

✓ BPL_CHO_Perfusion_cspr_openloop script with FMPy

The key library FMPy is installed.

After the installation a small application BPL_CHO_Fedbatch is loaded and run. You can continue with this example if you like.

```
!lsb_release -a # Actual VM Ubuntu version used by Google
```

```
⇒ No LSB modules are available.
   Distributor ID: Ubuntu
   Description:    Ubuntu 22.04.3 LTS
   Release:        22.04
   Codename:       jammy
```

```
%env PYTHONPATH=
```

```
⇒ env: PYTHONPATH=
```

```
!wget https://repo.anaconda.com/miniconda/Miniconda3-py312_24.3.0-0-Linux-x86_64.
!chmod +x Miniconda3-py312_24.3.0-0-Linux-x86_64.sh
!bash ./Miniconda3-py312_24.3.0-0-Linux-x86_64.sh -b -f -p /usr/local
import sys
sys.path.append('/usr/local/lib/python3.12/site-packages/')
```

```
⇒ --2024-05-23 09:52:27-- https://repo.anaconda.com/miniconda/Miniconda3-py312
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.191.158, 104.16.32.1
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.191.158|:443... co
HTTP request sent, awaiting response... 200 OK
Length: 143351488 (137M) [application/octet-stream]
Saving to: 'Miniconda3-py312_24.3.0-0-Linux-x86_64.sh'
```

```
Miniconda3-py312_24 100%[=====>] 136.71M 31.7MB/s in 4.5s
```

```
2024-05-23 09:52:32 (30.1 MB/s) - 'Miniconda3-py312_24.3.0-0-Linux-x86_64.sh'
```

```
PREFIX=/usr/local
Unpacking payload ...
```

```
Installing base environment...
```

```
Preparing transaction: ...working... done
Executing transaction: ...working... done
installation finished.
```

```
!conda update -n base -c defaults conda --yes
```

```
⇒ Channels:
   - defaults
   Platform: linux-64
   Collecting package metadata (repodata.json): done
```

Solving environment: done

Package Plan

environment location: /usr/local

added / updated specs:

- conda

The following packages will be downloaded:

package	build	
conda-24.5.0	py312h06a4308_0	1.2 MB
frozendict-2.4.2	py312h06a4308_0	36 KB
openssl-3.0.13	h7f8727e_2	5.2 MB
Total:		6.5 MB

The following NEW packages will be INSTALLED:

frozendict pkgs/main/linux-64::frozendict-2.4.2-py312h06a4308_0

The following packages will be UPDATED:

conda 24.3.0-py312h06a4308_0 --> 24.5.0-py312h06a4308_0

openssl 3.0.13-h7f8727e_0 --> 3.0.13-h7f8727e_2

Downloading and Extracting Packages:

openssl-3.0.13	5.2 MB	:	0% 0/1 [00:00<?, ?it/s]
conda-24.5.0	1.2 MB	:	0% 0/1 [00:00<?, ?it/s]
openssl-3.0.13	5.2 MB	:	0% 0.002992912657148922/1 [00:00<00:48]
frozendict-2.4.2	36 KB	:	44% 0.43853215920344746/1 [00:00<00:00,
conda-24.5.0	1.2 MB	:	1% 0.01293349794914382/1 [00:00<00:11,
frozendict-2.4.2	36 KB	:	100% 1.0/1 [00:00<00:00, 2.89it/s]
conda-24.5.0	1.2 MB	:	100% 1.0/1 [00:00<00:00, 1.87it/s]

Preparing transaction: done

Verifying transaction: done

Executing transaction: done

!conda --version

!python --version

🔄 conda 24.5.0

Python 3.12.2

!conda install -c conda-forge fmpy --yes # Install the key package



Preparing transaction: done
Verifying transaction: done
Executing transaction: done

```
#!conda install matplotlib --yes
```

```
#!conda install scipy --yes
```

```
#!conda install openpyxl --yes
```

```
#!conda install xlrd --yes
```

