

---

# findCPcli

Release 0.0.8-22-g9b70189

Alex Oarga

May 03, 2021

## Contents

<b>1</b>	<b>1. Introduction</b>	<b>1</b>
1.1	1.1 Chokepoint reactions . . . . .	2
1.2	1.2. Dead-End Metabolites (DEM) . . . . .	2
1.3	1.3. Essential reactions . . . . .	2
1.4	1.4. Example . . . . .	2
<b>2</b>	<b>2. Installation</b>	<b>3</b>
<b>3</b>	<b>3. Pseudocode</b>	<b>3</b>
<b>4</b>	<b>4. Compute chokepoints</b>	<b>4</b>
4.1	4.1. Spreadsheet data . . . . .	5
<b>5</b>	<b>5. Compute growth dependent chokepoints</b>	<b>5</b>
5.1	5.1. Procedure pseudocode . . . . .	6
<b>6</b>	<b>6. Remove Dead-End Metabolites</b>	<b>6</b>
<b>7</b>	<b>7. Refine model with FVA</b>	<b>6</b>
<b>8</b>	<b>8. Tool parameters</b>	<b>6</b>
<b>9</b>	<b>9. Low Level API</b>	<b>7</b>
<b>10</b>	<b>10. License</b>	<b>8</b>

---

## 1 1. Introduction

findCPcli is a command line python-tool for the computation of chokepoint reactions in genome-scale metabolic models. The main purpose of the tool is to compute chokepoints by taking into account both the topology and the dynamic information of the network. In addition to the computation of chokepoints, findCPcli can compute and remove dead-end metabolites, find essential reactions and update the flux bounds of the reactions according to the results of Flux Variability Analysis.

findCPcli takes as input an SBML files of genome-scale models, and provides as output a spreadsheet file with the results of the chokepoint computation.

## 1.1 1.1 Chokepoint reactions

Chokepoint reactions are those reactions that are either the unique consumer or the only producer of a given metabolite. findCPcli makes use of the flux bounds of the model to determine consumer and producer reactions, and in turn, to compute chokepoint reactions.

## 1.2 1.2. Dead-End Metabolites (DEM)

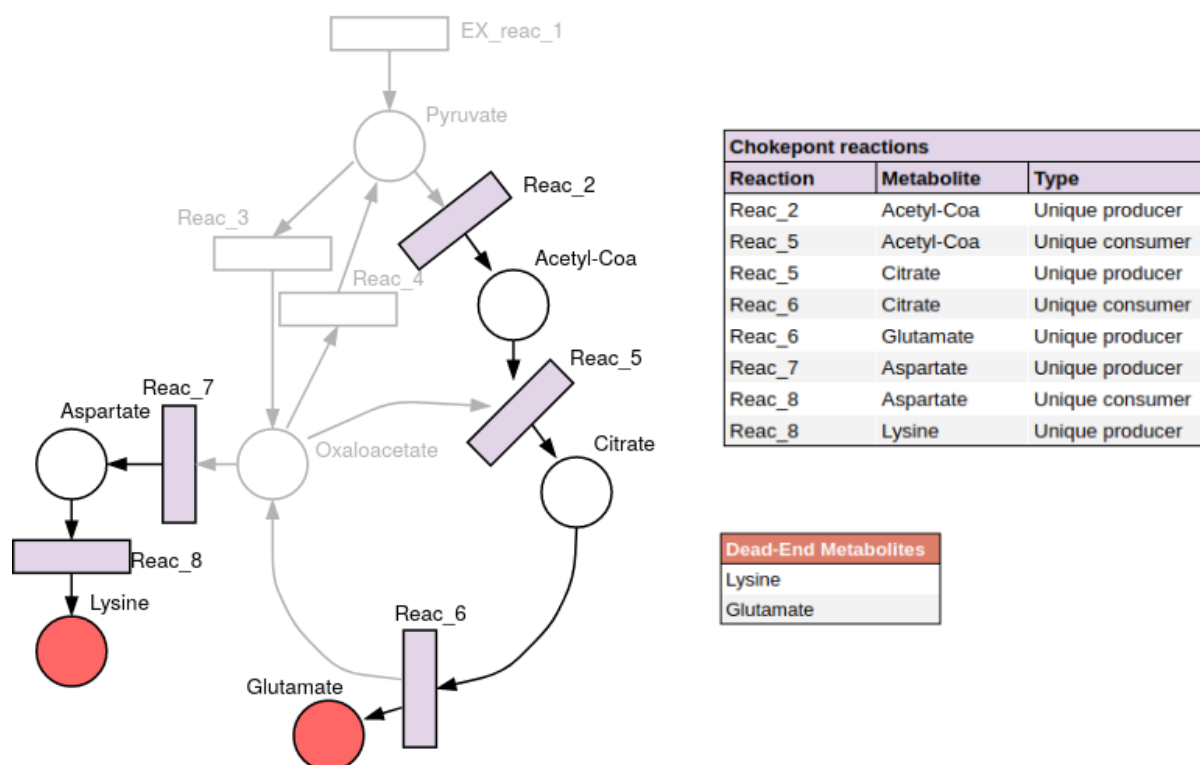
Dead-End Metabolites (DEM) are those metabolites that are not produced or consumed by any reaction.

