Tutorial – Retrosynthesis using RetroPath2.0

Last update: 2018/09/20

This file is a step-by-step tutorial to compute retrosynthesis with RetroPath2.0 workflow. We also highlight a subtlety related to the use of the Sink and the scope generation.

We take here the example of pinocembrin, a flavonoid of medicinal interest at four enzymatic steps from *E. coli* metabolism. All needed data as well as pre-generated results are available in the `tutorial_data/pinocembrin` folder.

Notice: the `tutorial_data` folder also contains folders for other examples (Beta-carotene, nanringenin and violacein). These examples can be reproduced by following the procedure described here (see "Step-by-step how-to" part) and using the appropriate input parameters (provided in a specific `readme.pdf` file for each example).

Content

Biological context

> *"L-phenylalanine is first deaminated to yield cinnamic acid by the action of phenylalanine ammonia lyase (PAL). The resulting cinnamic acid is converted to pinocembrin chalcone by 4-coumarate-CoA ligase (4CL) and chalcone synthase (CHS). Finally, pinocembrin chalcone is rapidly converted to pinocembrin under alkaline conditions or by chalcone isomerase (CHI)."*

Reference: PMID: 25085569, http://www.nature.com/articles/srep32640

Parameters of RetroPath2.0

Source

Source are compounds targeted by RetroPath2.0 to be used with the rules, i.e. compounds that will be used as a starting point to predict reactions. In a retrosynthesis context, the Source should be the compound(s) for which you wish to find a pathway.

Here, source is pinocembrin (`source.csv` file).

Reaction rules

Rules should encode all biological reactions that could occur in your system. In a retrosynthesis context, this list should be as exhaustive as possible to reflect the enzymatic potential of the chassis organism.

Here, for the sake of simplicity, we will only the rules associated to enzymes of the pinocembrin pathway as described in the literature. Those rules are provided in the rules.csv file.

Details on rules generation

Using the references, (i) find the EC number of the concerned enzymes, then (ii) manually query to MetaNetX to gather the MNXR identifiers. Below the EC number(s) and associated MNXR IDs for each step:

Enzyme	EC number(s)	MetaNetX reactions
PAL	4.3.1.24, 4.3.1.25	MNXR7145, MNXR93681
4CL	6.2.1.12	MNXR1041, MNXR2251, MNXR227, MNXR14993, MNXR60189
CHS	2.3.1.74	MNXR84871, MNXR85701, MNXR85702, MNXR27480
CHI	5.5.1.6	MNXR60602, MNXR70709, MNXR73989, MNXR84948, MNXR85459, MNXR85703, MNXR76242

All those MNXR were then converted into reaction SMARTS to be used by the workflow. These rules are provided in the rules.csv file.

Sink

The Sink should comprise all compounds that are considered as granted in your system. RetroPath2.0 will never try to apply any rule on Sink compounds.

In a retrosynthetic context, the Sink should be the list of all compounds that could be considered as a starting point for a biological pathway. For instance, the metabolites of your chassis organism.

We provide two distinct sinks to showcase a subtlety of the workflow related to the generation of the scope (a special file comprising the shortest paths between Source and Sink):

- A: the compound expected to be at the origin of the pathway, phenylalanine (`sink_A.csv` file)
- B: all *E. coli* compounds (`sink_B.csv` file)

An important thing to keep in mind when choosing a Sink is that a scope is generated only if all compounds needed for a path between Source and Sink can themselves be linked to the Sink.

Consequently, with Sink “A” RetroPath2.0 will not yield any scope (at 4 steps, d12) since not all compounds needed for the biological pathway from phenylalanine to pinocembrin are in the Sink. Please notice that phenylalanine is in the Sink but that it is not sufficient. On the contrary, with Sink “B”, RetroPath will yield a scope (at 4 steps, d12) because all intermediate compounds can be linked to *E. coli* metabolism.

