DAMOCO: MATLAB toolbox for multivariate data analysis, based on coupled oscillators approach
Example 2

January 16, 2011

1 Data and method

This example is implemented by the function `co_example2`; it uses the data file `co_vdp2.mat`. The data for this example are generated by a model of two bidirectionally coupled van der Pol oscillators:

$$\ddot{x}_1 - \mu(1 - x_1^2)\dot{x}_1 + \omega_1^2 x = \varepsilon_1(\dot{x}_2 - \dot{x}_1) \,,$$
$$\ddot{x}_2 - \mu(1 - x_2^2)\dot{x}_2 + \omega_2^2 x = \varepsilon_2(\dot{x}_1 - \dot{x}_2) \,. \tag{1}$$

Parameters are: $\mu = 0.5$, $\omega_1 = 1.11$, $\omega_2 = 0.89$, and $\varepsilon_1 = \varepsilon_2 = 0.1$.

In this example, we illustrate every step of the procedure and plot all intermediate results. The Fourier-based technique is used, see manual and related publications for details. You need the matlab optimization toolbox to run this example.

2 Output and comments

```
--------  Starting co_example2  -----------
--  Plotting a piece of bivariate data  --
Sampling frequency =7.9577
--  Computung and plotting protophases via Hilbert transform  --
Minimal amplitude over average amplitude is 0.8727
--  Performing 1D theta --> transformation  --
--  Reconstructing phase dynamics from 1D transformed protophases  --
--  Performing 2D transformation  --

Equation solved.

fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.



Constant terms of the coupling functions
