

## ✓ BPL\_TEST2\_Batch script with FMPy

The key library FMPy is installed.

After the installation a small application BPL\_YEAST\_COB\_Batch 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.sh
```

```
!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/')
```

```
➤ --2024-08-29 10:15:05-- https://repo.anaconda.com/miniconda/Miniconda3-py312_24.3.0-0-Linux-x86_64.sh
  Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.191.158, 104.16.32.241, 2606:4700::6810:20f1, ...
  Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.191.158|:443... connected.
  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 121MB/s in 1.1s
```

```
2024-08-29 10:15:06 (121 MB/s) - 'Miniconda3-py312_24.3.0-0-Linux-x86_64.sh' saved [143351488/143351488]
```

```
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
```

```
➤
```

frozendict-2.4.2	36 KB	: 100% 1.0/1 [00:00<00:00, 1.95it/s]
certifi-2024.7.4	159 KB	: 100% 1.0/1 [00:00<00:00, 2.07it/s]
openssl-3.0.14	5.2 MB	: 100% 1.0/1 [00:01<00:00, 1.10s/it]
conda-24.7.1	1.2 MB	: 100% 1.0/1 [00:01<00:00, 1.09s/it]

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

```
!conda --version
!python --version
```

```
↔ conda 24.7.1
   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 xlrd --yes
```

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

```
# For some reason conda installation does not work for optlang
!pip install optlang
```

```
Collecting optlang
  Downloading optlang-1.8.2-py2.py3-none-any.whl.metadata (8.1 kB)
Collecting swiglpk>=5.0.8 (from optlang)
  Downloading swiglpk-5.0.10-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (5.5 kB)
Collecting sympy>=1.12.0 (from optlang)
  Using cached sympy-1.13.2-py3-none-any.whl.metadata (12 kB)
Collecting mpmath<1.4,>=1.1.0 (from sympy>=1.12.0->optlang)
  Using cached mpmath-1.3.0-py3-none-any.whl.metadata (8.6 kB)
Downloading optlang-1.8.2-py2.py3-none-any.whl (141 kB)
141.8/141.8 kB 3.7 MB/s eta 0:00:00
Downloading swiglpk-5.0.10-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (2.3 MB)
2.3/2.3 MB 31.1 MB/s eta 0:00:00
Using cached sympy-1.13.2-py3-none-any.whl (6.2 MB)
Using cached mpmath-1.3.0-py3-none-any.whl (536 kB)
Installing collected packages: swiglpk, mpmath, sympy, optlang
Successfully installed mpmath-1.3.0 optlang-1.8.2 swiglpk-5.0.10 sympy-1.13.2
```

## ✓ BPL\_YEAST\_COB\_Batch setup

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

- FMU - BPL\_YEAST\_COB\_Batch\_linux\_om\_me.fmu
- Setup-file - BPL\_YEAST\_COB\_Batch\_fmpy\_explore.py

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

```
Cloning into 'BPL_YEAST_COB_Batch'...
```

```
%cd BPL_YEAST_COB_Batch
```

```
/content/BPL_YEAST_COB_Batch
```

```
run -i BPL_YEAST_COB_Batch_fmpy_explore.py
```

```
Linux - run FMU pre-comiled OpenModelica
```

```
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]
```

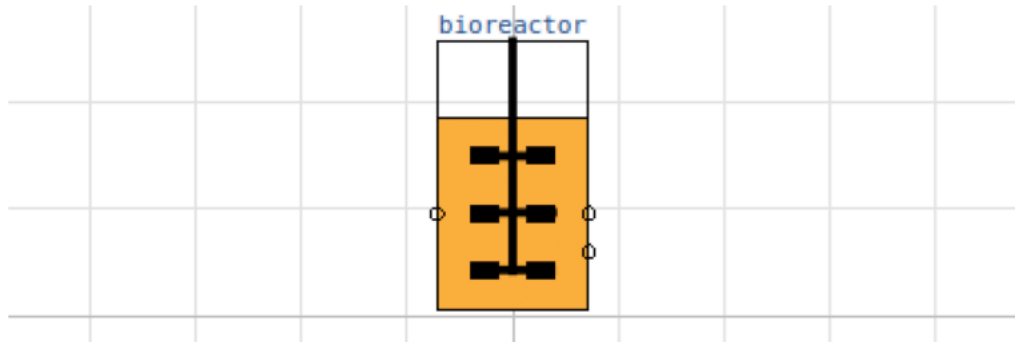
```
import warnings
warnings.filterwarnings("ignore")
```

## ✓ BPL\_YEAST\_COB\_Batch - demo

```
from optlang import Model, Variable, Constraint, Objective
```

```
process_diagram()
```

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



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

```
# Pump schedule parameter
```

➡ *Saccharomyces cerevisiae* – default parameters for strain H1022

```
# Define culture constraint-based model
def culture(G, E):
```

```
    # LP calculation of the optimal qGr, qEr based on G and E values
```

```
    # - parameters
```

```
    q02max = 6.9e-3; kog = 2.3; koe = 1.6; YGr = 3.5; YEr = 1.32;
    alpha = 0.01; beta = 1.0
```

```
    # - transfer data from dynamic reactor model to static LP model
```

```
    qGr_opt = Variable('qGr_opt', lb=0)
    qEr_opt = Variable('qEr_opt', lb=0)
```