✓ BPL_CHO_Perfusion_cspr_opennloop setup

Now specific installation and the run simulations. Start with connecting to Github. Then upload the two files:

- FMU - BPL_CHO_Perfusion_cspr_opennloop_linux_om_me.fmu
- Setup-file - BPL_CHO_Perfusion_cspr_opennloop_fmpy_explore

```
# Filter out DeprecationWarnings for 'np.float as alias' is needed – wish I could
import warnings
warnings.filterwarnings("ignore")
```

```
%%bash
git clone https://github.com/janpeter19/BPL_CHO_Perfusion
```

```
🔄 Cloning into 'BPL_CHO_Perfusion'...
```

```
%cd BPL_CHO_Perfusion
```

```
🔄 /content/BPL_CHO_Perfusion
```

✓ BPL_CHO_Perfusion_cspr_opennloop - demo

Author: Jan Peter Axelson

```
run -i BPL_CHO_perfusion_cspr_opennloop_explore_fmpy.py
```

```
🔄 Linux – run FMU pre-compiled OpenModelica 1.23.0-dev
```

Model for bioreactor has been setup. Key commands:

- par() - change of parameters and initial values
- init() - change initial values only
- simu() - simulate and plot
- newplot() - make a new plot
- show() - show plot from previous simulation
- disp() - display parameters and initial values from the last simulation
- describe() - describe culture, broth, parameters, variables with values/units

Note that both `disp()` and `describe()` takes values from the last simulation and the command `process_diagram()` brings up the main configuration

Brief information about a command by `help()`, eg `help(simu)`
Key system information is listed with the command `system_info()`

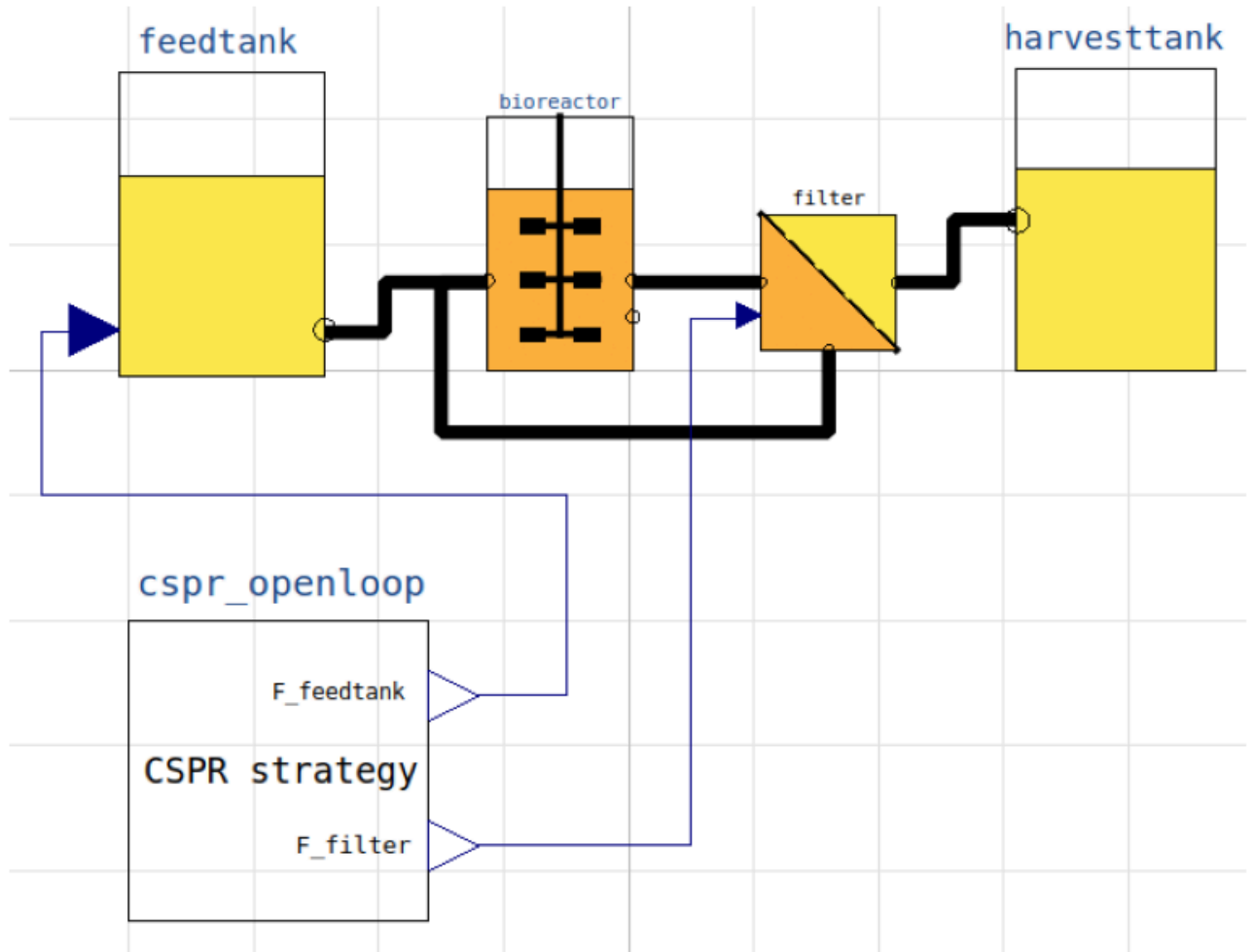
```
%matplotlib inline  
plt.rcParams['figure.figsize'] = [25/2.54, 20/2.54]
```