Step-by-step How-To

This section presents a step-by-step illustrated walkthrough in order to predict metabolic pathway for producing pinocembrin using the RetroPath2.0 workflow.

Step #1 -- Configure RetroPath2.0 Input

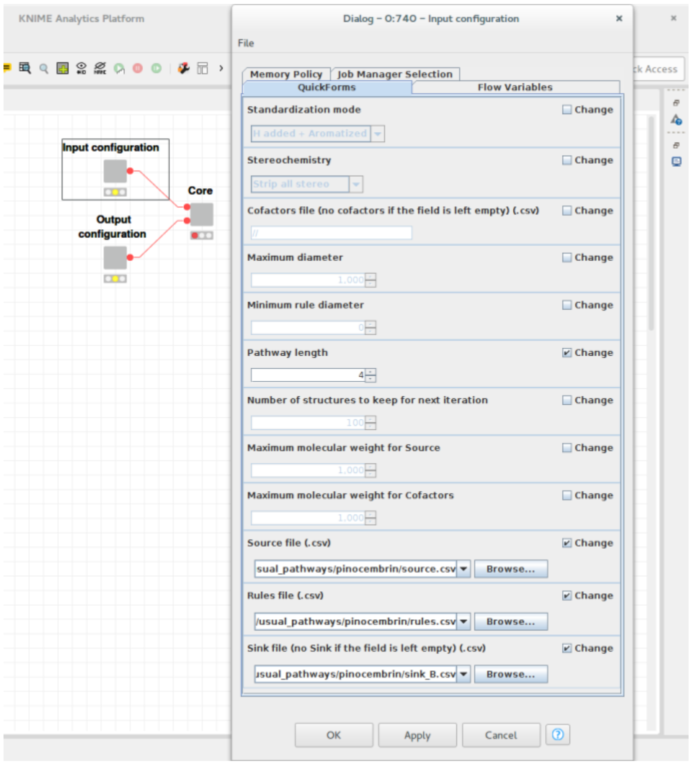
Open the Input configuration metanode (right click on `Input` metanode > `Configure...`) and change relevant fields by first checking the `Change` check boxes and then providing adequate values.

For the present example, let's change the following fields:

- Pathway length: `4`
- Source file: browse to the `source.csv` file provided in the pinocembrin folder (example of path: `path/to/tutorial_data/pinocembrin/source.csv`)
- Rules file: browse to the `rules.csv` file provided in the pinocembrin folder (example of path: `path/to/tutorial_data/pinocembrin/rules.csv`)
- Sink file: browse to the `sink_B.csv` file provided in the pinocembrin folder (example of path: `path/to/tutorial_data/pinocembrin/sink_B.csv`)

Once the desired configuration is set, click on the OK button (at the bottom of the configuration window).

Notice that one can also get detailed information on each parameter using the embedded help panel by clicking on the Question mark "**?**" button (at the bottom of the configuration window).



Standardization mode ☐ Change
H added + Aromatized

Stereochemistry ☐ Change
Strip all stereo

Cofactors file (no cofactors if the field is left empty) (.csv) ☐ Change
//

Maximum diameter ☐ Change
1,000

Minimum rule diameter ☐ Change
0

Pathway length ☑ Change
4

Number of structures to keep for next iteration ☐ Change
100

Maximum molecular weight for Source ☐ Change
1,000

Maximum molecular weight for Cofactors ☐ Change
1,000

Source file (.csv) ☑ Change
sual_pathways/pinocembrin/source.csv Browse...

Rules file (.csv) ☑ Change
/usual_pathways/pinocembrin/rules.csv Browse...

Sink file (no Sink if the field is left empty) (.csv) ☑ Change
ısual_pathways/pinocembrin/sink_B.csv Browse...