## 1.3 1.3. Essential reactions

A reaction is considered an essential reaction if its deletion, this is, restricting its flux to zero, causes the objective (e.g. cellular growth) to be zero.

## 1.4 1.4. Example

Chokepoint reactions and dead-end metabolites example:



The computation of chokepoints can also be exploited programmatically via the [Low Level API](#) which is based on [COBRAPy](#)<sup>1</sup>.

For citation purposes please refer to:

**Note:** Oarga et al. **Growth Dependent Computation of Chokepoints in Metabolic Networks.** International Conference on Computational Methods in Systems Biology. Springer, Cham, 2020. [https://doi.org/10.1007/978-3-030-60327-4\\_6](https://doi.org/10.1007/978-3-030-60327-4_6).

<sup>1</sup> <https://github.com/opencobra/cobrapy>

## 2 2. Installation

findCPcli can be installed with **pip** package manager:

```
$ pip install findCPcli
```

## 3 3. Pseudocode

The following section includes pseudocode of some of the main operations performed by findCPcli.

- Find chokepoints on a model

```
function find_chokepoints(model)
  chokepoint_list = empty list
  for reaction in model
    if reaction upper flux bound not equal 0 and lower flux bound not equal 0
      for reactant in reaction
        if reaction is the only consumer of reactant
          chokepoint_list = chokepoint_list + (reaction, reactant)
      for product in reaction
        if reaction is the only producer of product
          chokepoint_list = chokepoint_list + (reaction, product)
  return chokepoint_list
```

- Find dead-end metabolites on a model

```
function find_dead_end_metabolites(model)
  dem_list = empty list
  for metabolite in model
    if length(metabolite.consumers) == 0 or length(metabolite.producers) == 0
      dem_list = dem_list + metabolite
  return dem_list
```

- Remove dead-end metabolites on a model

```
function remove_dead_end_metabolites(model)
  while number of metabolites in model does not change:
    find_dead_end_metabolites(model)
    delete all dead-end metabolites in model
    for reaction that produced or consumed dead-end metabolites:
      if reaction produces or consumes 0 metabolites [and is not exchange nor
↪demand]:
        delete reaction on model
    find_dead_end_metabolites(model)
  return model
```

- Update model flux bounds with Flux Variability Analysis

```
function update_flux_bounds_with_fva(model, fraction_of_optimum_growth)
  max_fva, min_fva = flux_variability_analysis(model, fraction_of_optimum_growth)
  for reaction in model
    reaction.upper_flux_bound = max_fva[reaction]
    reaction.lower_flux_bound = min_fva[reaction]
  return model
```