✓ 1 About the process model

We can get information about the process and liquid phase by the command `describe()`. Here is no gas-phase included. This command can also be used to bring up information about a specific variable or parameter. However, you should use `describe()` after a simulation to get the values used during the simulation.

```
process_diagram()
```

➡ No processDiagram.png file in the FMU, but try the file on disk.



```
describe('culture'); print(); #describe('liquidphase')
```

➡ Reactor culture CHO-MAb – cell line HB-58 American Culture Collection ATCC

The molecular weight of the recombinant protein (MAb) is somewhat arbitrarily chosen and the value not used in the simulations.

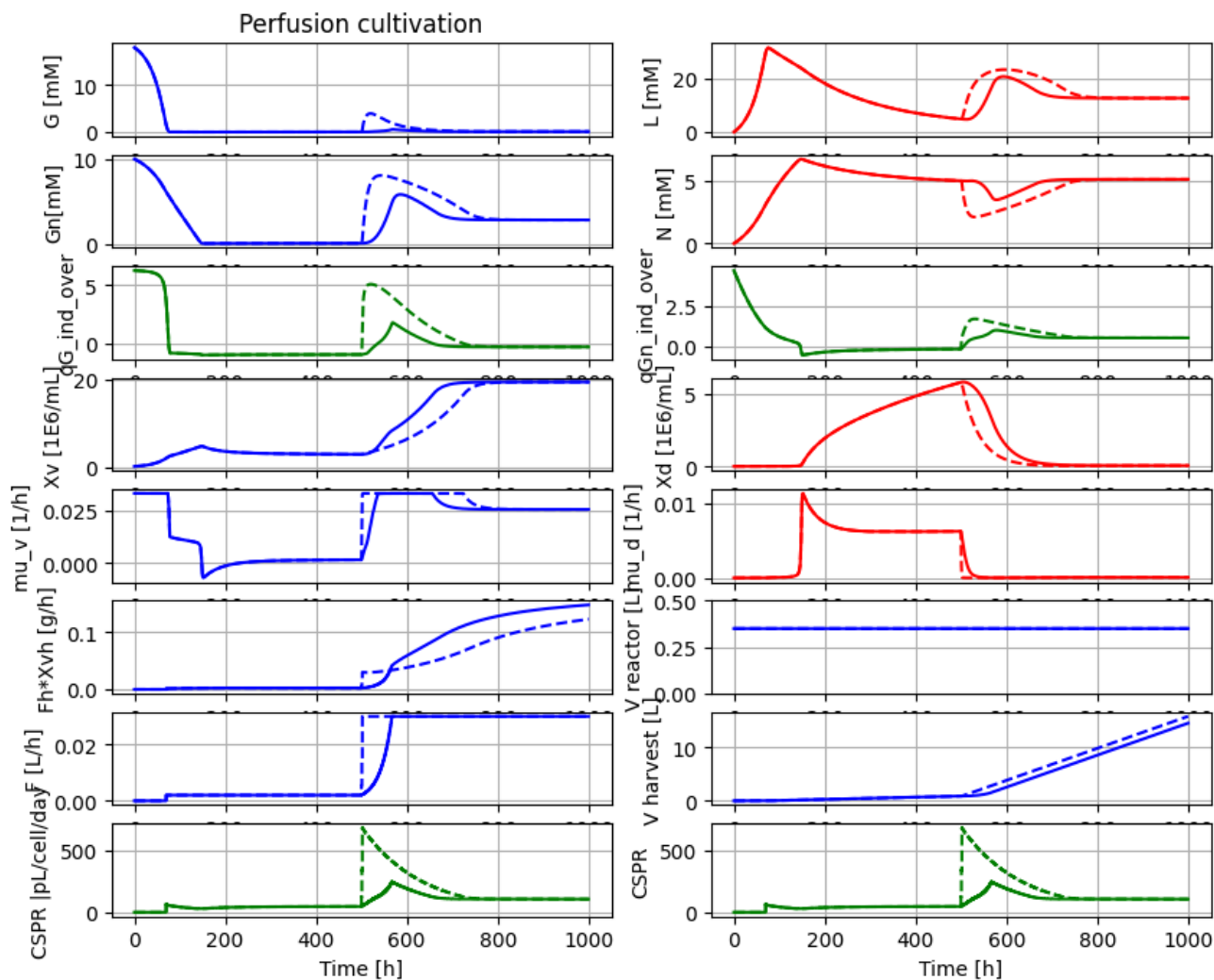
```
describe('MSL')
```

➡ MSL: 3.2.3 – used components: RealInput, RealOutput, CombiTimeTable, Types

```
newplot('Perfusion cultivation', plotType='Extended')
```

```
par(samplePeriod=1); par(F1=0.0020)
par(mu_ref=0.04); simu()
par(mu_ref=1); simu()
```

```
# General parameters
# First simulation (solid)
# Second simulation (dashed)
```



```
model_get('CSPR')
```



