This file contains the list of MATLAB program/script names that generated the figures in this paper.

The root folder name is “FinalWorkAdaptiveNotchFilter.”

8. Figure #8 is generated by script \Jian2009IIR\_ANFTracking\ MainTestFile.m. It is discussing about the comparison of the proposed technique and one of the state of the art ANF (IIR second order discrete model).

1. Figures 1, 2 and 3 are generated by the program file, NonUniformTest.m, with different adjustments of signal frequency (60/170), corresponding 10% error in initialization, and position of labels the figures in the paper can be obtained. It uses two function files: FourthOrderANFFixedBlock.m for uniform sampling and NonUniform4thOrderANFFixedBlock.m for non-uniform samples.
2. Figures 4, 5 are generated by the program files: ComparingOrdersDerivatives\_Nonuniform.m (main program),NonUniformSecondOrderANFFixedBlock.m, NonUniformThirdOrderANFFixedBlock.m, NonUniform4thOrderANFFixedBlock.m, are the function files for 2nd , 3rd and 4th order approximations to the solution of the dynamical system.

Some adjustments like selection of range of frequencies are needed for generating both figures.

1. Figures 6,7 are generated by the program files: ComparingCLRBANF.m and crlb\_test1.m

Compare file generates the comparison for mean and variance of the estimation algorithm and the next file simply computes the CLRB lower bound according to the derived formula.

1. Figures 8-12 are generated by PoinCareMap\_PhasePotrait.m and the function file FourthOrderANFFixedBlock.m which uses 4th order approximation for the solution to the dynamical system. Fig 8 describes the signal, fig 9-12 the Poincare maps.
2. Figure 13 – 14.

Used program files and their function description:

ANFNonUniformComparingAdomianAndAgent.m - It is the main program comparing Runge-Kutta, Adomain decomposition method and our proposed method.

ANFAdomianSpecialFunctionGenerator.m - It generates the Adomain Polynomials as defined in the paper.

FunctionalANFAdomianSpecial.m - Is the function that computes the solution using Adomain decomposition as defined in the paper.

RungeKutta4thOrderANF.m - Is the function that computes the solution of the given dynamical system using 4th order classical formula.

1. Figures 15 is generated using Microsoft PPT file, block diagram for High Frequency estimation.

HighFrequencyTrick.m is the program that implements the above mentioned block diagram and estimates a very high frequency f = 870 Hz. Fig 16 is generated using it.

1. Figure 17 is generated again using Microsoft PPT file block diagram for the extension of our method.

Additional program files and their function description that generated Figure 18 are:

ThirdOrderMojiriANFBlock.m - Is the function that computes the solution of the dynamical system using 3rd order approximation.

FunctionMarinoAdaptive.m implements the dynamical system described in Marino-Tomei paper (2002).

FunctionXiaAdaptive.m implements a similar dynamical system described in Xia paper (2002).

TestXiaMarinoOurHsu.m finally is the program that does the comparison and generate the figure 18.

1. Figures 19-20 is generated by ANFHopsStudy.m.
2. Figures 21-22 is generated by Chirped\_Sinusoid.m
3. Figures 23-24 is generated by Mega\_Chirped\_Sinusoid.m
4. Figures 25-27 are Microsoft PPT files for description of block diagrams in the paper.
5. Figures 28- 30 are generated by WithPreFilters2Freq.m
6. Figures 31-33 are generated using PathologicalCaseWithPreFilters2Freq.m. With different changes in amplitudes as discussed in the paper all three scenarios can be produced with a single file.
7. Figures 34-35 are generated using WithPreFiltersFiveFrequenciesHsuAgentsHarmonics.m file.
8. Figures 36-38 are generated using WithPreFiltersSevenArbitrary.m file.