Adaptive FM Synthesis

Chad McKell
MUS 270A Final Project
Department of Music
University of California San Diego

December 10, 2019

Presentation Outline

1. Motivation
a) Literature review
b) Why Adaptive FM?

2. Method Overview
a) Original FM
b) Adaptive FM

3. Matlab Code
a) Variable delay
b) Lagrange Interpolation

4. Results
a) Flute modification
b) Clarinet modification

5. Future Directions
a) Real-time implementation

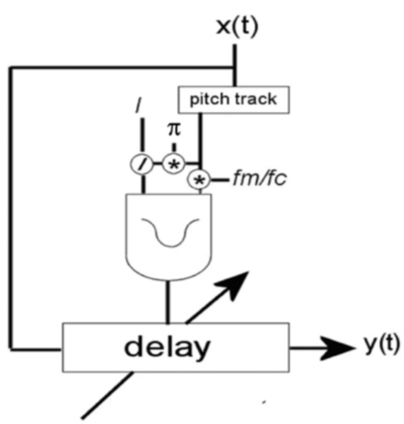
6. Questions

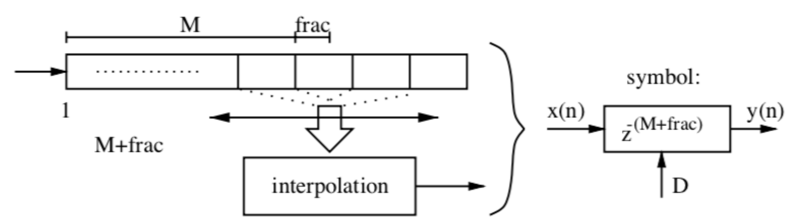
Motivation: Literature Review

– Adaptive FM synthesis introduced by Lazzarini in 2007 [1].

– Technique is based on 'audio-signal-driven' sound synthesis [2] and delay-line based phase modulation [3].

– Figures from [1] and [3]:

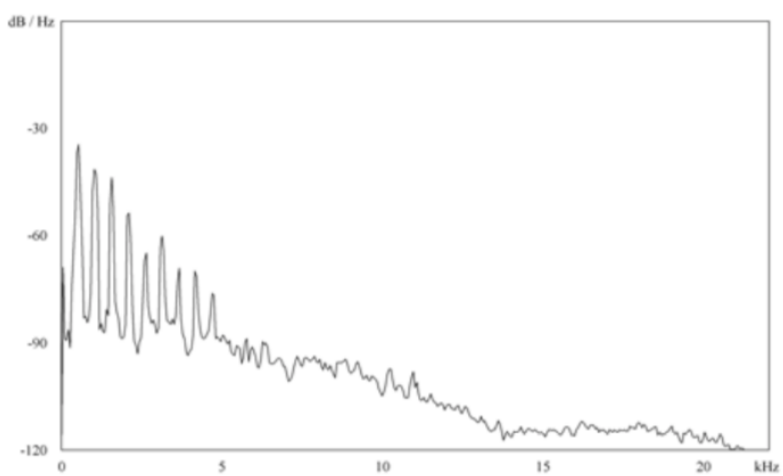


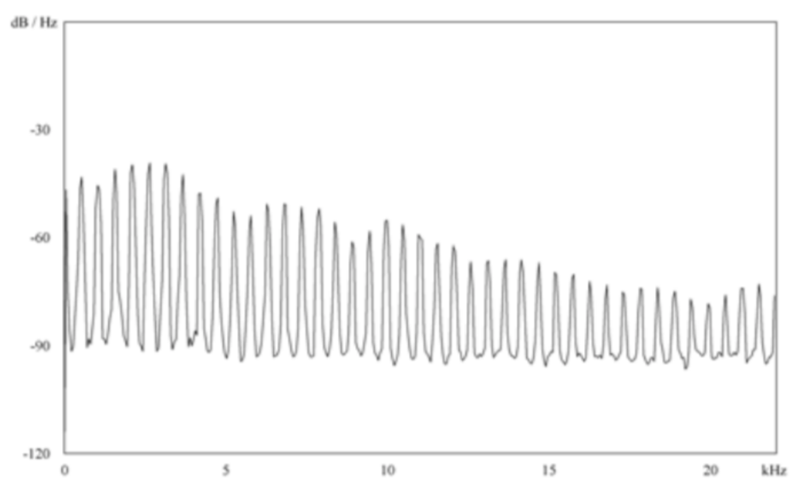


Motivation: Why Adaptive FM?