Step #2 -- Configure RetroPath2.0 Output

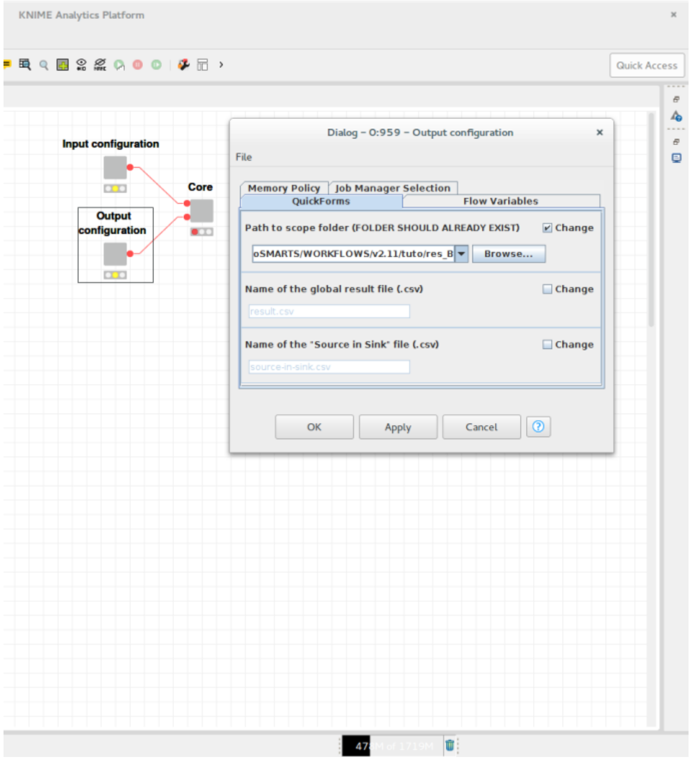
Open the `Output configuration` metanode and change the relevant fields (first check the `Change` check boxes, second provide values).

Let's change the following fields:

- Path to the result folder: browse to the folder that will contains all results (example of path: `path/to/tuto/res_B`).

Please notice that this output folder should exist and be empty before moving to the next step. Once the desired configuration is set, click on the `OK` button (bottom of the configuration window).

Notice that one can also get detailed information on each parameter using the embedded help panel by clicking on the Question mark "**?**" button (at the bottom of the configuration window).



