

Metabolic networks : Dynamics

"genome-scale network reconstruction"

- scan DNA
- identify enzymes
- cross-ref with db of known reactions
- construct bipartite network of metabolites & reactions

e.g. CarveMe python package <https://carveme.readthedocs.io/>

- 🌊 • use Flux Balance Analysis (FBA) to study metabolic dynamics

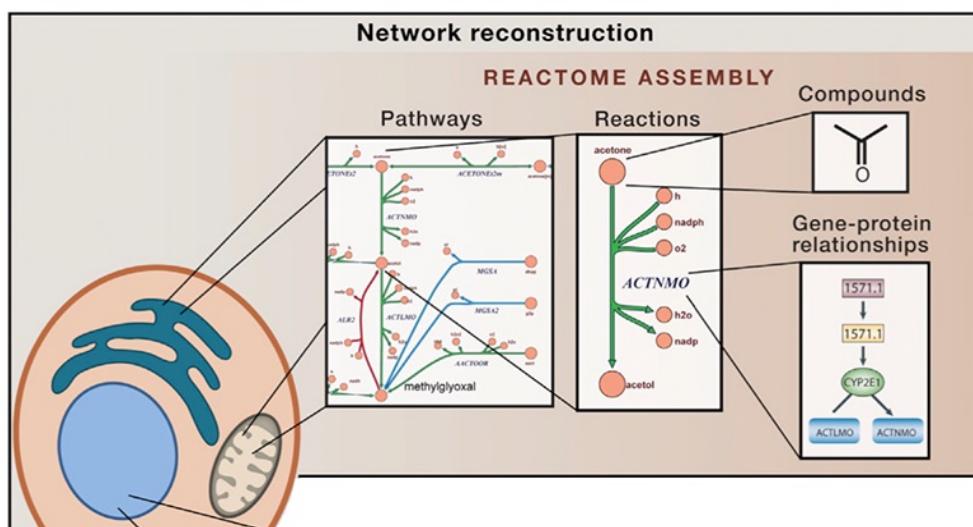


Figure 1. Network Reconstruction

An organism's reactome can be assembled in a way that is analogous to DNA-sequencing assembly. (Right to left) First the interacting compounds must be identified. Then, the reactions acting on these compounds are tabulated and the protein that catalyzes the reaction and the corresponding open reading frame is identified in the organism of interest. These reactions are assembled into pathways that can be laid out graphically to visualize a cell's metabolic map at the genome scale. Several tools for reactome assembly and curation exist, including the COBRA Toolbox (Ebrahim et al., 2013; Schellenberger et al., 2011b), KEGG (Kanehisa et al., 2014), EcoCyc (Keseler et al., 2013), ModelSeed (Henry et al., 2010b), BiGG (Schellenberger et al., 2010), Rbionet (Thorleifsson and Thiele, 2011), Subliminal (Swainston et al., 2011), Raven toolbox (Agren et al., 2013), and others.

Flux Balance Analysis

Inputs and outputs
↗ "knockouts"

- * How might an organism's metabolism behave (dynamics) under different conditions?

Caveat: FBA most well-developed for e.g. E. coli

- ① reference set of reactions is best
- ② complexities from regulation & cell-type specialization are absent

questions we can explore w/ FBA

- ① what is the max growth rate we can expect given in/out?
- ② how fast/much of a given metabolite is produced?
- ③ how robust is system?
- ④ how does behavior differ b/w O₂ & non-O₂ env.?

An aside: network dynamics

FBA is a formal model of network dynamics

similar to
dynamics
on network

network epidemic models (epi)

structure + dynamics
= function

boolean network models (PPINs)

dynamics of network
duplication-mutation-complementation models (PPINs)

toolbox model (metabolism)

Note: we studied structure first because

structure constrains dynamics

structure of metabolic net **constrains** what fluxes are **possible**
hence influence what functions they can do.