1. Add higher harmonic components to signal.

– Example from [1]: spectra of a flute playing C4 before (left) and after (right) adaptive FM:

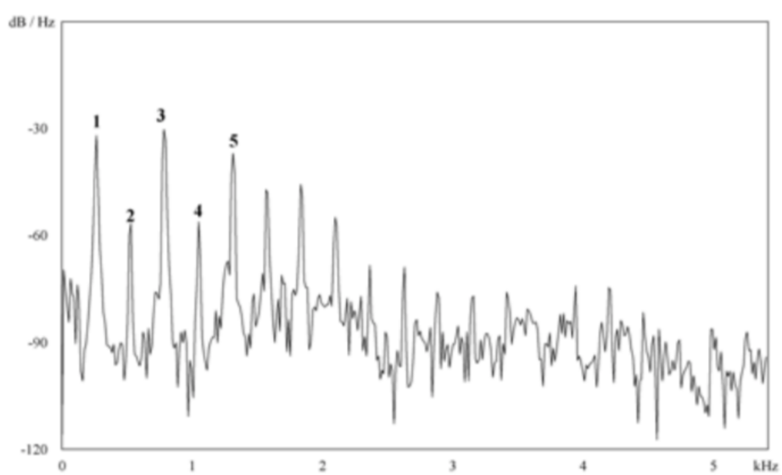


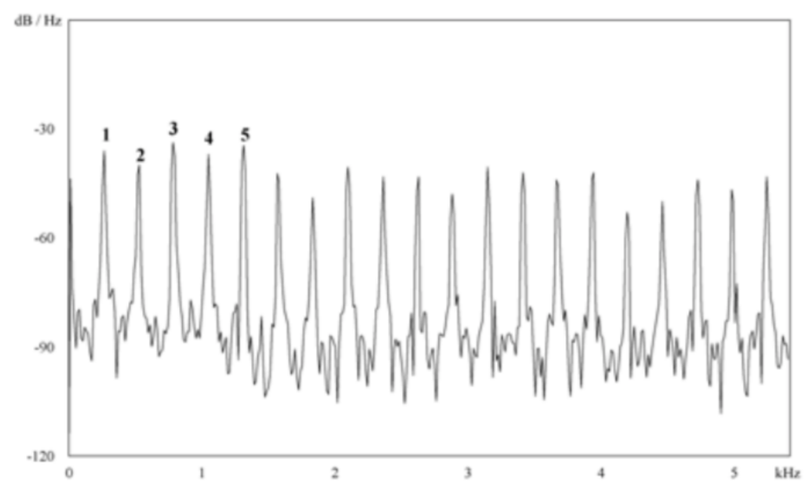


Motivation: Why Adaptive FM?

2. Change the odd-even harmonic balance of a signal.

– Example from [1]: spectra of a clarinet playing D3 before (left) and after (right) adaptive FM:





Method Overview: Original FM

In "phase modulation" form, we have:

$$y[n] \;=\; A[n]\cos\big(2\pi f_c nT + I[n]\cos(2\pi f_m nT)\big)$$

where

$$\begin{aligned} A[n] &: \text{ variable ADSR amplitude} \\ f_c &: \text{ carrier frequency (Hz)} \\ T &: \text{ sampling period (sec)} \\ I[n] &: \text{ variable index of modulation} \\ f_m &: \text{ modulator frequency (Hz)} \end{aligned}$$

