ANC Matlab Code Documentation

Changrong Liu

October 2024

1 Description of core functions

ALMS N.m (LMS filter with order N)

input:
primary: primary signal $x(t)$;
reference: reference signal $r(t)$;
order: order the filter (length of tap-weight vector).

output:
cancelled: cleaned up signal $\hat{x}(t) = x(t) - \hat{c}(t)$;
adap: weight vector of the adaptive filter $\mathbf{w_k}$ when converges;
fit: estimate of clutter $\hat{c}(t)$.

ARLS N.m (RLS filter with order N): basically the same as LMS except the updating rule of the filter weights.

input:
lambd: forgetting factor in RLS

output:
P: equation (16) when converges.

ANC filter.m: (input simulate data, output a Viterbi score for claiming a detection or not; can replace simulate data with real data)
inputs: h, f0, ampr0, ampr, H, Q, gamma are parameters to be specified. e.g, amplitude of GW signal, base-line frequency, amplitude of noise, time delays. These parameter are input arguments of **simulate data.m**.

viterbi for ANC.m(input time series data, output viterbi score and estimated frequency lines)

input:
gamma: standard deviation of GW freq

Y1: cleaned signal $\hat{x}(t)$

N: number of frequency bins (decides frequency resolution for performing a FFT).

Nb: number of blocks (how many segments are we going to divide $\hat{x}(t)$ for HMM)

T: sampling time interval (determined by N, W)

W: sampling frequency (decides the range of spectrum to be tracked in HMM)

2 For demonstration

All the other functions including **script.m** (for one single simulation – generate one point on the roc curve and one estimated frequency trace); **roc curves.m** and **untitled.m** are both for multiple simulations and then plotting the whole roc curve.