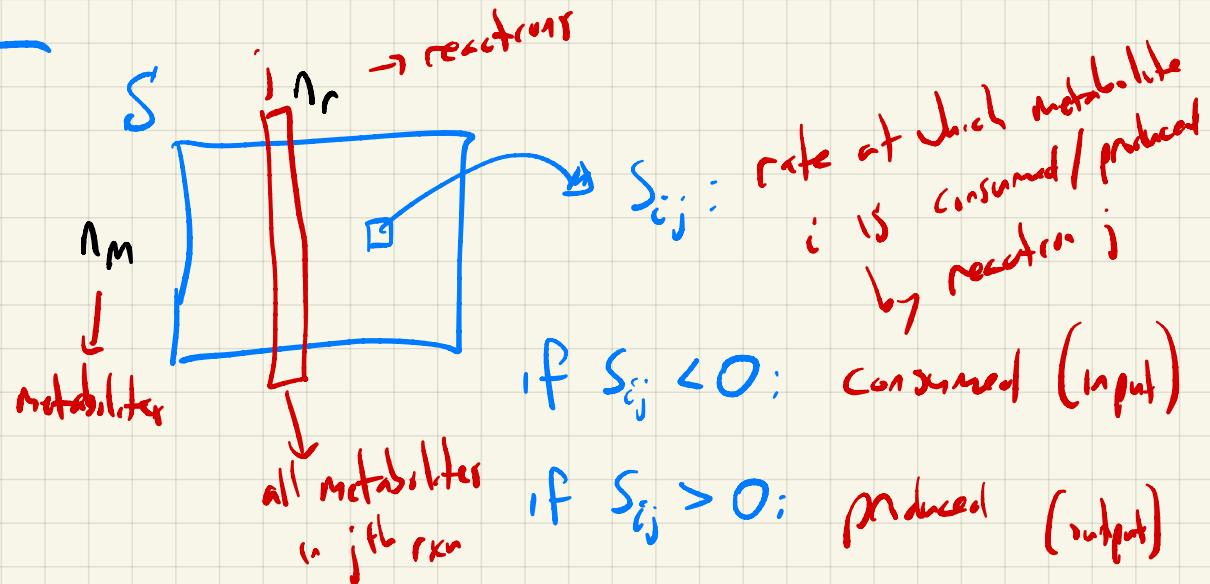
→ change $P_r(k)$ or change C

→ change dynamical properties

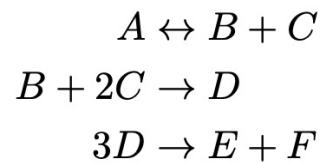
the stoichiometric matrix

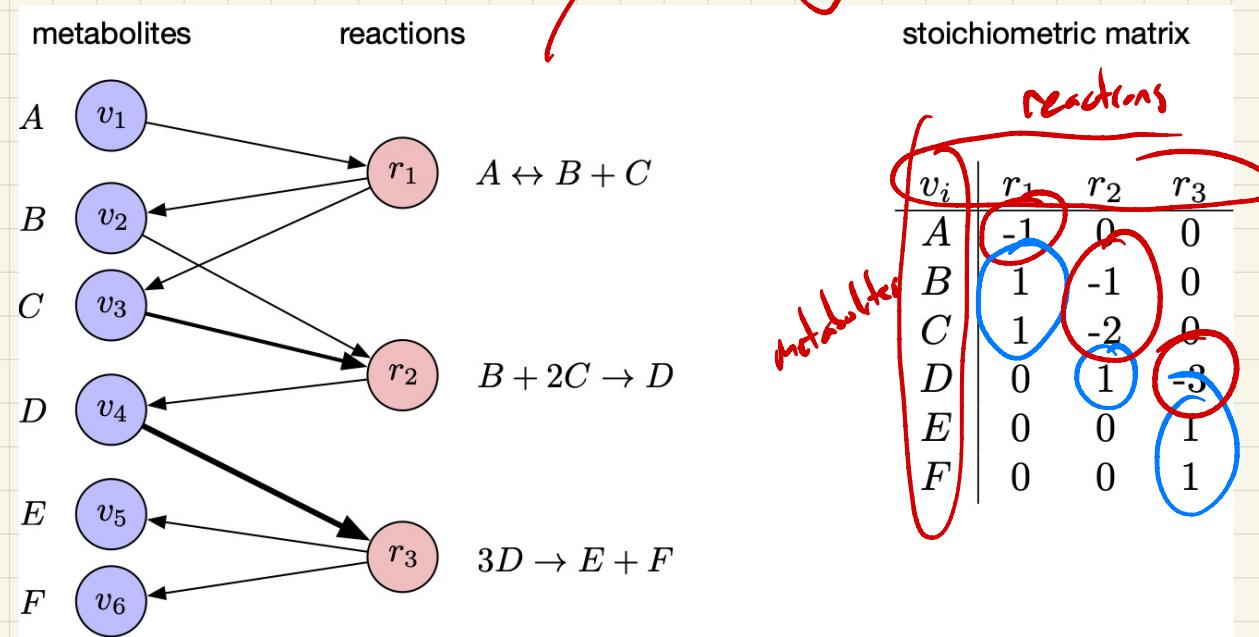
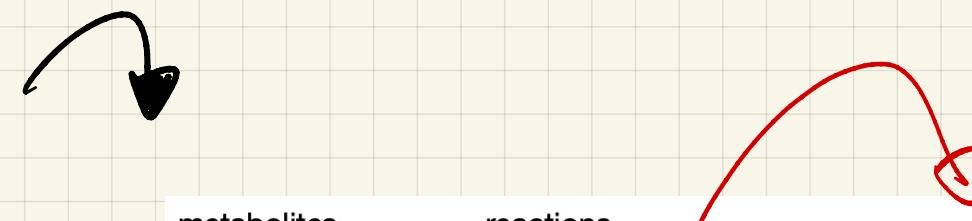
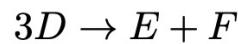
a model of metabolism