Method Overview: Original FM

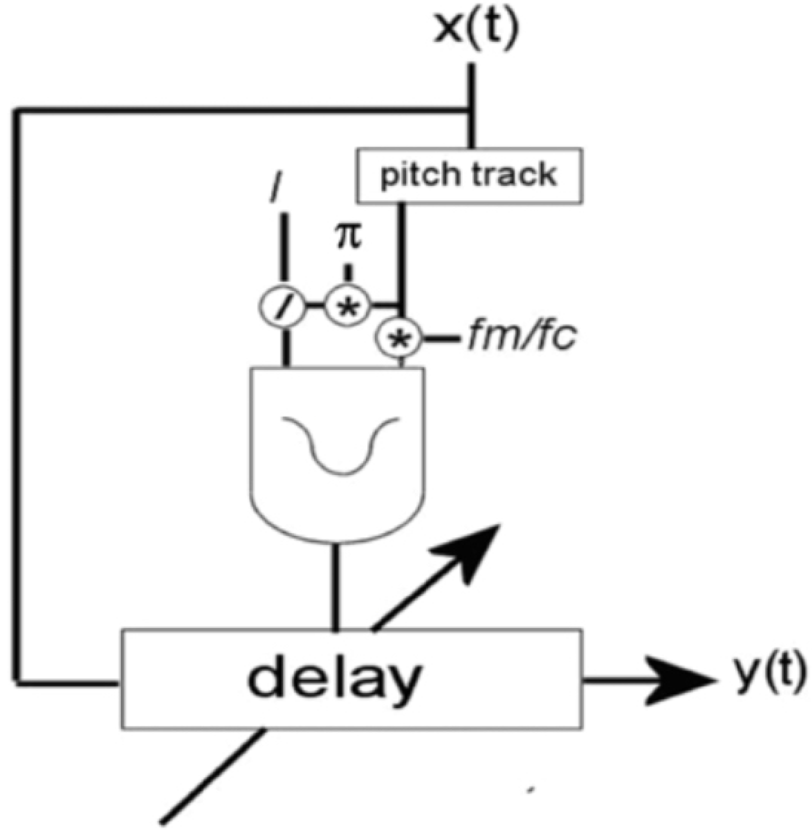
– Recall that:

$$\frac{f_m}{f_c} = \frac{N_1}{N_2}$$

where N_1 and N_2 are integers with no common factors (except 1).

– Pitch tracking is used to maintain this ratio for a live audio feed.

Method Overview: Adaptive FM



Method Overview: Adaptive FM

1. **Pitch tracking (real-time only)**

2. **Variable delay/phase modulation**

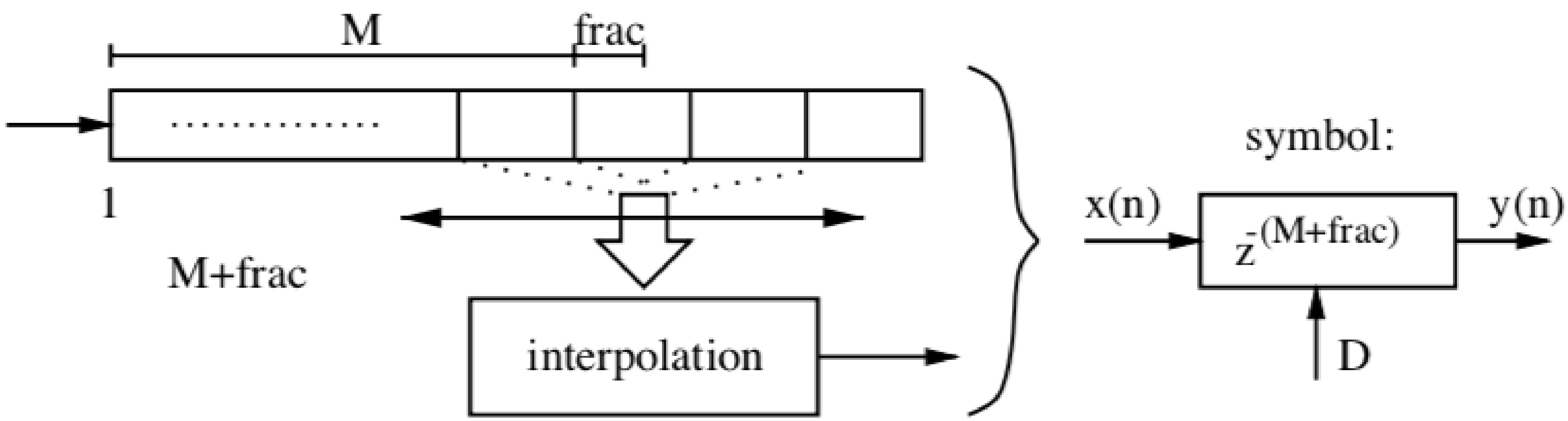
$$
\begin{aligned}
y[n] &= x[n - D[n]] \\
Y(z) &= X(z) z^{-D[n]} \\
Y(e^{j\omega T}) &= X(e^{j\omega T}) e^{-j\omega T D[n]}
\end{aligned}
$$

where

$$
\begin{aligned}
D[n] &= \mathrm{DEPTH} + \mathrm{SWING} * \cos(\omega_m n T) \\
D[n] &= \mathrm{M} + \mathrm{frac}
\end{aligned}
$$