```
    # - LP model constraint and objective
```

```
    mu_max = Objective(YGr*qGr_opt + YEr*qEr_opt, direction='max')
    q02lim = Constraint(kog*qGr_opt + koe*qEr_opt, ub=q02max)
    qGlim = Constraint(qGr_opt, ub=alpha*max(0,G))
    qElim = Constraint(qEr_opt, ub=beta*max(0,E))
```

```
    # - put together the LP model
```

```
    yeast_model = Model(name='Yeast bottleneck model')
    yeast_model.objective = mu_max
    yeast_model.add(q02lim)
    yeast_model.add(qGlim)
    yeast_model.add(qElim)
```

```
    # - do LP optimization
```

```
    yeast_model.optimize()
```

```
    return (yeast_model.objective.value, yeast_model.variables.qGr_opt.primal, yeast_model.variables.qEr_opt.primal, q02lim.
```

```
# Initialization
```

```
V_start=1.0
```

```
init(V_start=V_start, VX_start=V_start*2.0, VG_start=V_start*10, VE_start=V_start*3.0)
```

```
# Loop of simulations
```

```
t_final = 8.0
```

```
t_samp = 0.0333
```

```
n_samp = t_final/t_samp + 1
```

```
# Simulate n sample steps
```

```
newplot(title='Batch cultivation', plotType='TimeSeries2')
```

```
ax1.set_xlim([0, t_final]); ax2.set_xlim([0, t_final]); ax3.set_xlim([0, t_final])
```

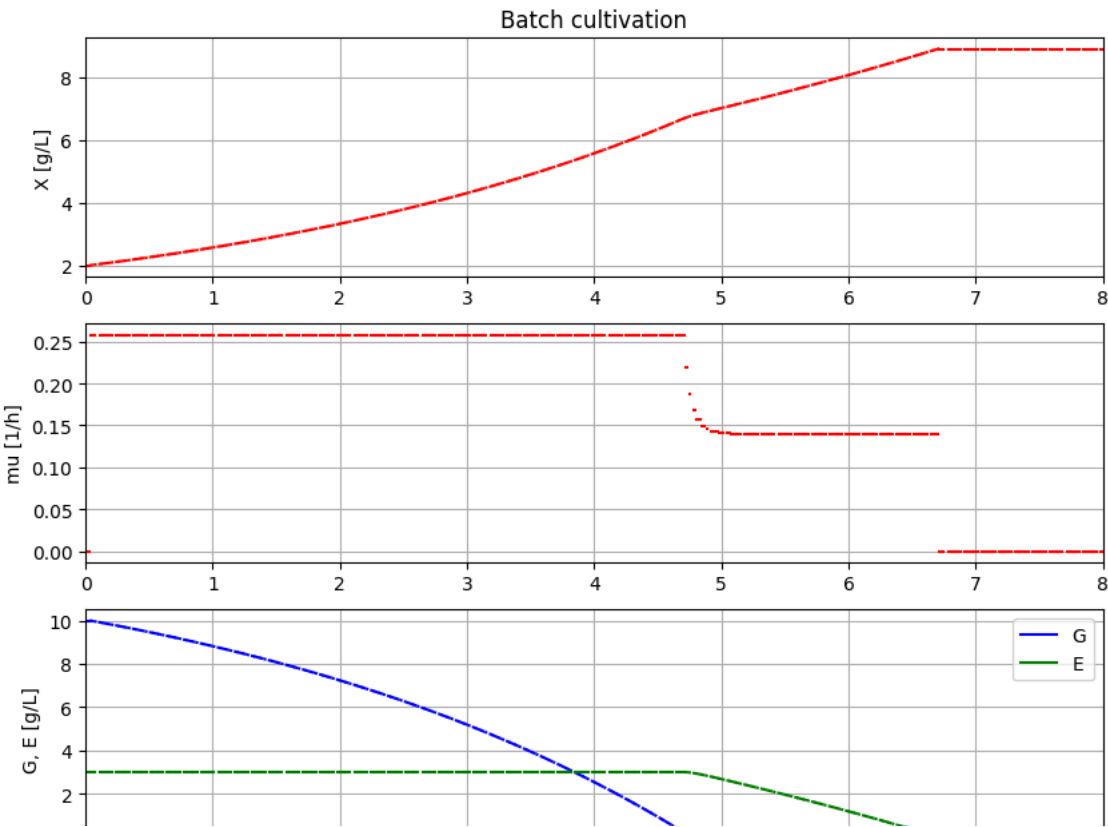
```
simu(t_samp, options=opts_fast)
```

```
for i in range(int(n_samp)):
```

```
    (mum_opt, qGr_opt, qEr_opt, q02_opt) = culture(sim_res['bioreactor.c[2]'][-1], sim_res['bioreactor.c[3]'][-1])
```

```
    par(mum=mum_opt, qGr=qGr_opt, qEr=qEr_opt, q02=q02_opt)
```

```
    simu(t_samp, 'cont', options=opts_fast)
```



system\_info()



System information  
-OS: Linux  
-Python: 3.10.12  
-Scipy: not installed in the notebook  
-FMPy: 0.3.20  
-FMU by: OpenModelica Compiler OpenModelica 1.23.1  
-FMI: 2.0  
-Type: ME  
-Name: BPL\_YEAST\_COB.Batch  
-Generated: 2024-08-29T11:44:30Z  
-MSL: 3.2.3  
-Description: Bioprocess Library version 2.2.1 - GUI  
-Interaction: FMU-explore for FMPy version 1.0.1

!conda list optlang



```
# packages in environment at /usr/local:
#
# Name          Version      Build Channel
optlang        1.8.2        pypi_0    pypi
```