- need this representation to do FBA



Example: Consider 3 reactions + 6 metabolites





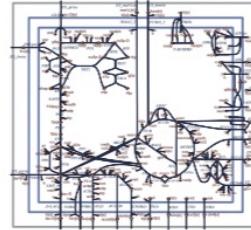
* Which compound(s) are the input? A

* Which S_{ij} entries are consumed?
* Which are produced?

* Which compound(s) are the output? E+F

①

Genome-scale metabolic network reconstruction



Reaction 1

Reaction 2

...

Reaction n

②

Bipartite network representation, of metabolic reactions and constraints

like most real world nets, S is sparse

no energetics
no regulation

only mass flow

$$\begin{array}{c} \text{Metabolites} \\ | \\ \begin{matrix} 1 & 2 & \dots & n \\ -1 & 1 & -1 & \dots \\ 1 & -2 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots \\ m & & & \end{matrix} \\ | \\ \text{Reactions} \\ | \\ \text{Stoichiometric matrix, } S \\ | \\ \text{Biomass} \\ \text{Glucose} \\ \text{Oxygen} \end{array} * \begin{array}{c} \text{Fluxes, } v \\ | \\ \begin{matrix} v_1 & v_2 & \dots & v_n \\ v_{\text{biomass}} & v_{\text{glucose}} & \dots & v_{\text{oxygen}} \end{matrix} \\ | \\ = 0 \end{array}$$

output toxic inputs

Solving FBA problems

Given a stoichiometric matrix S

Inputs \rightarrow outputs is a constraint satisfaction problem

2 kinds of constraints:

① fluxes

S tells us how things flow

require conservation of mass

(\rightarrow input flow = output flow
(at steady state!)

② input / output

add special reactions representing

Metabolites						
	Reactions	...	n	Biomass	Glucose	Oxygen
A	1	2	...			
B	-1					
C	1	-1				
D	1	-2				
...						
m				-1	-1	

Stoichiometric matrix, S

① glucose consumption

② oxygen consumption

③ biomass production

→ and ④ limits on internal fluxes

experimentally informed
metabolic
network

Solving for flux

define v the flux vector

v_j : the flux across reaction j

many possible choices of v

$$\begin{array}{c} \text{Reactions} \\ \begin{matrix} 1 & 2 & \dots & n \\ \hline A & -1 & & \\ B & 1 & -1 & \\ C & 1 & -2 & \\ D & & 1 & \\ \vdots & & & \\ m & & & \end{matrix} \end{array} * \begin{array}{c} \text{Metabolites} \\ \begin{matrix} \text{Biomass} \\ \text{Glucose} \\ \text{Oxygen} \end{matrix} \\ \downarrow \\ \begin{matrix} v_1 \\ v_2 \\ \vdots \\ v_n \\ v_{\text{Biomass}} \\ v_{\text{Glucose}} \\ v_{\text{Oxygen}} \end{matrix} = 0 \end{array}$$

Stoichiometric matrix, S

what we want: at steady state, all flux in = flux out

$$2 \rightarrow \sum v = 0$$

how many valid v are possible?

which v flux vectors are not allowed?

this is an underdetermined system ($n_r > n_m$)

search for v that maximizes growth (biomass reaction)

v_{n_r+1}

This is a linear program
Solve using simplex alg!