3. **Interpolated output**

Method Overview: Adaptive FM



Matlab Code: Variable Delay

From derivation in [1], we get:

$$\begin{aligned}
D[n] &= \text{DEPTH} + \text{SWING} * \cos(\omega_m nT) \\
D[n] &= \frac{I[n]}{\pi f_c}\left(0.5 + 0.5 * \cos(2\pi f_m nT)\right) \\
D[n] &= \beta + \beta \cos(2\pi f_m nT)
\end{aligned}$$

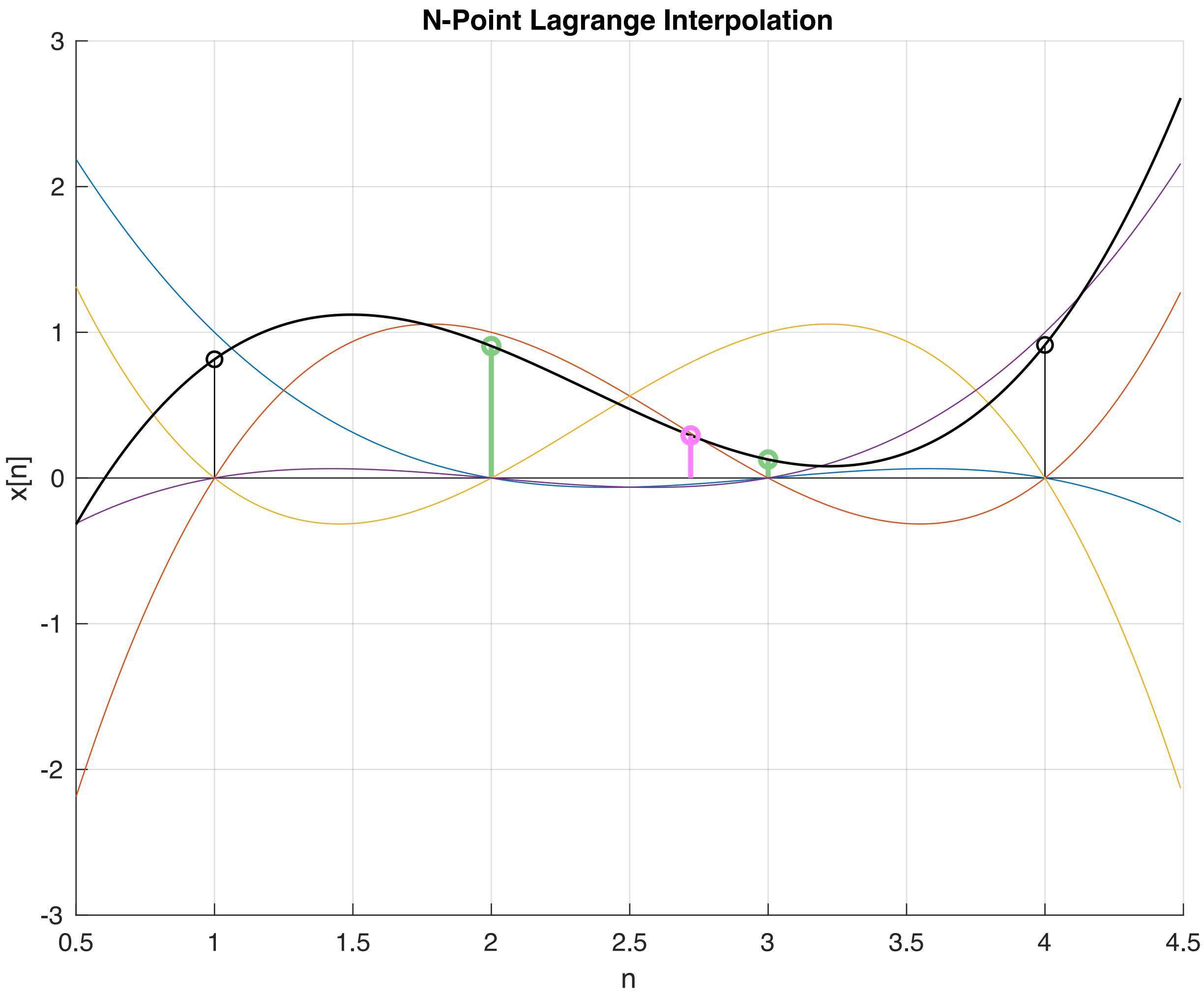
where

$$\begin{aligned}
I[n] &= I_{\max}\alpha[n] \\
\alpha[n] &: \text{ variable ADSR amplitude}
\end{aligned}$$

