

# Description of the library of functions for approximating a signal partition

The library contains MATLAB functions implementing the methods described in the paper:

COOPERATIVE GREEDY PURSUIT STRATEGIES FOR SPARSE SIGNAL REPRESENTATION BY  
PARTITIONING

Laura Rebollo-Neira  
Mathematics Department  
Aston University

The whole library is available for download on the website

<http://www.nonlinear-approx.info/examples/node01.html>

All the functions are dedicated to be applied with trigonometric dictionaries and take advantages of the FFT.

The main directory is named **Cooperative**. It is split into four subdirectories:

**HBW-Pursuit**

**Num-Exam-OMP**

**Num-Exam-OOMP**

**Signals**

**Tools** Each of the above subdirectories contains the files below.

---

## HBW-Pursuit

Contains the functions:

[OHBW-OMP-FFT.m](#)

[OHBW-OOMP-FFT.m](#)

[HBW-BOOMP.m](#)

[HBW-SR-OMP-FFT.m](#)

[HBW-SR-OOMP-FFT.m](#)

[fast-choose-atom.m](#)

[fast-choose-atom-oomp.m](#)

[OOMP-FFT.m](#)

[OMP-FFT.m](#)

[OHBW-OMP-FFT.m](#) Implements the Hierarchized Blockwise OMP method, revised as discussed in *Proposition 1, Sec 2*.

[OHBW-OOMP-FFT.m](#) Implements the Hierarchized Blockwise OOMP method, *c.f. Sec 2.1*.

[HBW-BOOMP-FFT.m](#) Implements the Hierarchized Blockwise Backwards OOMP method *c.f. Proposition 2, Sec 2.2*.

[HBW-SR-OMP-FFT.m](#) and [HBW-SR-OOMP-FFT.m](#): Implement the Hierarchized Blockwise Swapping Refinement of OMP and OOMP *c.f. Sec 3*. They use [BioFor2.m](#) and [BioBack2](#) for upgrading/downgrading biorthogonal and orthogonal vectors.

The routines:

[fast-choose-atom.m](#)

[fast-choose-atom-oomp.m](#)

are needed for the dedicated implementations with trigonometric dictionaries using FFT.

The routines:

[OMP-FFT.m](#)

[OOMP-FFT.m](#)

are included to enable comparisons with the proposed HBW versions of both methods.

---

## **Num-Exam-OMP**

Contains the functions

[Num-Exa-I-OMP.m](#)

[Num-Exa-II-OMP.m](#)

[Num-Exa-III-OMP.m](#)

[Num-Exa-IV-OMP.m](#)

[Get-Example-I-OMP-Settings.m](#)

[Get-Example-II-OMP-Settings.m](#)

[Get-Example-III-OMP-Settings.m](#)

[Get-Trig-Dictionaries.m](#)

The scripts [Num-Exa-I-OMP.m](#), [Num-Exa-II-OMP.m](#), [Num-Exa-III-OMP.m](#), [Num-Exa-IV-OMP.m](#) run the **Numerical Examples I, II, III, IV**, in the paper, all corresponding to the OMP strategy. By default the scripts run with a mixed cosine/sine dictionary redundancy 4 (the option of best performance in all the senses). For other options change the parameters in the corresponding [Get-Example-\(‘number’\)-OMP-Settings.m](#) file, as indicated there.

---

## **Num-Exam-OOMP**

Contains files with the same description as **Num-Exam-OMP** but corresponding to the OOMP strategy

---

## **Signals**

Contains the files

[pno-cs.wav](#)

[flute.wav](#)

[piazzola4-piece.wav](#)

which are the test signals in Figures 1, 2 and 3, respectively, used in the numerical examples.

## **Tools**

Contains auxiliary functions