- Find essential reactions

```

function find_essential_reactions(model)
    essential_reactions = empty list
    for reaction in model
        knock out reaction
        if flux_balance_analysis on model is 0
            essential_reactions = essential_reactions + reaction
        undo knock out
    return essential_reactions

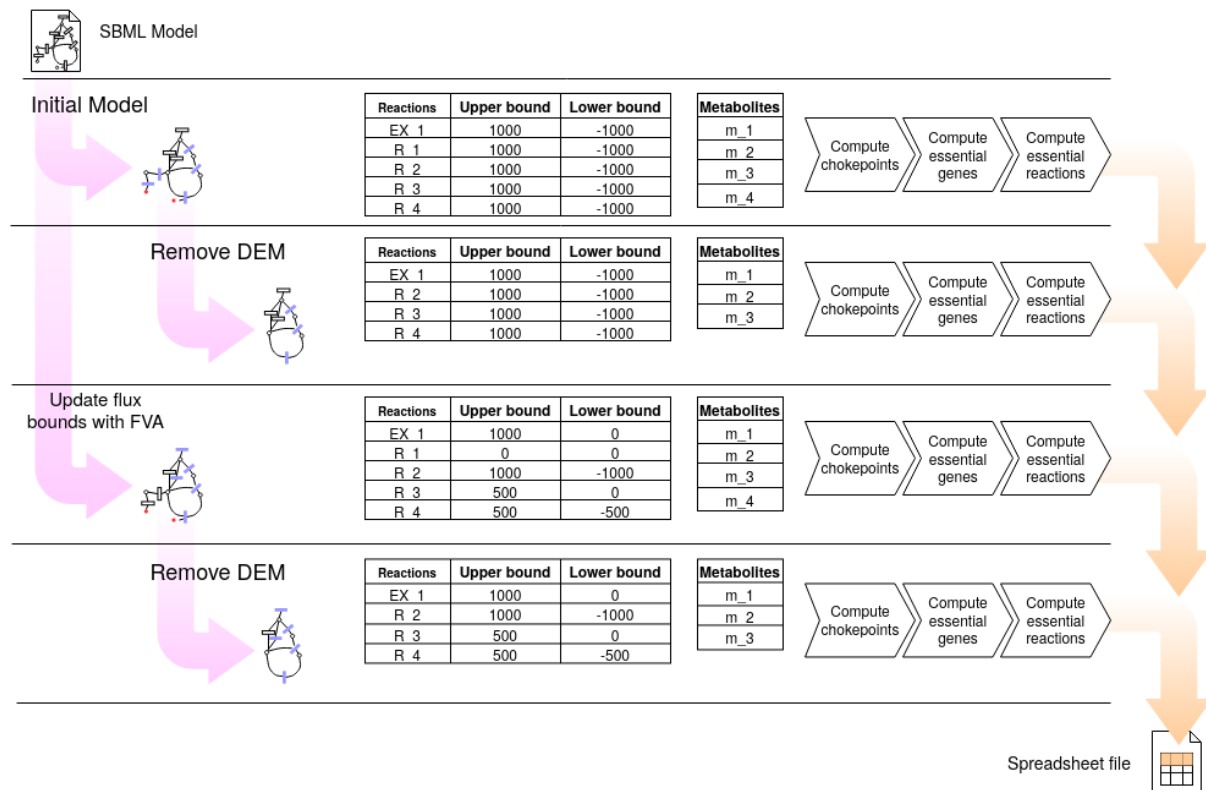
```

## 4 4. Compute chokepoints

Given a genome-scale model in SBML format, findCPcli computes chokepoints, dead-end metabolites, essential reactions, and essential genes, and saves the results in a spreadsheet. findCPcli can be run as follows, where `model.xml` is the file with the SBML model and `results.xls` is the file where the results will be saved.

```
$ findCPcli -i model.xml -o results.xls
```

The following figure shows the pipeline of the chokepoint computation process. For a given SBML file, computations are performed on 4 models: i) model in the SBML file; ii) model without DEM; iii) model refined with FVA, i.e. with flux bounds updated according to FVA; and iv) model refined with FVA and without DEM.



## 4.1 4.1. Spreadsheet data

The previous command produces a spreadsheet file containing the following sheets:

- `model_info`: General model information.
- `reactions`: List of reactions of the model
- `metabolites`: List of metabolites of the model
- `genes`: List of genes of the model
- `reactions_FVA`: Upper and lower flux bound of each reaction obtained with Flux Variability Analysis.
- `metabolites_FVA`: Upper and lower flux bound of each reaction obtained with Flux Variability Analysis grouped by metabolite.
- `reversible_reactions`: List of reversible reactions of the model before and after FVA refinement.
- `chokepoints`: Chokepoint reactions and the metabolite/s they produce/consume. Chokepoints are computed in 4 different models:
  1. Input model in the SBML file.
  2. Model without DEM.
  3. Model refined with FVA.
  4. Model refined with FVA and without DEM.
- `dead-end`: Dead-end metabolites before and after FVA refinement.
- `essential_genes`: List of essential genes of the model. Essential genes are computed in the 4 previously listed models.
- `essential_reactions`: List of essential reactions of the model. Essential reactions are computed in the 4 previously listed models.
- `comparison`: Comparison of chokepoint, essential reactions and essential gene reactions in the 4 previously listed models.
- `summary`: Comparison the size of the previous sets and their intersections.