Matlab Code: Variable Delay

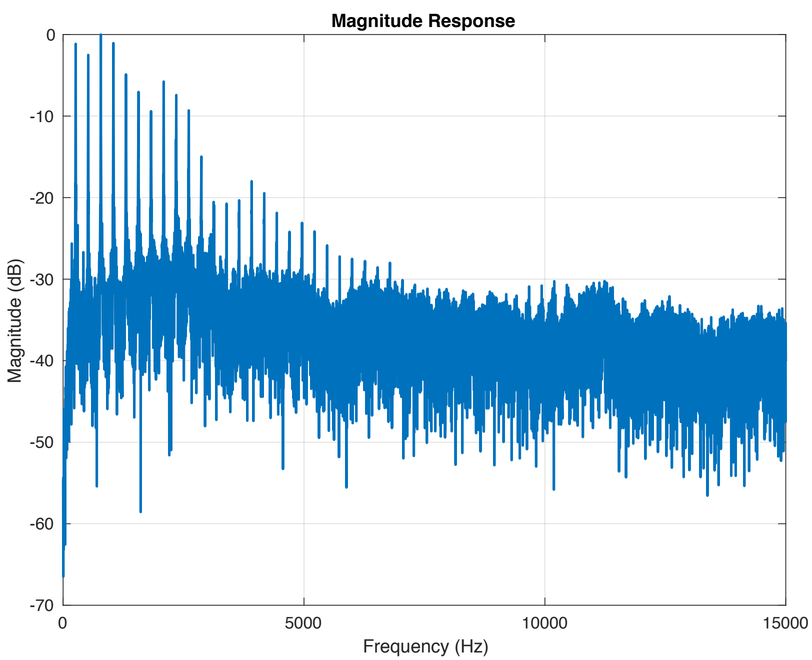
```
% Define variable delay function D
Ienv = Imax * adsr(0.05, 0.05, 0.1, 0.8, L);
beta = Ienv * 0.5*Fs/pi/f0;
D = beta + beta.*cos(2*pi*fm*n*Ts);

% Let y equal x from n=1 to n=maxD (max integer value of D)
maxD = round(max(D), 0);
y(1:maxD) = x(1:maxD);

% main loop
for n = maxD+1:L-1
    y(n) = lagrange_estimate(x(n-M(n))
end
```