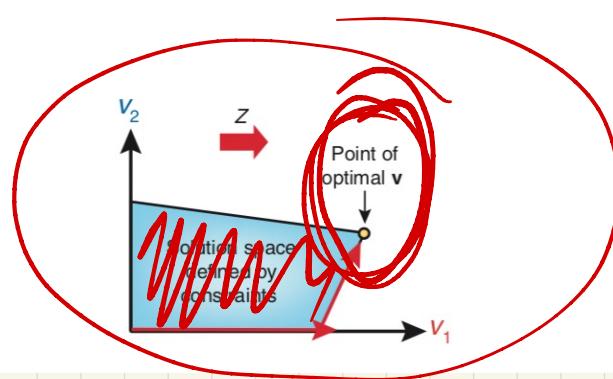
Mass balance defines a system of linear equations

$$\begin{aligned}-v_1 + \dots &= 0 \\ v_1 - v_2 + \dots &= 0 \\ v_1 - 2v_2 + \dots &= 0 \\ v_2 + \dots &= 0\end{aligned}\text{etc.}$$

Define objective function ($Z = c_1^* v_1 + c_2^* v_2 \dots$)

To predict growth, $Z = v_{\text{biomass}}$

Calculate fluxes that maximize Z



a prediction! → can verify in the lab!

$$Sv = 0$$

Limitations of FBA

FBA is powerful! But...

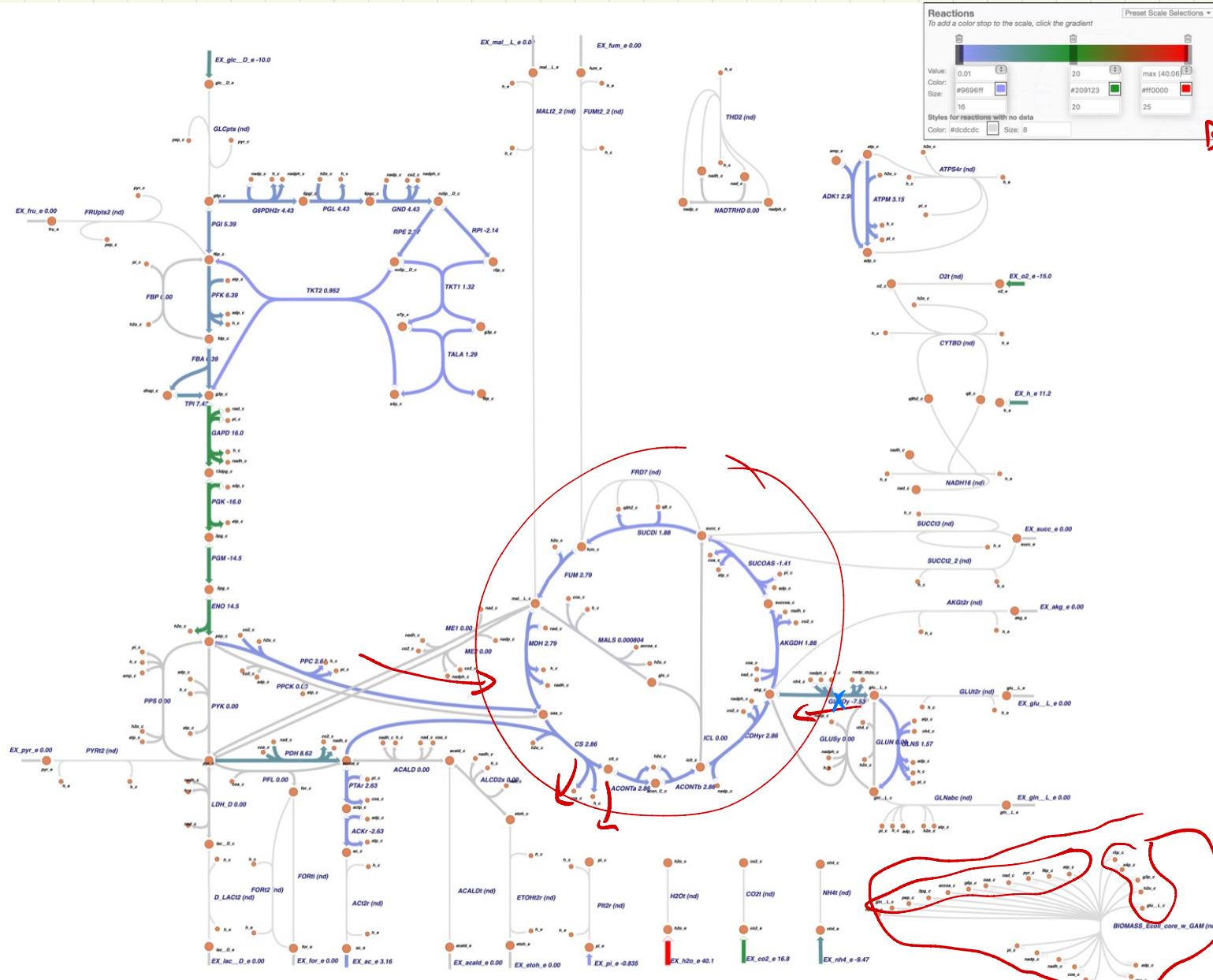
- ① FBA doesn't estimate metabolic concentrations
- ② no regulation (activation / inhibition of enzymes)
- ③ focused on generic biomass production