Path to scope folder (FOLDER SHOULD ALREADY EXIST) ☑ Change

oSMARTS/WORKFLOWS/v2.11/tuto/res_B

Name of the global result file (.csv) □ Change

result.csv

Name of the "Source in Sink" file (.csv) □ Change

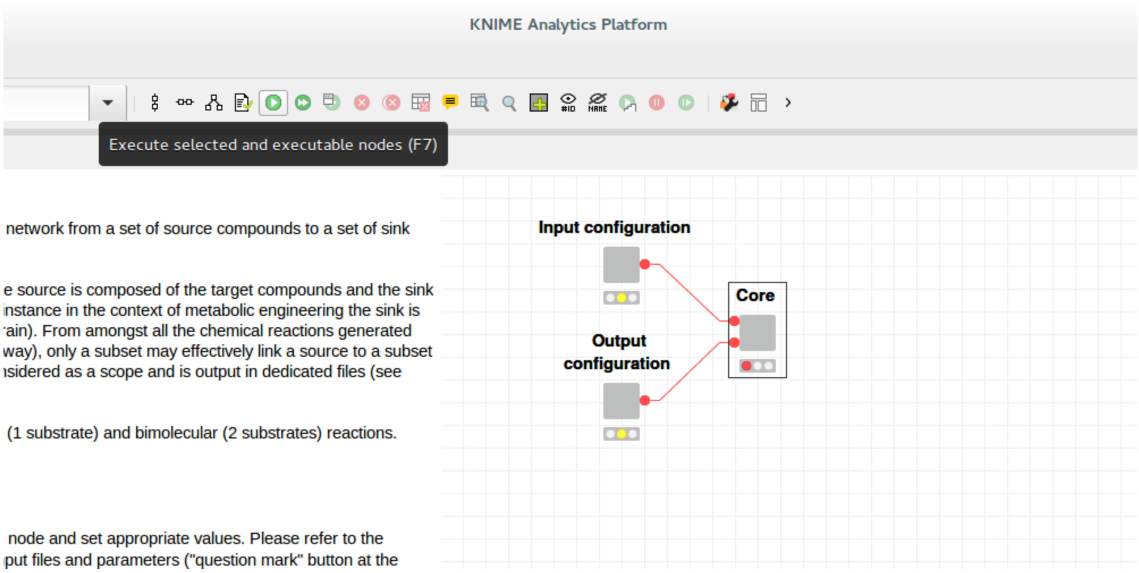
source-in-sink.csv

Step #3 -- Launch the computation

Select the `Core` metanode and launch its execution (right click > `Execute`).

Notice that depending of the combinatory (number of steps, diameters of rules, matching rules...) the execution time can range from few minutes to hours. But do not be afraid, this simple case should not take more than a minute!

Once the computation is completed, a green dot will appear below the Core metanode.

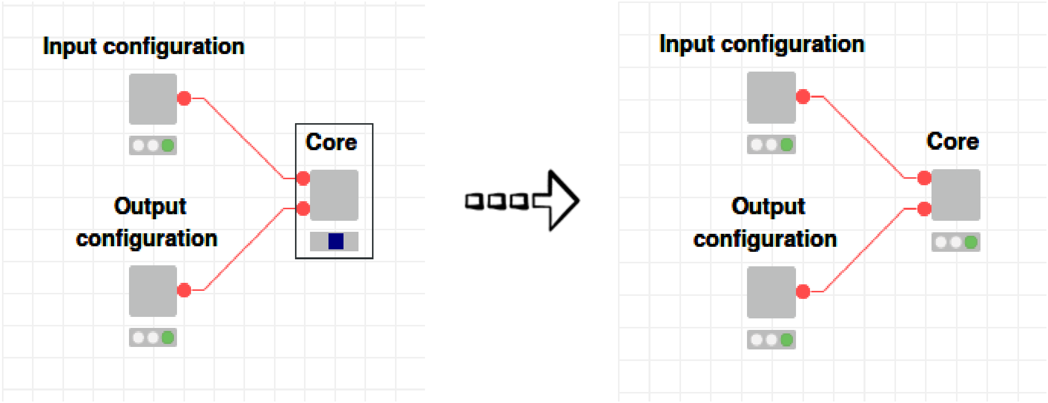


network from a set of source compounds to a set of sink

e source is composed of the target compounds and the sink
nstance in the context of metabolic engineering the sink is
ain). From amongst all the chemical reactions generated
way), only a subset may effectively link a source to a subset
isidered as a scope and is output in dedicated files (see

(1 substrate) and bimolecular (2 substrates) reactions.

node and set appropriate values. Please refer to the
put files and parameters ("question mark" button at the



Work is in progress

Work is completed

Step #4 -- Tree structure of the result folder

The output folder (here output folder is named `res_B`) now contains the generated results. It has the following structure if a scope was found:

```
res_B
|__ pinocembrin_scope.csv
|__ pinocembrin_scope.json
|__ results.csv
|__ source-in-sink.csv
|__ svg
```

Below some details on each file:

