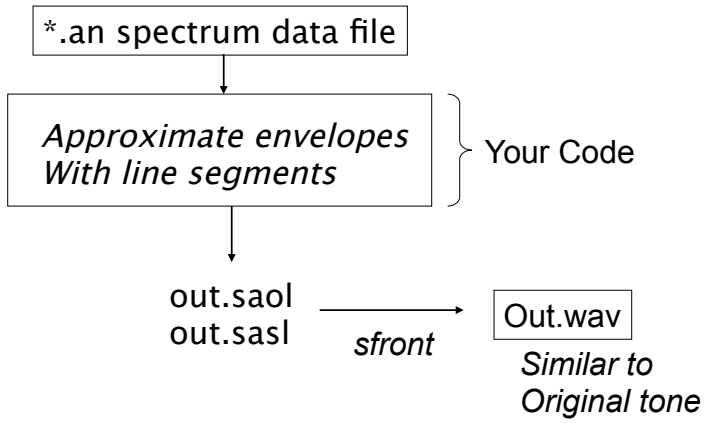
Out.saol file excerpts (con't)

```
frenv = kline(1.016*fr,
0.006431,.9903*fr,
0.003215,1.011*fr,
0.003215,.9895*fr,
0.009646,1.0058*fr,
0.598071,1.0018*fr,
1.871383,1.0018*fr);
y = aenv1 * oscil(cyc,frenv)
+ aenv2 * oscil(cyc, 2*frenv)

[etc etc]

+ aenv20 * oscil(cyc, 20*frenv);
output(y);
}
```


Out.sasl file

0.0 vtone 2.493569 311.000000
2.493569 end

Spring 2009 Project 2



Project 2 Goals

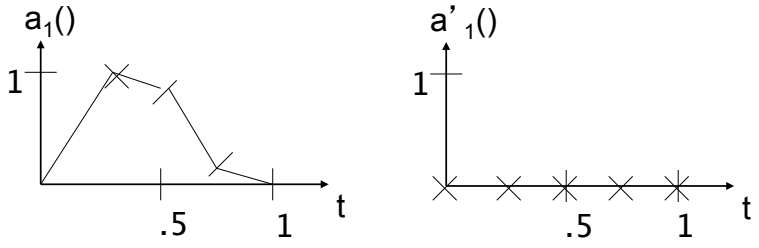
- Given: original acoustic tone (*.snd) and its spectral data (*.pv.an)
- Use first 20 harmonics only, for simplicity.
- Choose "important" data points of amplitude envelope as breakpoints
- (i.e., 20 approximated amplitude envelopes)
- All 20 harmonics share one frequency envelope (see later slides).
- Generate SAOL and SASL files
- Synthesize tone with sfront
- Result tone should be similar to original acoustic tone.

Step 1: process amplitudes

For each harmonic k,
Measured amplitudes
$a_k(0)$ to $a_k(N-1)$ for frames 0 to N-1
Estimated amplitudes
$a'_k(0)$ to $a'_k(N - 1)$ data points

Initialization: set endpoints
$a'_k(0) = a_k(0)$
$a'_k(N-1) = a_k(N-1)$ (usually zero)
Interpolate middle points with straight line

Example instrument harmonic 1 a_1 time/amplitude points:
(0,0) (0.25, 1) (0.5,0.8) (0.75, .25) (1,0)
Set end points: (0, 0) and (1, 0)
Interpolate middle points:
(0,0) (0.25, 0) (0.5, 0) (0.75, 0) (1,0)

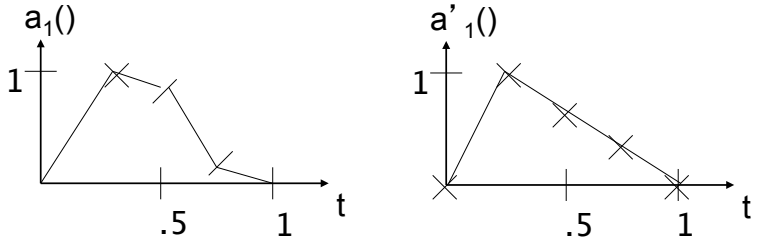


1) From remaining data points, find amplitude point i with the largest error:
   abs($a'_k(i) - a_k(i)$) >= abs($a'_k(j) - a_k(j)$) for j=0 to N-1
2) Point i becomes the new breakpoint: $a'_k(i) = a_k(i)$
3) Save new breakpoint
4) Interpolate remaining data points

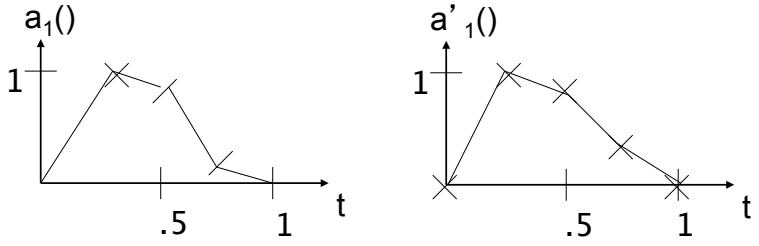
Repeat (1) - (4) until N_{bk} breakpoints are found for harmonic k

Repeat for all harmonics!

Example (continued):
Point with largest error is at t = .25
Set $a'_1(.25) = a_1(.25) = 1$
Interpolate the rest; $a'_1()$ become
(0, 0) (.25, 1) (.5, .67) (.75 , .33) (1, 0)
Breakpoints (including endpoints) are:
(0, 0) (.25, 1) (1, 0)



Example (continued):
New point with largest error is at t = .5
Set $a'_1(.5) = a_1(.5) = .8$
Interpolate the rest; $a'_1()$ become
(0, 0) (.25, 1) (.5, .8) (.75 , .4) (1, 0)
Breakpoints (including endpoints) are:
(0, 0) (.25, 1) (.75, .4) (1, 0)



Step 2: process frequency

Measured frequency envelope for harmonic k:

$f_k(0)$ to $f_k(N-1)$ for frames 0 to N-1

Combine frequency envelopes for first five harmonics into one frequency envelope:

For frame n, est. frequency envelope for harmonic 1:

$$f'_1(n) = [1/ (a_1(n) + a_2(n) + a_3(n) + a_4(n) + a_5(n))] * [a_1(n)f_1(n) + a_2(n)f_2(n)/2 + a_3(n)f_3(n)/3 + a_4(n)f_4(n)/4 + a_5(n)f_5(n)/5]$$

For harmonic k:

$$f'_k(n) = k\, f'_1(n)$$

Note: sufficient to use one frequency envelope (scaled) for all harmonics!

Then perform line segment approximation for $f'_1(n)$.

Procedure:
TBD

Step 3: generate SAOL and SASL files

Generate out.saol and out.sasl files with

- 20 amplitude envelopes with line segment approximations
- 1 frequency envelope with line segment approximations scaled for each harmonic