Simulating a FBA knockout experiment

Example using E. coli iJO1366 in CobraPy

CobraPy or PyFBA

1367 genes, $n_p = 2583$ reactions,
 $n_m = 1805$ metabolites
 + input/output



red = high flux
 green = medium flux
 blue = low flux

Note: Not all reactions shown!

basic simulation: no perturbations

```
import cobra
import cobra.test

model = cobra.test.create_test_model("ecoli") # import the model

# set the biomass objective
model.reactions.get_by_id("BIOMASS_Ec_iJ01366_core_53p95M").objective_coefficient = 0
model.reactions.get_by_id("BIOMASS_Ec_iJ01366_WT_53p95M").objective_coefficient = 1.0

# set input constraints
model.reactions.get_by_id("EX_glc__D_e").lower_bound = -10 # input, glucose
model.reactions.get_by_id("EX_o2_e").lower_bound = -15 # input, oxygen

solution = model.optimize()    solve for  $S_V=0$ 

print(f'Growth Rate: {str(solution.objective_value)} 1/h')  2 → 0.9009 1/h
model.summary()
```

outputs

inputs

this gives a baseline for comparison with knockouts

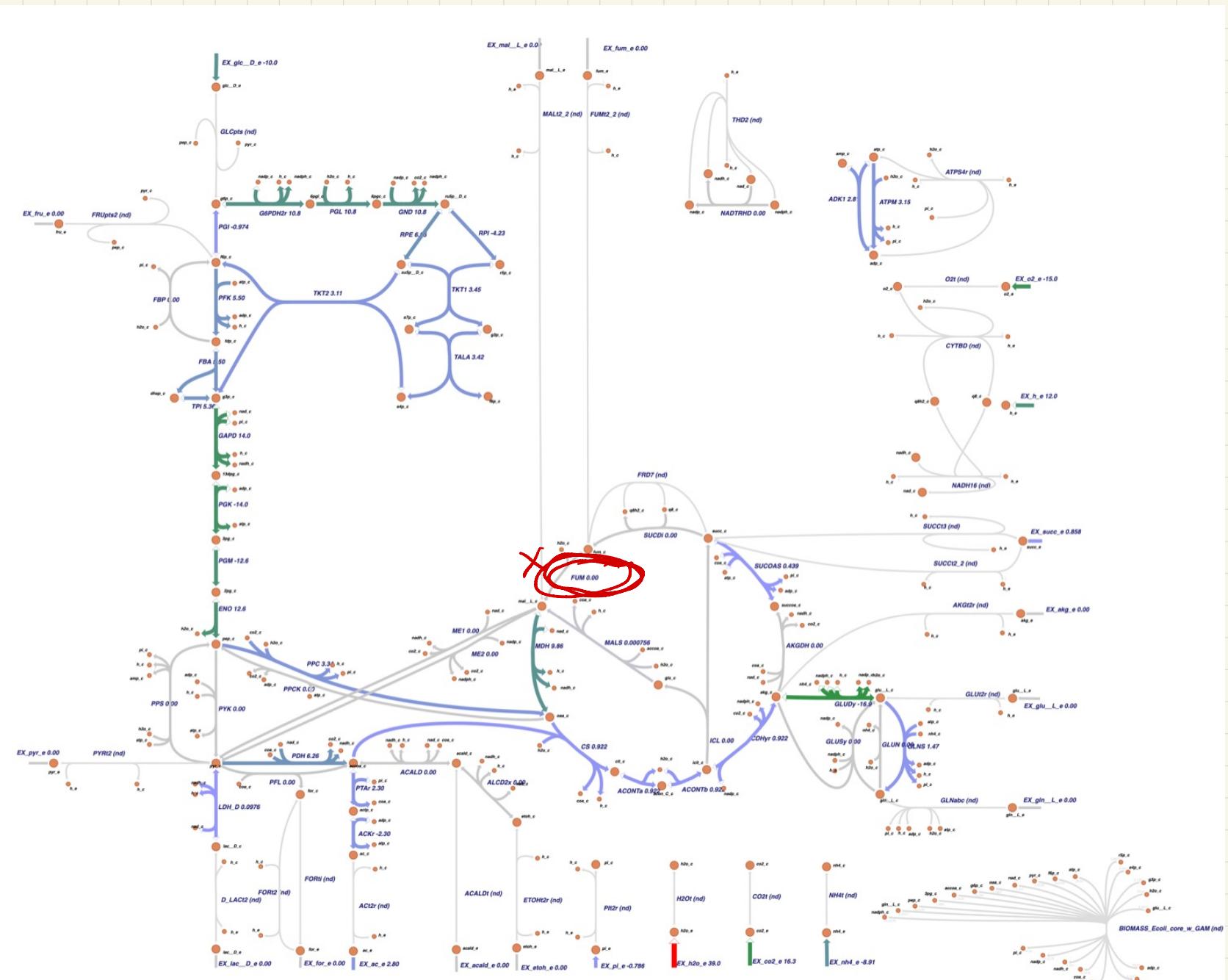
```
model.reactions.FUM.knock_out() # knockout FUM
```

set FUM flux = 0
solve $S_v = 0$ → 0, 84776 h

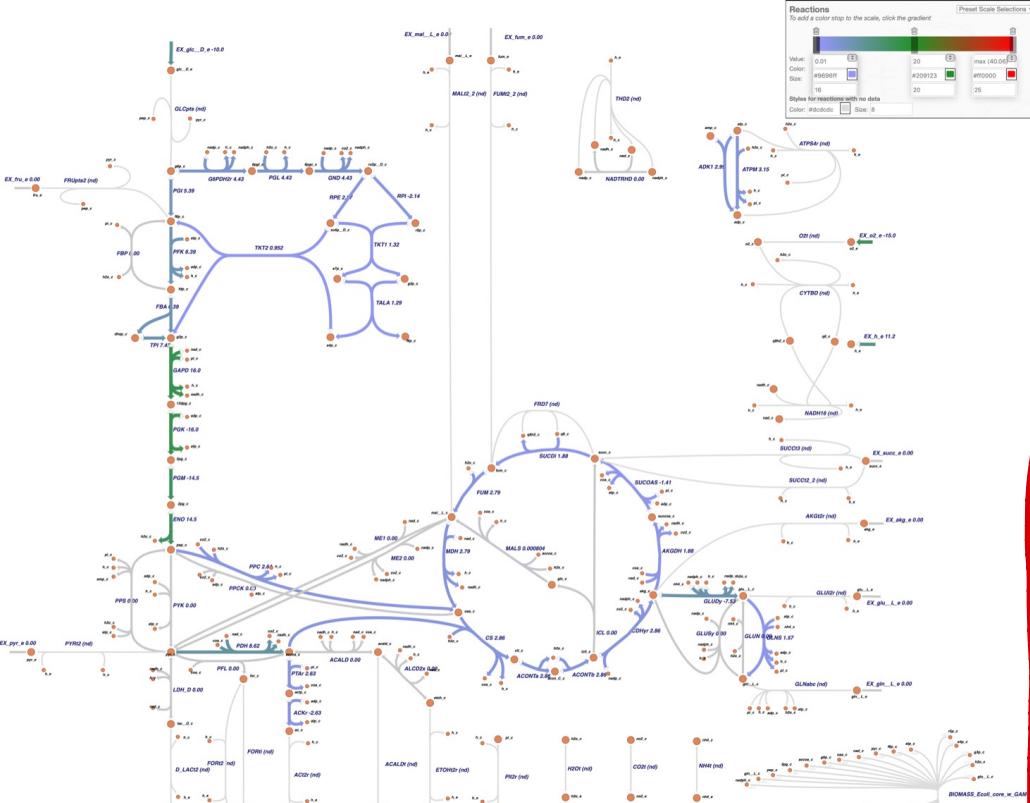
```
solution = model.optimize() # reoptimize the model
```

```
print(f'Growth Rate: {str(solution.objective_value)} 1/h')
```

