

SMALLbox cheat sheet

Startup

Run `SMALLboxInit` to add all the subdirectories of SMALLbox to the Matlab path and create the `SMALL_path` variable to the SMALLbox root.

Structures

Problem

`A` – a matrix or operator representing a dictionary in which the signal is sparse

`b` – a vector or matrix representing the signal or signals to be represented

`reconstruct` – a function handle to reconstruct the signal from coefficients

`signalSize` – the dimension of the signal

`sizeA` – if matrix `A` is given as an operator the size of the dictionary needs to be defined in advance.

`p` – number of atoms in the dictionary (for dictionary learning problems only)

Solver

`toolbox` – a field with toolbox name (e.g. `sparselab`)

`name` – the name of solver from the particular toolbox (e.g. `SolveOMP`)

`param` – the parameters in the form given by the toolbox API

`solution` – the output representation

`reconstructed` – the signal reconstructed from the solution

`time` – the time taken in calculating the sparse representation.

Dictionary learning (DL)

`toolbox` – a field used to discriminate the API

`name` – the name of dictionary learning function from the particular toolbox

`param` – a field containing parameters for the particular DL technique and in the form given by the toolbox API

`D` – a field where the learned dictionary will be stored

`time` – a field to store the time elapsed during the learning stage.

Usage

Create structures

```
problem = create_problemName_problem(parameters)
```

Creates a problem structure.

```
solver = SMALL_init_solver(toolbox, name, param, profile)
```

Creates a solver structure with the given values of `toolbox`, `name`, `param` and `profile`. If those values are omitted, creates a blank solver structure that can be filled by hand after.

```
DL = SMALL_init_DL(toolbox, name, param, profile)
```

Creates a dictionary learning structure with the given values of `toolbox`, `name`, `param` and `profile`. If those values are omitted, creates a blank dictionary learning structure that can be filled by hand after.

Call a solver

```
solver = SMALL_solve(problem, solver)
```

Applies a given solver to a given problem and returns the solver structure with updated fields `solution`, `reconstructed` and `time`.

Learn a dictionary

```
DL = SMALL_learn(problem, DL)
```

Applies a given dictionary learning algorithm to a given problem and returns the DL structure with updated fields `D` and `time`.

Add-on creation

If your add-on contains a new solver (respectively dictionary learning algorithm), then you need to add a case to the file `smallbox/config/SMALL_solve_config_local.m` (respectively `SMALL_learn_config_local.m`). If that file does not exist, create it as a copy of the file `SMALL_solve_config.m` (respectively `SMALL_learn_config_local.m`).