## 5 5. Compute growth dependent chokepoints

`findCPcli` can be used to compute [Growth Dependent Chokepoints](#)<sup>2</sup> as follows: a) the flux of the objective function is set to a given fraction of its optimal value; b) FVA is run to compute lower and upper flux bounds of the reactions; c) the flux bounds are used to identify reversible, non reversible and dead reactions (i.e. reaction with lower and upper bound equal to 0); and d) this directionality of reactions is used to determine consumer and producer reactions, and in turn, chokepoints. The tool produces a spreadsheet showing how the size of the set of chokepoints varies with the fraction of the optimal value set to the objective function.

```
$ findCPcli -i model.xml -cp results.xls
```

<sup>2</sup> [https://doi.org/10.1007/978-3-030-60327-4\\_6](https://doi.org/10.1007/978-3-030-60327-4_6)

## 5.1 Procedure pseudocode

The pseudocode of this task is included below:

```
model = read_model()
initial_reversible_reactions    = all reactions with upper flux bound > 0 and lower flux_
↳bound < 0
initial_dead_reactions         = all reactions with both upper and lower flux bound_
↳equal to 0
initial_non_reversible_reactions = (model.reactions - reversible_reactions) - dead_
↳reactions
initial_chokepoint_reactions    = find_chokepoints(model)

for fraction in [0,0 ... 1,0]
    model = read_model()
    model = update_flux_bounds_with_fva(model, fraction)
    reversible_reactions[fraction]    = all reactions with upper flux bound > 0 and lower_
↳flux bound < 0
    dead_reactions[fraction]         = all reactions with both upper and lower flux_
↳bound equal to 0
    non_non_reversible_reactions[fraction] = (model.reactions - reversible_reactions) -_
↳dead_reactions
    chokepoint_reactions[fraction]    = find_chokepoints(model)
```

## 6 6. Remove Dead-End Metabolites

The following command exports a new generated model without Dead-End Metabolites from an input SBML model.

```
$ findCPcli -i model.xml -swD new_model.xml
```

## 7 7. Refine model with FVA

findCPcli can generate a new model in which the flux bounds of the reactions have been updated with the values obtained in the computation of FVA . In this way the model can receive a different topology and the number of chokepoints, essential reactions or dead reactions, among others, can vary.

```
$ findCPcli -i model.xml -sF new_model.xml
```

Alternatively a new model can be generated refined with FVA and with DEMs removed after.

```
$ findCPcli -i model.xml -swDF new_model.xml
```

## 8 8. Tool parameters

More information about the parameters of the tool can be obtained by executing findCPcli -h.

```
$ findCPcli [-h] [-v] [-l] -i <input file> [-o <output file>]
               [-cp <output file>] [-swD <output file>] [-sF <output file>]
               [-swDF <output file>]
```

(continues on next page)

```

optional arguments:
  -h, --help            show this help message and exit
  -v, --verbose          Print feedback while running.
  -l, --license          View license info.
  -i <input file>       Input metabolic model. Allowed file formats: .xml .json
                        .yaml
  -o <output file>      Output spreadsheet file with results. Allowed file
                        formats: .xls .xlsx .ods
  -cp <output file>     Output spreadsheet file with growth dependent
                        chokepoints. Allowed file formats: .xls .xlsx .ods
  -swD <output file>    Save output model without Dead End Metabolites. Allowed
                        file formats: .xml .json .yaml
  -sF <output file>     Save output model with reactions bounds updated with
                        Flux Variability Analysis. Allowed file formats: .xml
                        .json .yaml
  -swDF <output file>   Save output model with reactions bounds updated with
                        Flux Variability Analysis and without Dead End
                        Metabolites. Allowed file formats: .xml .json .yaml
  -objective <reaction id>
                        Reaction id to be used as objective function with Flux
                        Balance Analysis

```

## 9 9. Low Level API

The computation of chokepoints can also be exploited via [findCPcore](#)<sup>3</sup> which is used by findCPcli. [findCPcore](#)<sup>4</sup> documentation can be found at [readthedocs](#)<sup>5</sup>.

Example of network refinement and chokepoint computation:

```

from findCPcore import CobraMetabolicModel

model = CobraMetabolicModel("aureus.xml")

# update flux bounds with FVA
model.fva(update_flux=True)

# compute chokepoints
model.find_chokepoints()

# get chokepoints
model.chokepoints()

```

<sup>3</sup> <https://github.com/findCP/findCPcli/blob/master/LICENSE>

<sup>4</sup> <https://github.com/findCP/findCPcli/blob/master/LICENSE>

<sup>5</sup> <https://findcpcore.readthedocs.io/en/latest/>

## 10 10. License

findCPcli is released under GPLv3<sup>6</sup>.

For citation purposes please refer to:

---

**Note:** Oarga et al. **Growth Dependent Computation of Chokepoints in Metabolic Networks.** International Conference on Computational Methods in Systems Biology. Springer, Cham, 2020. [https://doi.org/10.1007/978-3-030-60327-4\\_6](https://doi.org/10.1007/978-3-030-60327-4_6) .

---

---

<sup>6</sup> <https://github.com/findCP/findCPcli/blob/master/LICENSE>