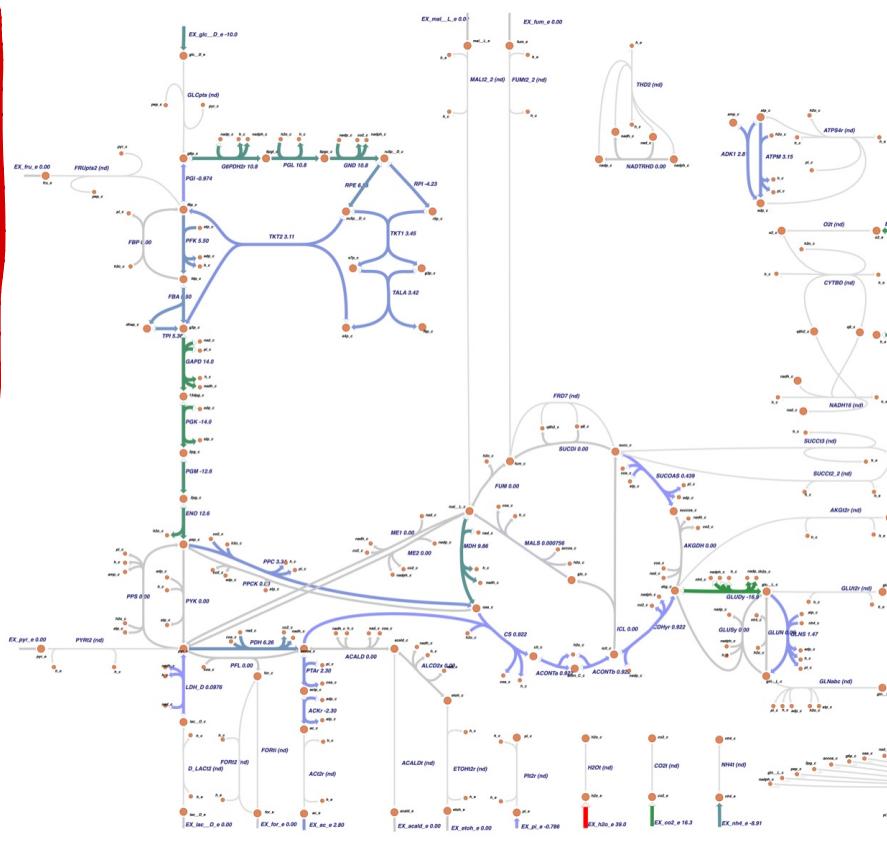
```
model.summary()
```



baseline



no FUM



$$\text{growth} = 0.9009 \text{ } \mu\text{M}$$

pretty robust!

$$\text{growth} = 0.84776 \text{ } \mu\text{M}$$

imagine knocking out other reactions

- which drop growth to 0? which increase growth? how many knockouts but still >0 growth? etc.