Acoustic echo cancellation using the LMS adaptive filter

Description

Acoustic echo cancellation is a signal processing technique used to suppress echo effects in hands-free communication systems. In this approach, the signal played through the loudspeaker is treated as a reference input, while the microphone receives a combination of the reflected echo, the speaker's voice, and ambient noise. An adaptive filter based on the Least Mean Squares (LMS) algorithm is employed to estimate the unknown acoustic path between the loudspeaker and the microphone. Initially, the filter coefficients are set to zero and are updated iteratively as the system operates. At each sampling instant, the adaptive filter produces an estimated echo by processing the reference signal. This estimated echo is subtracted from the microphone signal, resulting in an error signal that primarily contains the desired near-end speech. The LMS adaptation rule uses this error signal to adjust the filter coefficients in a way that minimizes the mean square error. As the adaptation progresses, the filter gradually converges toward the actual echo path, leading to effective suppression of acoustic echo. The algorithm functions continuously in real time and remains reliable even in noisy environments and during simultaneous speech, thereby improving speech intelligibility in hands-free systems.

Algorithm

```
import random

N = 2000

L = 20

mu = 0.01

noise_level = 0.05

x = [random.uniform(-1, 1) for _ in range(N)]

h = [0.9, 0.6, 0.3, 0.15, 0.05]

echo = [0.0] * N

for n in range(N):
```

```
    for k in range(len(h)):
        if n - k >= 0:
            echo[n] += h[k] * x[n - k]
near_end = [0.3 * random.uniform(-1, 1) for _ in range(N)]
d = [echo[i] + near_end[i] for i in range(N)]
noise = [noise_level * random.uniform(-1, 1) for _ in range(N)]
d = [d[i] + noise[i] for i in range(N)]
w = [0.0] * L
y = [0.0] * N
e = [0.0] * N
iterations = 10
for _ in range(iterations):
    for n in range(L, N):
        y[n] = 0.0
        for k in range(L):
            y[n] += w[k] * x[n - k]
        e[n] = d[n] - y[n]
        for k in range(L):
            w[k] = w[k] + mu * e[n] * x[n - k]
echo_power = 0.0
error_power = 0.0
for i in range(N):
    echo_power += echo[i] * echo[i]
    error_power += e[i] * e[i]


echo_power /= N
```

```
error_power /= N
print("Echo power before cancellation :", echo_power)
print("Residual power after cancellation (with noise):", error_power)

if error_power < (echo_power * 2):
    print("SUCCESS: Echo significantly reduced (noise present)")
else:
    print("WARNING: Echo cancellation not sufficient")
```

The program employs the LMS adaptive filtering method to simulate an acoustic echo cancellation system. A random signal models the audio heard through the speaker. The acoustic echo path is simulated by employing a static impulse response that replicates sound reflections within a confined space. The echo effect is created by processing the sound from the speaker through this impulse response. By incorporating a low-level random signal to mimic near-end speech, a dual-talk situation is created. Enhanced ambient sounds are incorporated to enhance realism. The microphone signal is created by merging the echo, near-end speech, and noise elements. A filter with zero coefficients is employed to estimate the echo in the microphone signal. The estimated echo is removed from the microphone input to derive an error signal, primarily consisting of the desired speech and residual noise. The LMS algorithm modifies the filter coefficients incrementally to reduce the mean square error. Performance is gauged by assessing the average power of the echo against the power of the residual error signal.

Results and visualizations

```
=== RESTART: C:/Users/ADMIN/AppData/Local/Programs/Python/Python313/prob1.p
Echo power before cancellation : 0.43068070871228686
Residual power after cancellation (with noise): 0.032923206693408605
SUCCESS: Echo significantly reduced (noise present)
```

The program's result indicates that the echo cancellation technique is successful. Before implementing the adaptive filter, the echo signal shows considerable power, suggesting significant acoustic linkage between the speaker and microphone. Following the application of the LMS-based echo cancellation technique, the residual signal power noticeably diminishes, despite the ongoing presence of background noise. This decrease indicates that the adaptive filter successfully acquires the echo path and eliminates a significant portion of the echo. The program's success message reveals that echo suppression has been successfully implemented under actual operating scenarios, leading to enhanced speech clarity.

Discussion and Conclusion

The findings indicate that the LMS-based adaptive echo cancellation method successfully minimizes acoustic echo in hands-free communication setups. The notable reduction in signal strength following processing indicates that the adaptive filter can dynamically adapt to the acoustic setting and nearly match the echo path. Despite the presence of near-end speech and background noise, the system maintains its reliability. The convergence speed varies depending on parameters like filter length and step size. Erroneous choices of these parameters could compromise system stability or hinder adaptation in dual-talk scenarios. Although the LMS algorithm is straightforward, it boasts low computational requirements and real-time functionality, thus being appropriate for practical implementations. In summary, this approach showcases the effectiveness of adaptive filtering in reducing acoustic echo and improving speech clarity, setting the stage for future enhancements through the use of more sophisticated algorithms.