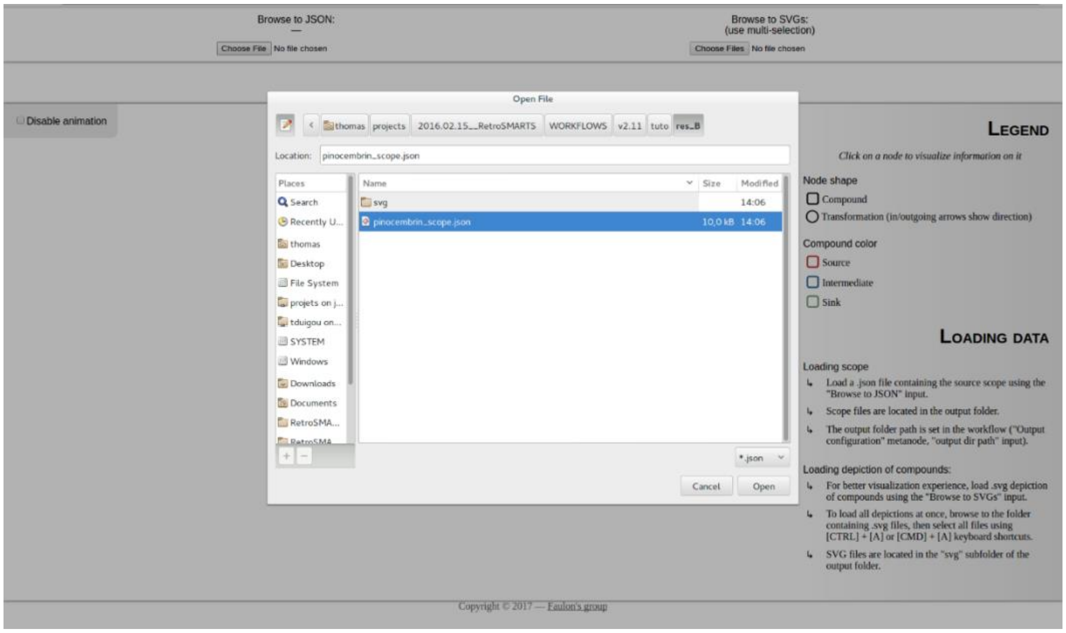
- `results.csv`: This file describes the whole metabolic graph computed by RetroPath2.0. Each row is a predicted reaction between two compounds. The detailed format and content of each column are described in the help panel of the output configuration (see **Step #2**).
- `source-in-sink.csv`: This file describes eventual source compounds that are already in the set of sink compounds. Notice that in such a case the workflow do not attempt to iterate on these compounds.
- `pinocembrin_scope.csv`: This file describes the scope, i.e. the metabolic subgraph that (i) only contains reactions contributing to link the source compound to some of the sink compounds and in which (ii) all initial substrates are sink compounds.
- `pinocembrin_scope.json`: This file also describes the scope but in a `.json` format. It can be read by the Scope Viewer (see **Step #5**) in order to visualize the reactions and compounds that are involved in.
- `svg`: This folder contains the depiction of each compound involved in the scope. These depictions can used with the Scope Viewer.

Notice that if no scope is found, then the result folder will not contain the `*_scope.csv, *_scope.json` files nor the `svg` folder. It will be the case if you use the `sink_A.csv` as a sink.

Step #5 -- Visualize scope outputted by RetroPath2.0

The Scope Viewer is a modest tool dedicated to the visualization of scope files outputted by RetroPath2.0. It is provided in the `scope_viewer` folder.

One can use it by opening the `scope_viewer.html` file in a modern web browser (e.g.: Firefox, Opera, Google Chrome, etc.). Instructions are then provided within the tool and consist of browsing to the `*_scope.json` file (see **Step #4**) in order to visualize the scope metabolic graph.



Location: pinocembrin_scope.json

Name	Size	Modified
svg		14:06
pinocembrin_scope.json	10,0 kB	14:06

LEGEND

Click on a node to visualize information on it

Node shape
- □ Compound
- ○ Transformation (in/outgoing arrows show direction)