   (estimates of natural frequencies) are:
```
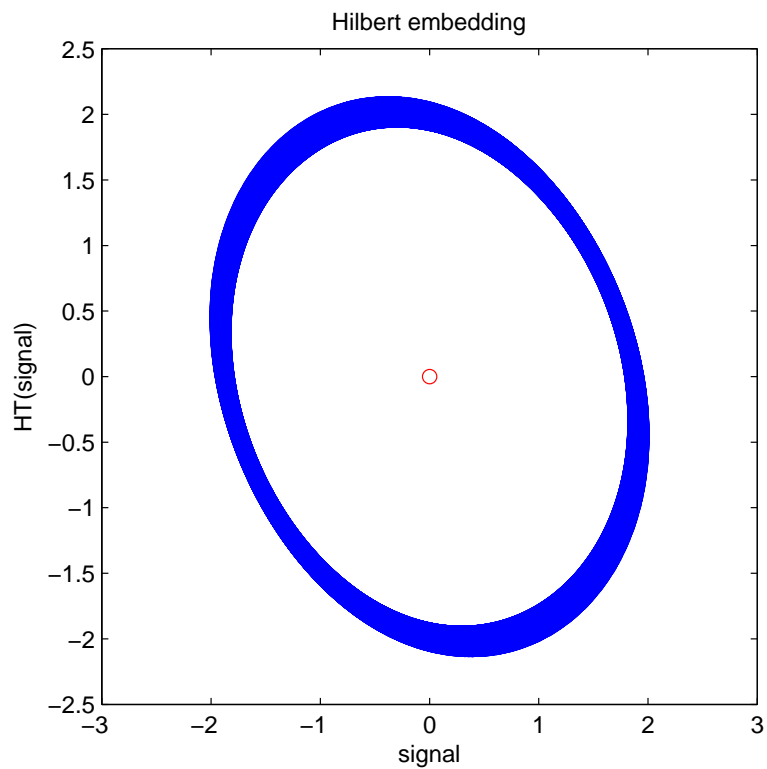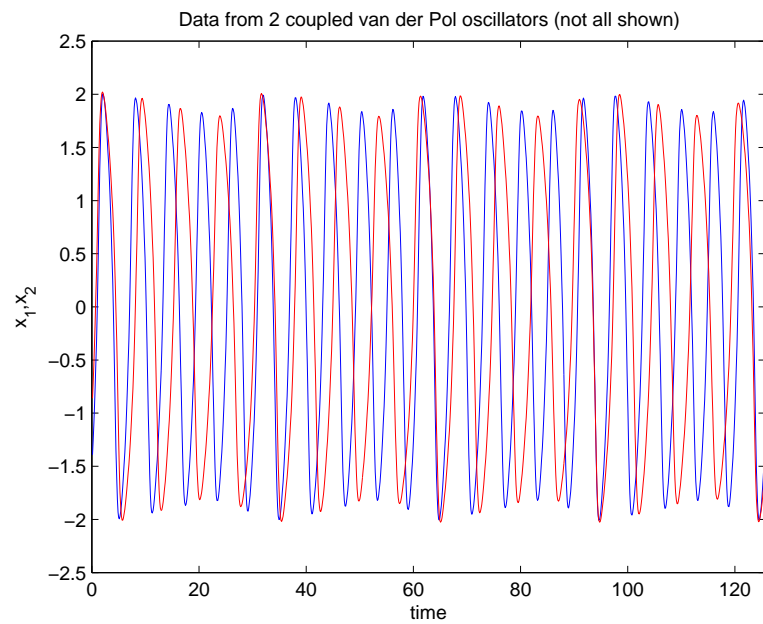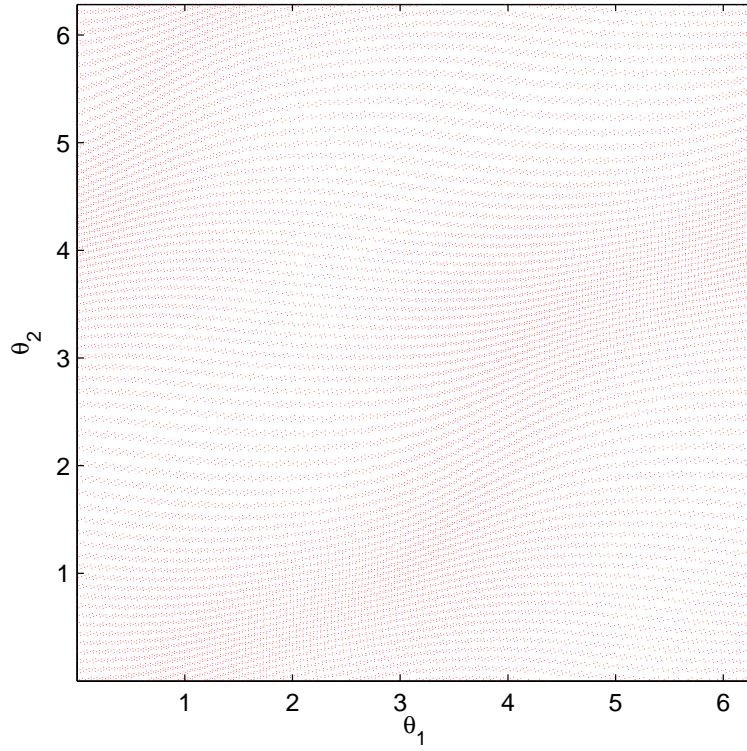
```
omega1 = 1.0634, omega2 = 0.83669
--  Quantification of interaction  --
Synchronization index from protophases 0.22319
Synchronization index from phases 0.22576
Directionality index (new version) = -0.07689
Directionality index (old version) = 0.035094
--------  End of co_example2  --------
```
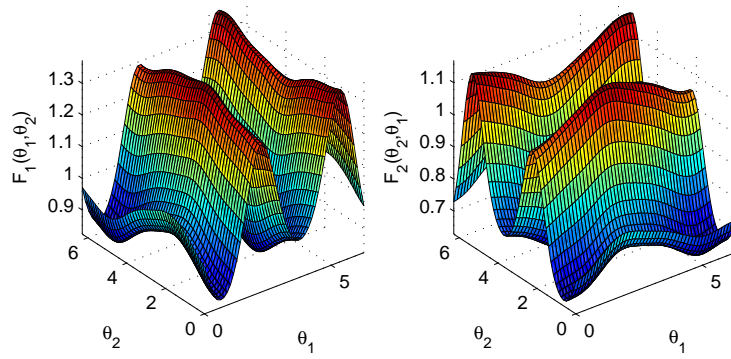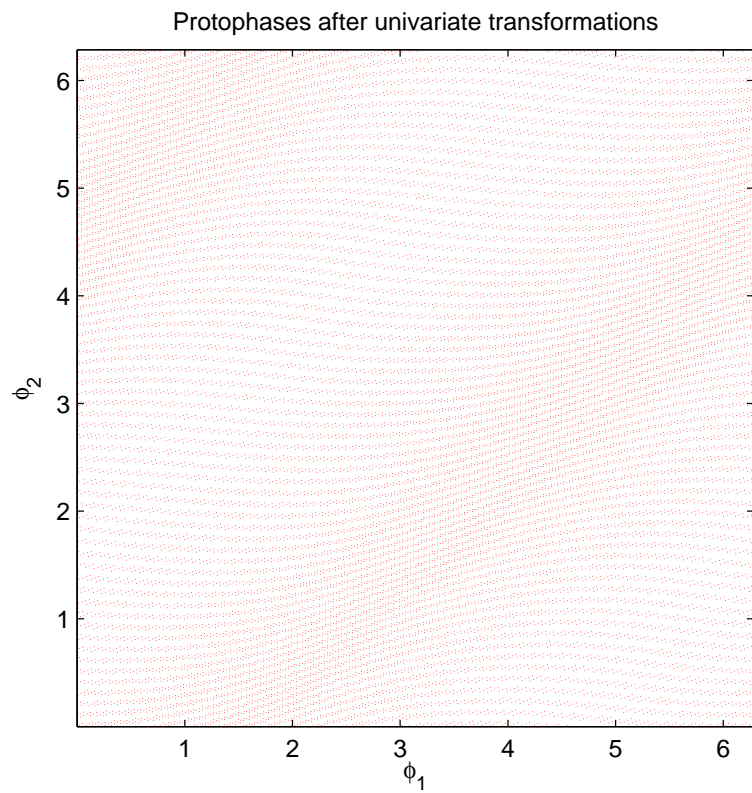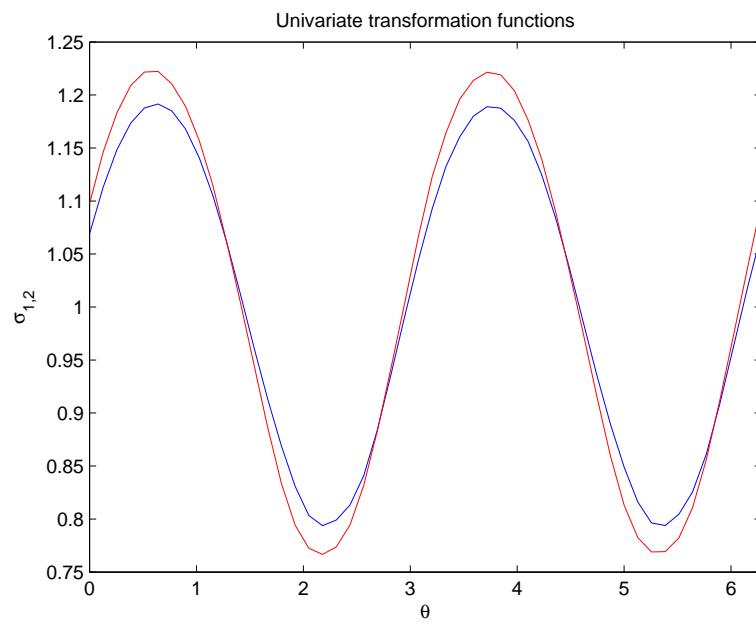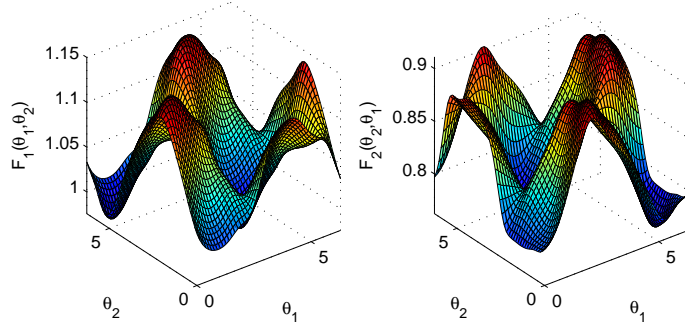
Data from 2 coupled van der Pol oscillators (not all shown)

