Active Noise Cancellation Headsets

Kuang-Hung liu, Liang-Chieh Chen, Timothy Ma, Gowtham Bellala, Kifung Chu

4 / 15 / 2008
Outline

- Motivation & Introduction
- Challenges
- Approach 1
- Approach 2
- Demonstration
- Conclusion & future work
Motivation

- Noise levels in human settings have come under scrutiny for reasons including health concerns and improvement of the quality of life.
- For low-frequency noises, passive methods are either ineffective or tend to be very expensive or bulky.
- Active Noise Cancellation (ANC) systems have become an effective technique for designing ANC headphones.
Introduction

- Application of adaptive signal processing.
- Use destructive interference to cancel out unwanted noise.
- ANC headsets works best for cancelling lower frequency sounds that are continuous and periodic.
- Higher frequency and impulse are hard to control.
Adaptive Filter Framework
Equivalent Model

\[ y[n] = x[n] - y[n] \]

DSK produces a signal to cancel out the noise
Challenge

- **Difficulty:**
  1. The acoustic superposition in the space (from the canceling loudspeaker to the error microphone) is sensitive to phase mismatch.
  2. ANC system is sensitive to uncorrelated noise.

- **Solution:**
  1. Compensate for the secondary-path transfer function $S(z)$, which includes the D/A converter, reconstruction filter, anti-aliasing filter, A/D converter.
  2. Use FPGA to reduce system delay.
  3. Add protection to the algorithm.
Matlab Example

- **noise**
- **observe noise**
- **generate inverse waveform**
- **error**

Matlab Example

W2008 EECS 452 / ANC headset project
Approach 1: off-line estimation of $S(z)$

- **Off-line estimation:**
  
  Send a sequence of training data to estimate $S(z)$ before Noise Cancellation.

- **Challenge:**
  
  It's better to train the filter with white noise. But because of the limited number of coefficients of the filter, we just train the filter with 200Hz sine waveform.
Adaptive Algorithms

- FxLMS
- FULMS
- Feedback
- Hybrid

![Block diagram of adaptive algorithms]

Adaptively adjust

W2008 EECS 452 / ANC headset project
FxLMS algorithm

Advantages:
- Simple and neat
- Incorporates secondary path effect
- Tolerant to errors made in estimation of $S(z)$ ➔ offline estimation sufficient

Disadvantages:
- Higher order filters ➔ slow
- Acoustic feedback
- Convergence rate depends on accuracy of the estimation of $S(z)$

Equations:
\[ y(n) = w^T(n)x(n) \]
Update weights: $w(n+1) = w(n) + \mu x'(n)e(n)$
where $x'(n) = \hat{s}(n) \otimes x(n)$
FxLMS algorithm

Advantages:
- Simple and neat
- Incorporates secondary path effect
- Tolerant to errors made in estimation of \( S(z) \)
  - offline estimation sufficient

Disadvantages:
- Higher order filters  ➔ slow
- Acoustic feedback
- Convergence rate depends on accuracy of the estimation of \( S(z) \)

Equations:
\[
y(n) = w^T(n)x(n)
\]
Update weights : \( w(n+1) = w(n) + \mu x'(n)e(n) \)
where  \( x'(n) = \hat{s}(n) \otimes x(n) \)
FuLMS Algorithm

Advantages:
- Feedback Neutralization
- Feedback path designed using IIR filter
- IIR filter – lower order sufficient

Disadvantages:
- IIR filter – can become unstable
- Global convergence not guaranteed

Equations:
\[ y(n) = a^T(n)x(n) + b^T(n)y(n-1) \]
Update weights:
\[ a(n+1) = a(n) + \mu x'(n)e(n) \]
\[ b(n+1) = b(n) + \mu \hat{y}'(n-1)e(n) \]
where \( \hat{y}'(n-1) = \hat{s}(n) \otimes y(n-1) \)
Feedback ANC

Advantages:

➢ Requires only one microphone
➢ Additional filter not required for acoustic feedback neutralization
➢ Computationally less complex

Disadvantages:

➢ Same issues with IIR filter

Equations:

\[ y(n) = w^T(n)x(n) \]

\[ x(n) = e(n) + \sum_{m=0}^{M-1} \hat{s}_m y(n - m) \]

Update weights: \[ w(n + 1) = w(n) - \mu x'(n)e(n) \]
Hybrid ANC

Advantages:
- Combines advantages of feedforward and feedback systems
- lower order filters
- relatively more stable than feedback ANC

Disadvantages:
- High computational complexity

Equations:
\[ y(n) = a^T(n)x(n) + c^T(n)\hat{d}(n) \]

Update weights:
\[ a(n+1) = a(n) + \mu x'(n)e(n) \]
\[ c(n+1) = c(n) + \mu \hat{d}'(n)e(n) \]

where \( \hat{d}'(n) = \hat{s}(n) \otimes \hat{d}(n) \)
\[ \hat{d}(n) = e(n) + \hat{s}(n) \otimes y(n-1) \]
Filter order = 32, \( \mu = 2000 \)
How to adaptively adjust the filter coefficients?

