Supporting Materials - Supporting Chemical reservoir computation in a self-organizing reaction network

Repository structure

- `/data` contains the processed data to perform the various reservoir computation tasks
- `/analysis` contains notebooks used to perform the various reservoir computation tasks and for creation of all data-related figures in the publication and supporting information.
- `/plots` and `/plots_extended` contain all data-related figures for the main manuscript and supporting information respectively (these are generated from the notebooks in the `analysis` directory)

Installation

The code in this repository requires Python 3.11 together with a few standard packages (numpy, pandas, scikit-learn, etc…). All packages can be installed by creating a Conda environment from `environment.yaml`.

Important: Installing AMICI requirements

For calculation of the *in silico* carbon metabolism model, we use AMICI. To correctly install AMICI, extra steps are required: Installation of the AMICI Python package has the following prerequisites:

- Python>=3.9
- SWIG>=3.0
- CBLAS compatible BLAS library (e.g., OpenBLAS, CBLAS, Atlas, Accelerate, Intel MKL)
- a C++17 compatible C++ compiler and a C compiler (e.g., g++, clang, Intel C++ compiler, mingw)

Ubuntu 22.04

On Ubuntu, the requirements can be installed via `apt`:

```
sudo apt install libatlas-base-dev swig

# optionally for HDF5 support (recommended):
sudo apt install libhdf5-serial-dev
```

OSX

On OSX, the requirements can be installed via `homebrew`:

```
brew install swig

# optionally for HDF5 support:
brew install hdf5

# optionally for parallel simulations:
brew install libomp
```

Windows

Installation on Windows directly is not recommended. We advice using the Windows Subsystem for Linux (WSL) and follow the instructions for installation on linux.

Create a Conda environment

After installing the AMICI requirements, the environment with all Python packages can be installed by running:

```
conda env create -f environment.yaml
```

after which the environment can be activated by running:

```
conda activate formose_rc
```

Figure fonts

Figures created in the Jupyter notebooks use Arial as a font. This font is not installed on Ubuntu by default, but can be installed by running:

```
sudo apt install ttf-mscorefonts-installer
sudo fc-cache -f
```

and subsequently deleting the matplotlib font cache:

```
rm ~/.cache/matplotlib -rf
```