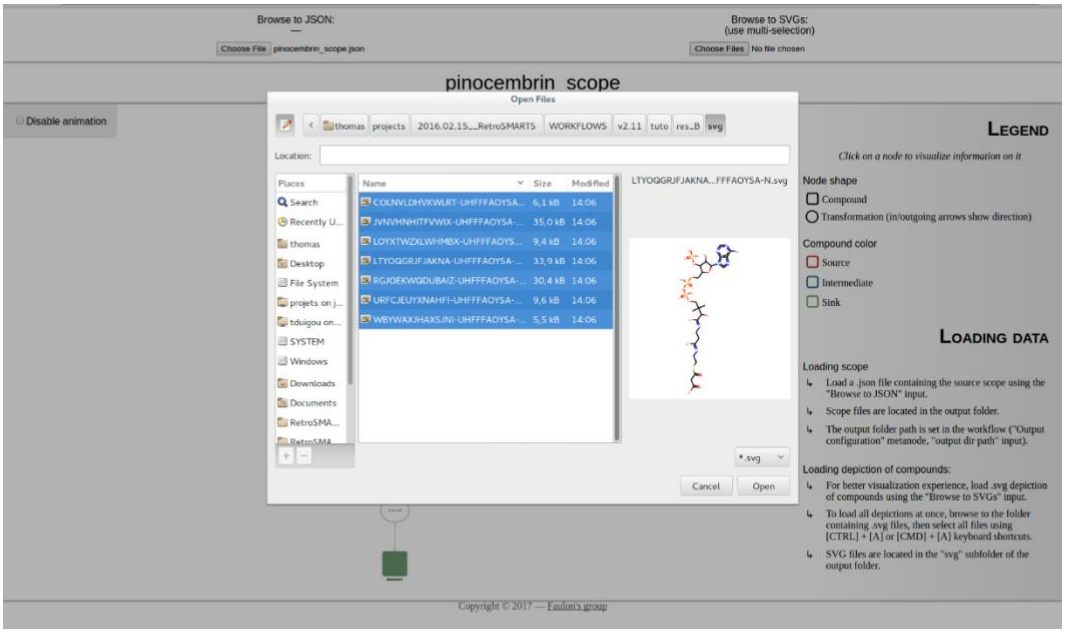
Compound color
- □ Source
- □ Intermediate
- □ Sink

LOADING DATA

Loading scope
- ↳ Load a .json file containing the source scope using the "Browse to JSON" input.
- ↳ Scope files are located in the output folder.
- ↳ The output folder path is set in the workflow ("Output configuration" metanode, "output dir path" input).

Loading depiction of compounds:
- ↳ For better visualization experience, load .svg depiction of compounds using the "Browse to SVGs" input.
- ↳ To load all depictions at once, browse to the folder containing .svg files, then select all files using [CTRL] + [A] or [CMD] + [A] keyboard shortcuts.
- ↳ SVG files are located in the "svg" subfolder of the output folder.

One can additionally browse and select all the depictions in the `svg` folder for displaying the compounds structures.



Name	Size	Modified
COLNVLDHVKWLRT-UHFFFAOYSA...	6,1 kB	14:06
JVNVHNHITFVWIX-UHFFFAOYSA-...	35,0 kB	14:06
LOYXTWZXLWHMBX-UHFFFAOYS...	9,4 kB	14:06
LTYOQGRJFJAKNA-UHFFFAOYSA-...	33,9 kB	14:06
RGJOEKWQDUBAIZ-UHFFFAOYSA-...	30,4 kB	14:06
URFCJEUYXNAHFI-UHFFFAOYSA-...	9,6 kB	14:06
WBYWAXJHAXSJNI-UHFFFAOYSA-...	5,5 kB	14:06

LEGEND

Click on a node to visualize information on it

Node shape
- □ Compound
- ○ Transformation (in/outgoing arrows show direction)

Compound color
- □ Source
- □ Intermediate
- □ Sink

LOADING DATA

Loading scope
- ↳ Load a .json file containing the source scope using the "Browse to JSON" input.
- ↳ Scope files are located in the output folder.
- ↳ The output folder path is set in the workflow ("Output configuration" metanode, "output dir path" input).

Loading depiction of compounds:
- ↳ For better visualization experience, load .svg depiction of compounds using the "Browse to SVGs" input.
- ↳ To load all depictions at once, browse to the folder containing .svg files, then select all files using [CTRL] + [A] or [CMD] + [A] keyboard shortcuts.
- ↳ SVG files are located in the "svg" subfolder of the output folder.