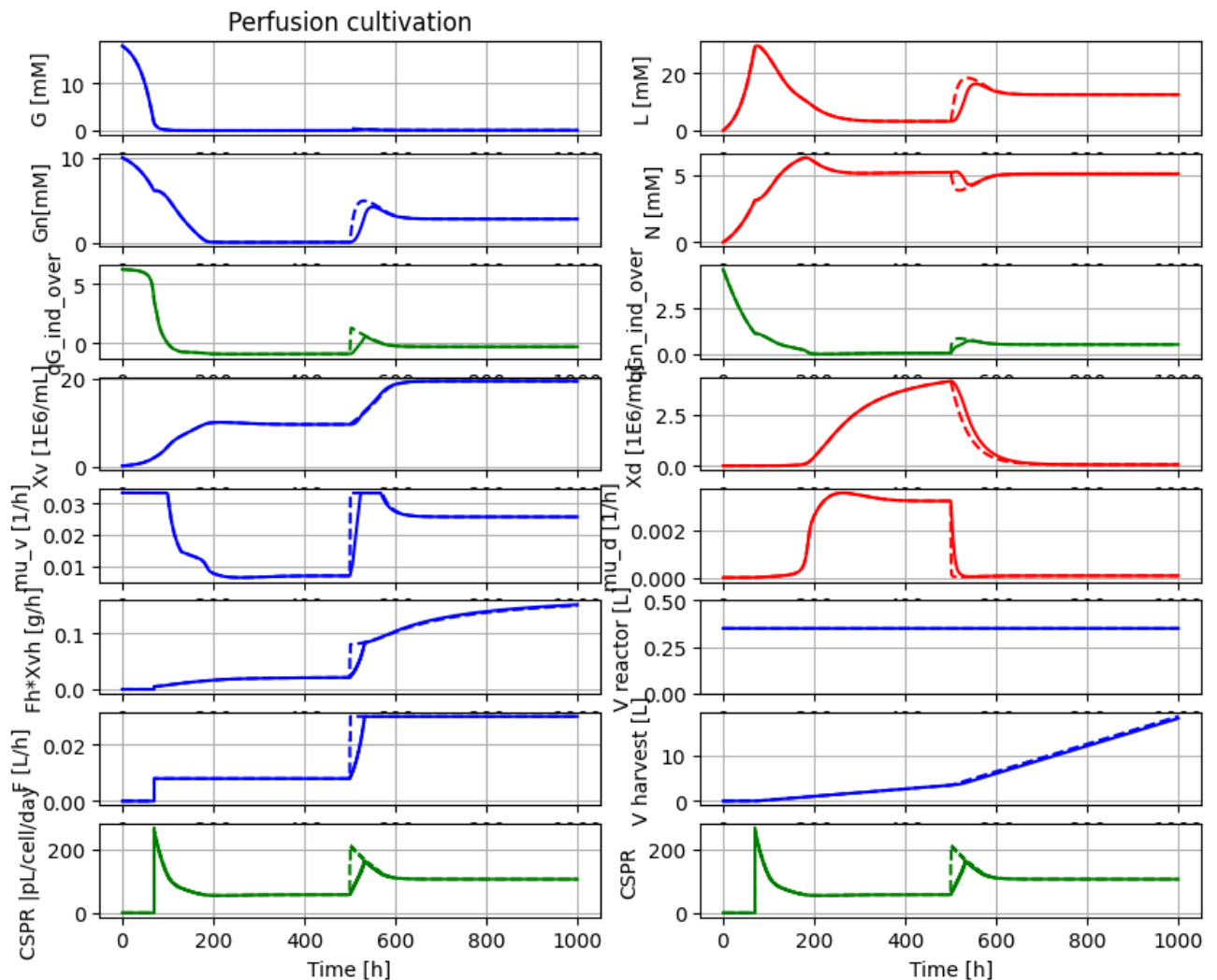
```
105.41563602162672
```

We see here the shorter settling time for cell conc X_v when the slower exponential increase of perfusion rate is used, compared to an abrupt change.

```
newplot('Perfusion cultivation', plotType='Extended')
```

