



CARNESS analysis package

Release 20150526.002

Alessandro Filisetti

May 26, 2015

CONTENTS

1	System Requirements	3
2	Installing	5
3	Usage	7
3.1	initializator Module	7
3.2	Chemistry Graph Analysis	7
3.3	Protocell Duplication Analysis	8
4	Indices	11
	Python Module Index	13
	Index	15

The CARNESS analysis python package contains a set of python scripts devoted to

1. the creation and the structural analysis of artificial catalytic reaction networks (CRS).
2. the analysis of the dynamical outcomes of the artificial catalytic reaction networks simulated by means of the CAlytic Reaction NEtworks Stochastic Simulator ([CARNESS](#)).

SYSTEM REQUIREMENTS

In order to use the scripts contained in this package, some further packages need to be installed. On UNIX systems you can install the necessary packages by the usual commands, for example on DEBIAN-based systems:

```
$ sudo apt-get update
$ sudo apt-get install python-numpy python-scipy python-matplotlib python-setuptools
python-networkx python-deap graphviz libgraphviz-dev pkg-config python-pygraphviz
```

On WINDOWS system, I personally suggest that the simplest solution may be to rely on the alternative solutions proposed [here](#).

INSTALLING

To get the packages (the package is entirely contained in the folder:

```
<path>/carness/_analysis/pythonTools
```

cloning the carness software, the carness analysis script will be automatically download:

```
git clone https://github.com/paxelito/carness.git
```

Since the package is still under development it is a good practice to check for possible updates or upgrades each time you use it. To search for new updates please run the following command within the “ACS_analysis” folder

```
git pull
```


To use the package you must run the opportune python script with the appropriate parameters. You can find a detailed description of the available scripts within this documentation. It is important to notice that not all the actually available scripts are depicted in this documentation. Hence, if you are familiar with python and with the CARNESSE simulator, you can search into the package whether the desired analysis have been already implemented. Our objective is that of preparing the documentation for all the packages in the next few months.

Current Available Scripts Analysis:

3.1 initializer Module

This script concerns all the aspect of the creation and the initialization of artificial catalytic reaction networks in the format requested by the [CARNESSE simulation platform](#). To have a description of all the parameters admitted by the initializer please digit:

```
python <path>/initializer.py -h
```

WARNING: In case of finite membrane passage (systemArchitecture=3 in acsm2s.conf) all the incoming species present in the _acsinflux.csv file will have the same values for the external concentrations and the transmembrane kinetic constants. To simulate different values all the _acsinflux.csv files must be modified manually. If several protocells or several seeds have been planned, then all the _acsinflux.csv files must be modified.

3.2 Chemistry Graph Analysis

This script evaluates a selected chemistry finding RAF, SCC and saving the bipartite multigraph and the catalyst-product network. RAF and SCC summaries are saved too.

Script SYNOPSIS:

usage: graph_chemistry_analysis.py [-h] [-p STRPATH] [-f LASTFLUXID] [-m MAXDIM]

Graph analysis of the chemistry

optional arguments:

- h, --help** show this help message and exit
- p STRPATH, --strPath STRPATH** Path of the artificial chemistry to analyze (def: ./)
- f LASTFLUXID, --lastFluxID LASTFLUXID** Last ID of the food species (def: 5)
- m MAXDIM, --maxDim MAXDIM** Max Dimension of the system (def: 6)

3.2.1 Bipartite MultiGraph legend

- **Nodes**
 - Red Circle: Molecular species
 - Blue Circle: Molecular species belonging to the SCCs (only in ALL reactions representations)
 - Green Square: reactions
- **Edges**
 - Grey: Substrate or product, according to the direction of the arrow, participation
 - Blue: Catalysis
 - Red: WARNING arrow. It means that a species is both a catalyst and a substrate of the reaction.

3.2.2 OUTPUT files

Output are stored in a folder named `_0_new_allStatResults` located within the chemistry directory. Files list:

- `completebipartitegraph.png.[png/net]` :: Bipartite Multigraph of all the chemistry
- `bipartiteRAFgraph.png.[png/net]` :: Bipartite Multigraph of the reactions involved in the RAF only
- `chemistry_cat_prod_graph.png.[png/net]` :: Catalyst -> Product representation of the chemistry
- `0_initRafAnalysis.csv` :: Summary of the RAF properties
- `0_initRafAnalysisALL` :: List of the reactions composing the RAF
- `0_initRafAnalysisLIST` :: Enumeration of the species and reactions composing the RAF

Currently graphs are exported in PAJEK (<http://mrvar.fdv.uni-lj.si/pajek/>) format, other formats are available at <http://networkx.lanl.gov/reference/readwrite.html>

3.3 Protocell Duplication Analysis

Script to compute the successive cell division times and the value of each molecule in proximity of the cell division. Please digit:

```
python <path>/acsDuplicationAnalysis.py -h
```

for the SYNOPSIS of the script.

An interesting example concerns the creation of a plot with just some molecular species:

```
python <path>/acsDuplicationAnalysis.py -p ./ -d50 -g -a -m30 -c 14 29
```

`-c 14 29` indicates that only the species 14 and 29 will be plotted

In particular, a folder named `_0_new_allStatResults` will be created into the chemistry and three files will be stored there:

1. `deltat_<CHEMISTRY>.csv`: In this file the cell division time and the overall amount of molecules at each division are stored
2. `delta_ALL_<CHEMISTRY>.csv`: In this file the overall amount of each molecular species at each division is stored

3. allTimes_<CHEMISTRY>.csv: If *param -alldynamics -a*, hence in this file the amount of the molecules in function of time is stored
4. divplot_<CHEMISTRY>.[png/eps]: If *param -graphs -g*, hence the plot of the species amount at each generation is generated.
5. all_time_plot_<CHEMISTRY>.[png.eps]: If *param -graphs -g* and *param -alldynamics -a*, hence the plot of the species amount at each time is generated.

INDICES

- *genindex*
- *modindex*
- *search*

a

`acsDuplicationAnalysis`, 8

g

`graph_chemistry_analysis`, 7

i

`initializer`, 7

A

`acsDuplicationAnalysis` (module), 8

G

`graph_chemistry_analysis` (module), 7

I

`initializer` (module), 7