Adaptive Filtering Optimization Function (AFOF)
for Matlab®
User Guide

Citation:

If you use ***AFOF***, please cite this article:

Martins, D.; Batista, A.; Mouriño, H.; Russo, S.; Esgalhado, F.; dos Reis, C.R.P.; Serrano, F.; Ortigueira, M. ***Adaptive Filtering for the Maternal Respiration Signal Attenuation in the Uterine Electromyogram****. Sensors 2022, 22, 7638. https://doi.org/10.3390/s22197638*

Global Description:

The ***AFOF*** was developed to help Matlab users to obtain the optimal adaptive filters and their parameters for a specific application. To run this function, ***Signal Processing*** and ***DSP System*** Toolboxes are necessary.

For this function, it is mandatory to use **synthetic data** closely related to real data. The output of the function will be a ranking of the best filters and their parameters for the application under study. Subsequently, the user should develop a short code for the application of the selected filter to the real data under analysis. The rationale behind the need to use synthetic data is explained in the paper above.

- The available filters to be studied are: Least Mean Square **(LMS)**, Frequency Domain Adaptive Filter **(FDAF)**, Block Least Mean Square **(BLMS)**, Filtered-X Least Mean Square **(FXLMS)**, Normalized Least Mean Square **(NLMS)**, Sign-Error Least Mean Square **(SELMS)**, Sign-Data Least Mean Square **(SDLMS)**, Sign-Sign Least Mean Square **(SSLMS)**, Recursive Least Squares **(RLS)**, Least Squares Lattice **(LSL)**, Householder Recursive Least Squares **(HRLS)**, Sliding-Window Recursive Least Squares **(SWRLS)**, Householder Sliding-Window Recursive Least Squares **(HSWRLS)**, QR-Decomposition Recursive Least Squares **(QRD-RLS)**, Fast Transversal Filter **(FTF)**, and Wiener.
- The available parameters to be optimized are filter length ($\boldsymbol{L}$), step size ($\boldsymbol{\mu}$), and forgetting factor ($\boldsymbol{\lambda}$).

To determine the optimized parameters for each filter and the best-performing filter algorithm for a specific application, the ***AFOF*** uses three different metrics: Root Mean Square Error (RMSE) between the pure signal and the filtered one; RMSE between spectrograms of the pure signal and the filtered one; and RMSE between marginals of the pure signal and the filtered one. For more details regarding the RMSE, see the above paper.

Syntax:

[rmse_signal_optimized_results, rmse_spectro_optimized_results, rmse_marginal_optimized_results] = filter_parameters_optimization_exchange*(pure_signal, noise_signal, fmax, fs, limits_stepsize, limits_L, limits_forgetting_factor, filters_type, bar_plot, additional_figures)*

Inputs:

The ***AFOF*** inputs are:

- ***pure_signal***: interest signal (signal without noise). The ***pure_signal*** should be simulated using the following criteria:

  - To be similar to the real denoised signal. This is the signal supposedly without noise. For instance in electrocardiogram (ECG) denoising applications, the pure signal should be a synthetic ECG without noise components. In this case, the user might use an ECG simulator application. However, the simulated ***pure_signal*** should be similar as possible to the real clean signal.
  - The number of samples of the ***pure_signal*** should be selected having into consideration the computational load and reasonability. Longer signals incur higher processing times. However, the signals should be long enough to be representative. For instance, in the ECG example, it would be expected the signal to have at least 15 to 20 beats.

- ***noise_signal***: interference signal that the user wants to attenuate. The ***noise_signal*** should be simulated using the following criteria:
  - This signal should be simulated to be similar to the expected real signal interference. For example for mains noise reduction in the ECG, a sinus-like wave should be generated with the correct frequency (50 Hz or 60 Hz). If harmonics are presented a sinus combination of them should be used.
  - The amplitude of the noise signal should be set realistically. Although adaptive filters typically try to compensate for higher amplitudes, it is, however, possible to have filter instability for unrealistically high noise amplitudes. For instance, for the ECG example, the mains noise amplitude should be 1/10 to 1/50 of the ***pure_signal*** amplitude.

- ***fs:*** sampling frequency of the simulated signals (in Hz).

- ***fmax:*** the maximum frequency of the spectrogram (in Hz). It should be less or equal to the value of the parameter (fs/2). The spectrogram is used to obtain the marginals from where RMSE is also obtained. The spectrogram plotting function is disabled, at the moment, but the users should fill this field.

- ***limits_stepsize***: it is necessary to insert a vector with the following configuration [lim_min_stepsize,interval_stepsize,lim_max_stepsize].
  - For default value, type []. The default value will test [0.05,0.05,1] interval.
  - For the NLMS filter, the maximum step size is internally set to 2 (please, see the dsp.LMSFilter Matlab® documentation for more details).

- ***limits_L***: it is necessary to insert a vector with the following configuration [lim_min_L,interval_L,lim_max_L].
  - For default value, type []. The default value will test [5,5,200] interval.

- ***limits_forgetting_factor***: it is necessary to insert a vector with the following configuration [lim_min_forgetting_factor,interval_forgetting_factor,lim_max_forgetting_factor].
  - For default value, type []. The default value will test [0.900,0.001,1.000] interval.
  - For the FTF filter, the minimum and the maximum forgetting factor values are internally set to 1-0.5/L and 1, respectively (please, see the dsp.FastTransversalFilter Matlab® documentation for more details).

- ***filters_type***: it allows the selection of the filters that will be tested. The possible values are: "LMS"; "FDAF"; "BLMS"; "FXLMS"; "NLMS"; "SELMS"; "SDLMS"; "SSLMS"; "RLS"; "LSL"; "HRLS"; "SWRLS"; "HSWRLS"; "QRD-RLS"; "FTF"; "Wiener". To fill this input, you must use the following configuration ["*filter_type_1*"; "*filter_type_2*";…]. If you want to study all the filters, type ["all"].

    - For default value, type [], which is equivalent to ["all"].

- ***bar_plot***: if you type 'yes', this input allows making bar plots related to the three different RMSE metrics for each filter and all the filter parameters tested. Otherwise, you can type 'no'. It is mandatory to fill up this input. These plots allow for a general overview of the filter's behavior.

- ***additional_figures***: if you type 'yes', this input will enable making plots related to the filtered signal, the spectrograms, and the marginal according to the RMSE between signals metric. Otherwise, you can type 'no'. It is mandatory to fill up this input. The additional figures plotting are disabled, at the moment, but the users should fill this field with 'yes' or 'no'. If you need these plots available, please contact me (see contact details section).

Outputs:

For each metric, the ***AFOF*** returns a table output with the following information: filter name, optimized L, optimized μ, optimized λ, and respective RMSE metric value. In total, three tables are returned by the program to be inspected by the user. These variables are:

- ***rmse_signal_optimized_results***: table output with a ranking for the filters and their optimal parameters using RMSE between signals metric. This metric is prone to be the most significant one since it deals directly with the differences between the pure signal and the filtered one.
- ***rmse_spectro_optimized_results***: table output with a ranking for the filters and their parameters using RMSE between spectrograms metric. This is an indirect metric using spectrograms.
- ***rmse_marginal_optimized_results***: table output with a ranking for the filters and their parameters using RMSE between marginals metric. This is an indirect metric using frequency marginal obtained from the spectrograms.

Contact details:

If you have any queries or if you detect any bugs please fill free to send me an email at: dsds.martins@campus.fct.unl.pt

Thank you for using the ***AFOF***,

Daniela Martins