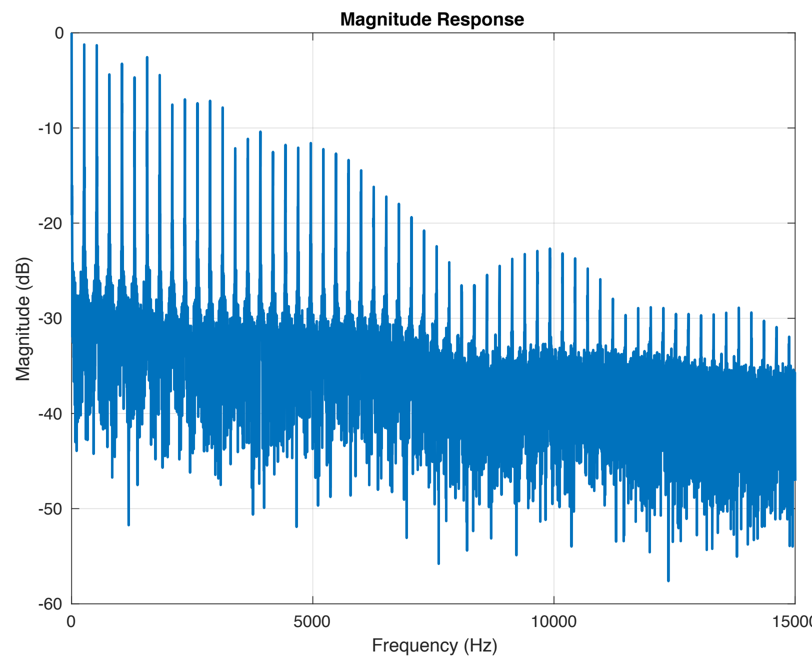
Matlab Code: Lagrange Interpolation



Results: Flute Modification

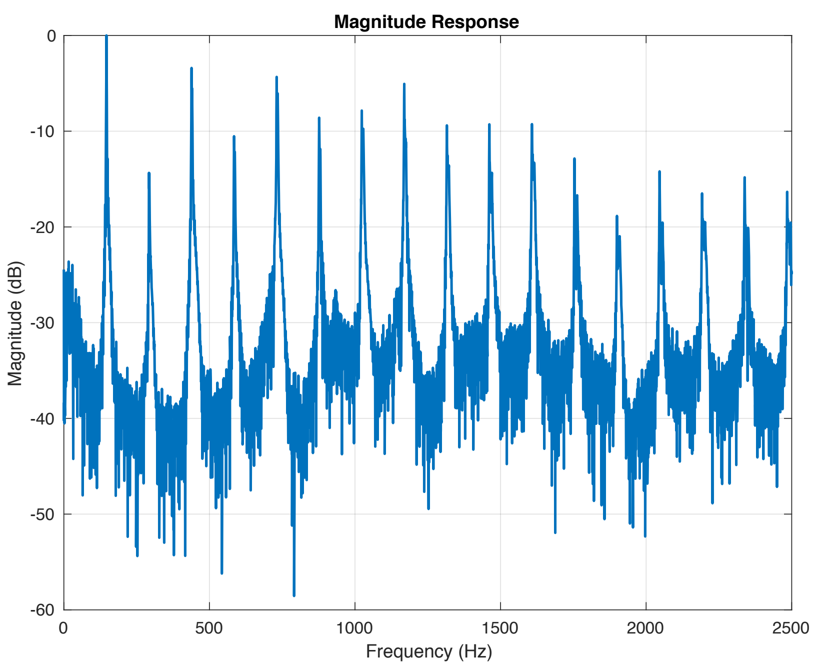
Given $I_{\max} = 2$ and $f_m/f_c = 1$, we strengthen the upper harmonics. See plots of original signal (left) and modified signal (right). Audio examples with various values for $I_{\max}$ are available here.

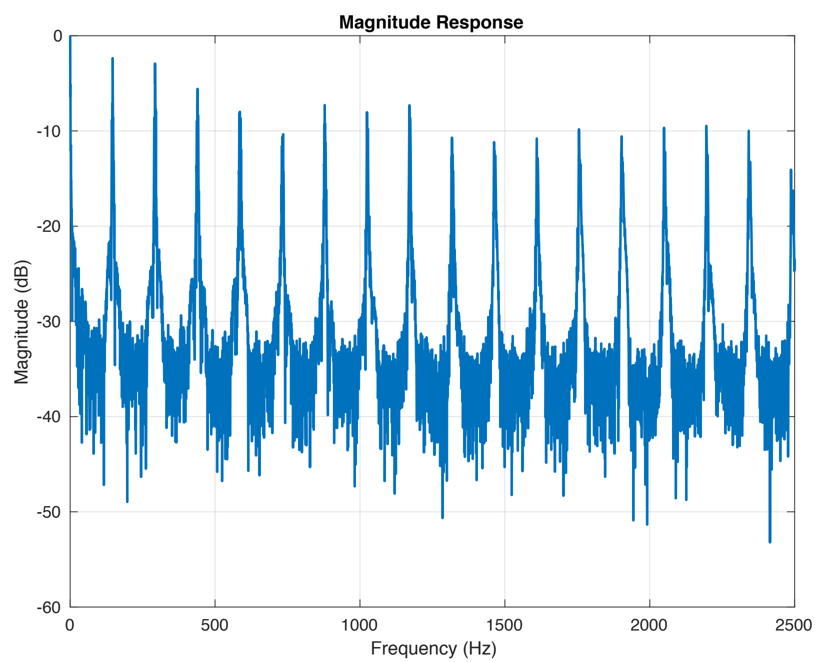




Results: Clarinet Modification

Given $I_{\max} = 2$ and $f_m/f_c = 1$, we enhance the low-order even harmonics. See plots of original signal (left) and modified signal (right). Audio examples with various values for $I_{\max}$ are available here.





Future Directions: Real-Time Implementation

1. Add pitch tracking algorithm (see paper by Puckette) in order to maintain the proper ratio f_m/f_c over time.

2. Convert sine calculation to wavetable read out in order to save on computation.

3. Translate Matlab code to C++ (use JUCE libraries)

References

(1) Lazzarini, Victor, Joseph Timoney, and Thomas Lysaght. "Adaptive FM synthesis." In Proc. 10th Conf. DAFx. (2007).

(2) Poepel, Cornelius, and Roger B. Dannenberg. "Audio signal driven sound synthesis." In ICMC. (2005).

(3) Disch, Sascha, and Udo Zölzer. "Modulation and delay line based digital audio effects." In 2nd Workshop on DAFx. (1999).

Questions