Hilbert embedding

Raw protophases

Coupling function from raw protophases

3

Univariate transformation functions

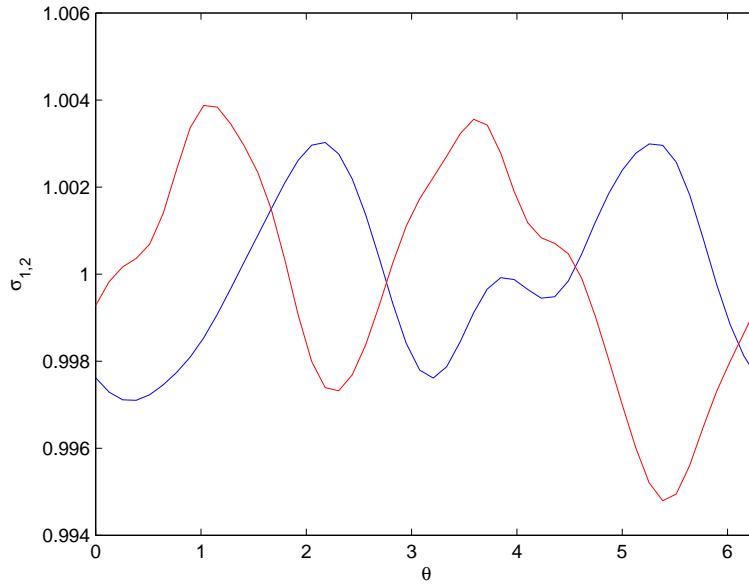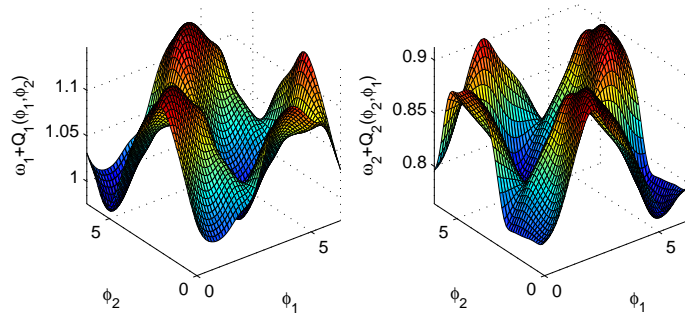Protophases after univariate transformations

Coupling function from 1D−transformed protophases

Bivariate transformation functions

True coupling functions after bivariate transformation

Several remarks are in order. One can see, that the univariate transformation practically does the job: the spurious dependence of the coupling function on the own protophases is removed after the univariate transformation. The difference between the functions after the univariate transformation and after the bivariate transformation is small; therefore, for many cases one can skip the bivariate transformation. Next, synchronization index computed from raw and transformed protophases is almost the same. This is, however, not always like that. If the data are not so smooth, the difference can be essential (see the example with the data from Hindmarsh-Rose neurons).