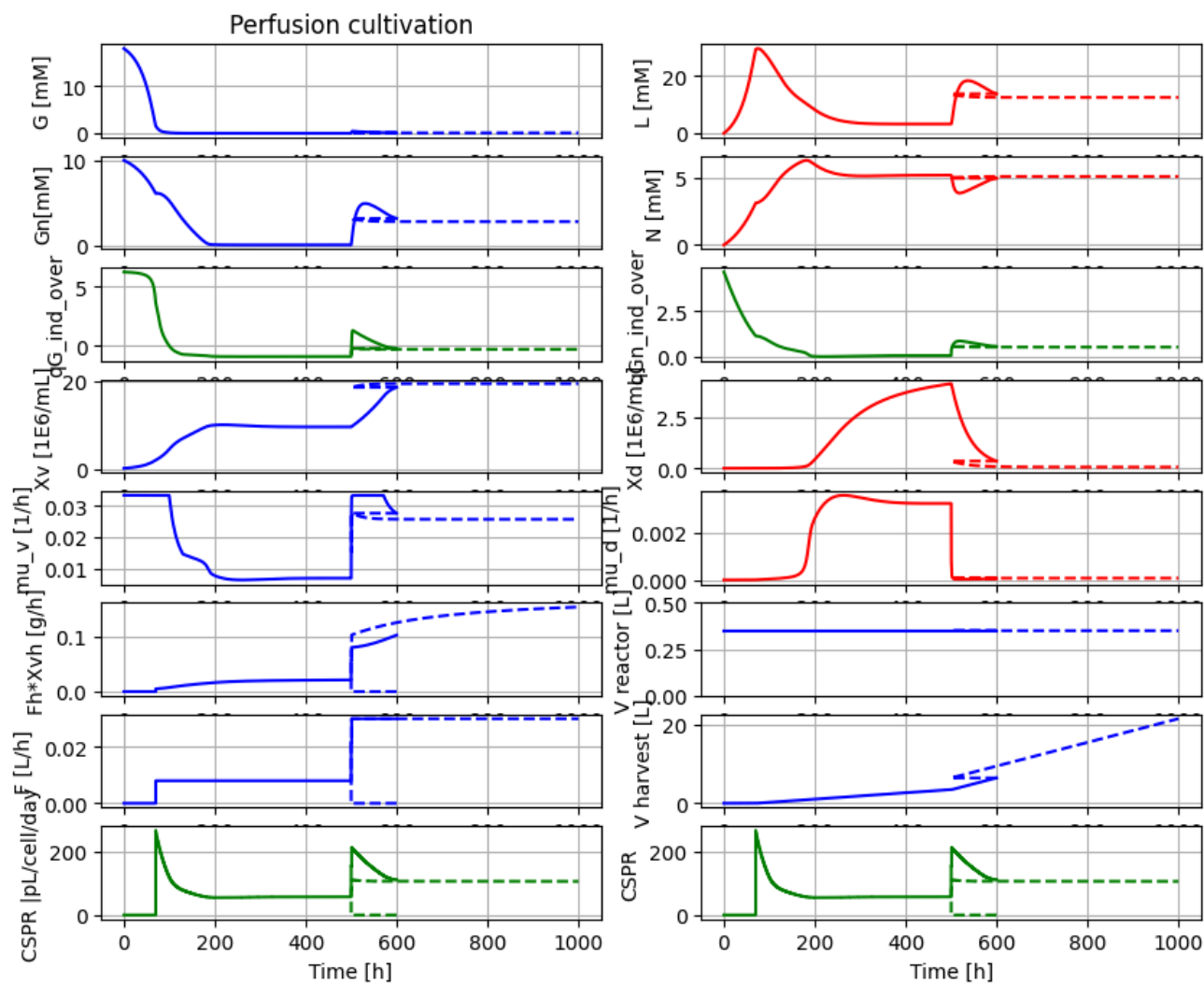
```
par(samplePeriod=1); par(F1=0.0080)
par(mu_ref=0.04); simu()
par(mu_ref=1); simu()
```

```
# General parameters
# First simulation (solid)
# Second simulation (dashed)
```



We see that a somewhat smaller change in perfusion rate make the difference in settling time of X_v for the two strategies to almost disappear.

```
newplot('Perfusion cultivation', plotType='Extended')
simu(600)
simu(400, 'cont')
```

```
describe('feedtank.W')
```

⇒ Variable not logged

```
describe('feedtank.V')
```

⇒ Feed volume : 78.577 [L]

```
-----> 1 describe('feedtank.V')
```

```
describe('bioreactor.broth_decay.k_decay')
```

⇒ Rate of decay of viable cells : 0.0 [1E9 cells/(L*h)]

```
43         except AttributeError:
```

```
describe('k_lysis')
```

⇒ Specific rate of lysis of dead cells : 0.0 [1/h]

```
disp('culture')
```

⇒ qG_max1 : 0.297
qG_max2 : 0.038
qGn_max1 : 0.124
qGn_max2 : 0.022
mu_d_max : 0.13

```
describe('mu')
```

⇒ Specific cell growth rate variable : 0.026 [1/h]

```
describe('parts')
```

⇒ ['bioreactor', 'bioreactor.broth_decay', 'bioreactor.culture', 'CSPR', 'cspr_