- Considering the computation time, we use LMS:
  \[ W(n+1) = W(n) + u \cdot e(n) \cdot W(n) \]
  - \( u \): step size
  - \( e(n) \): error
  - \( W(n) \): filter coefficients

- To make our system more stable, we use variations of LMS:
  (a) Leaky LMS: Introducing ‘\( a \)’ makes \( W(n) \) not change too rapidly
    \[ W(n+1) = a \cdot W(n) + u \cdot e(n) \cdot W(n), \text{ where } a < 1 \]
  (b) Normalize the coefficients \( W(n) \) to stabilize the output
    \[ W(n+1) = \frac{W(n+1)}{\sqrt{\text{sum}(W(n+1))}} \]
Approach 2: System architecture

- **DSPs**
  - Downsample to 5k
  - Adaptive filter algorithm
  - Upsample to 500k

- **FPGA**
  - 14-bits A/D converter
  - Sampling at 500kHz

- **I/O hardware interface**
  - Anti-aliasing filter
  - Microphone amplification circuit
  - Headset driving circuit

Inputs:
- \( x[n] \)
- \( e[n] \)

Outputs:
- \( y[n] \)
- \( e[n] \)
Input analog/digital interface

MIC amplification circuit

Single-Supply Sallen Key Low Pass Butterworth Filter with cut-off freq 7.5k
Output digital/analog interface

Amplification with gain 20
Low pass filter

MIC amplification circuit

Headphone driving circuit

to FPGA

x[n] → e[n] → y[n] → to headset

from FPGA

x[n] → e[n] → to FPGA
Adaptive filter

Initialization

c_t > 64

error_ac > error_last

error_ac > 3 * error_min

yes
reset_c_t++

no

w ← w_bk

reset_c_t > 60

yes
reset w

no

w_bk ← w
error_last = error_ac
update error_min

ct = ct + 1
accumulate error_ac

adaptively adjust filter coeff w using LMS algorithm
Sampling rate & filter size
design consideration

- Sampling rate determines processing speed constraint.
- Fix sampling rate
  - Large filter size: fine freq resolution, slow processing, slow response.
  - Small filter size: low freq resolution, fast processing, fast response.
- Fix filter size
  - High sampling rate: low freq resolution, less artifact in the D/A output.
  - Low sampling rate: high freq resolution, more artifact in the D/A output.
Experiment result (1)
Experiment result (2)

freq
mag

300 Hz

x[\text{n}]

w[\text{n}]
Experiment result (3)

![Graph showing frequency response with two peaks at 500 Hz.]

- Graph shows frequency response with two peaks at 500 Hz.
- Axes labeled: freq on the x-axis and mag on the y-axis.
- Two lines represented: x[n] (solid blue) and w[n] (dashed red).

W2008 EECS 452 / ANC headset project
Experiment result (4)
Incorporate music source

- Assume music source is uncorrelated with the noise, and noise is a zero mean w.s.s random sequences.

- Adaptive algorithm will adjust filter coefficient so as to minimize MSE.
  \[ E[e^2] = E[(d - Y)^2] \]
  \[ = E[(m + y - Y)^2] \]
  \[ = E[m^2] + E[(y - Y)^2] + 2E[(y - Y)m] \]

  \[ = 0 \quad Y, X \text{ uncorrelated with } m \]

- Since \( E[m^2] \) does not depend on filter coeff, adaptive algorithm will select coeff to minimize \( E[(y - Y)^2] \).
ANC system demonstration
part 1:

Single tone and multiple tone artificial noise
ANC system demonstration

part 2:

Real engine noise
ANC system demonstration
part 3:
Noise with music source
Future work (1)

Our ANC system

input

LPF (1kHz)

from microphone

to headset

output
Future work (2)

- Use more precise microphone and ADC, DAC to acquire more accurate measurement.
- Integrate the design components into a build-in embedded system to avoid feedback interference.
- Implementation in assembly language to save computation time.
Conclusion

- A workable ANC headset for both artificial and real world noise.
- Works for noise frequency ranging from 100 to 800 Hz.
- Incorporate music source.
- Implemented and compared LMS, FxLMS, Feedback, FuLMS and Hybrid algorithms:
  - For stability, Hybrid is the best.
  - For simplicity, FxLMS is recommended.
Thank you
References