

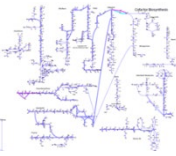
Production Envelopes



Learning Objectives

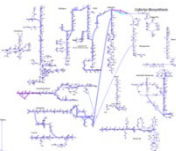
Each student should be able to:

- Explain the capabilities of robustness analysis
- Explain how shadow prices can be used in metabolic modeling
- Explain how reduced costs can be used in metabolic modeling
- Explain the capabilities of phenotype phase plane analysis
- Explain the purpose of production envelopes



Lesson Outline

- Robustness Analysis
- Shadow Prices
- Reduced Costs
- Phenotype Phase Plane Analysis
- Production Envelopes

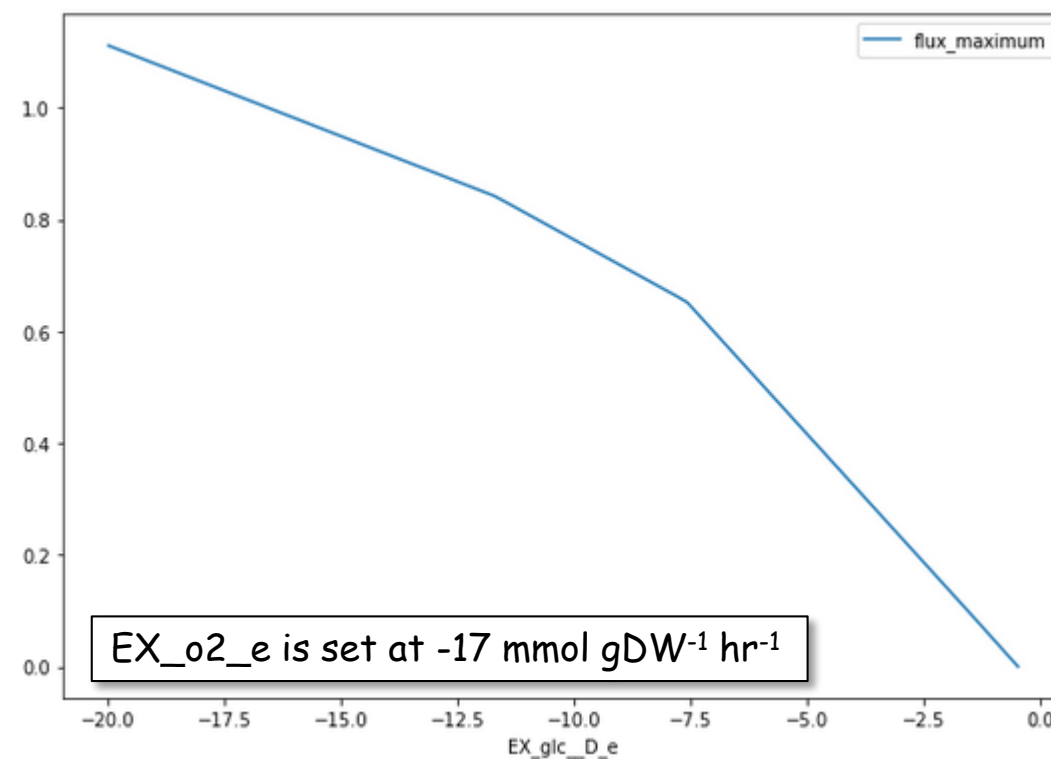


Robustness Analysis

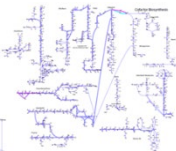
- The flux through one reaction is varied and the optimal objective value is calculated as a function of this flux. This reveals how sensitive the objective is to a particular reaction.
- In COBRApy use the "production_envelope"

```
from cobra.flux_analysis import production_envelope
```

```
RA_glc = production_envelope(model, ["reaction of interest"],
    objective="biomass function",
    carbon_sources="carbon sources",
    points="Number of points to calculate")
```

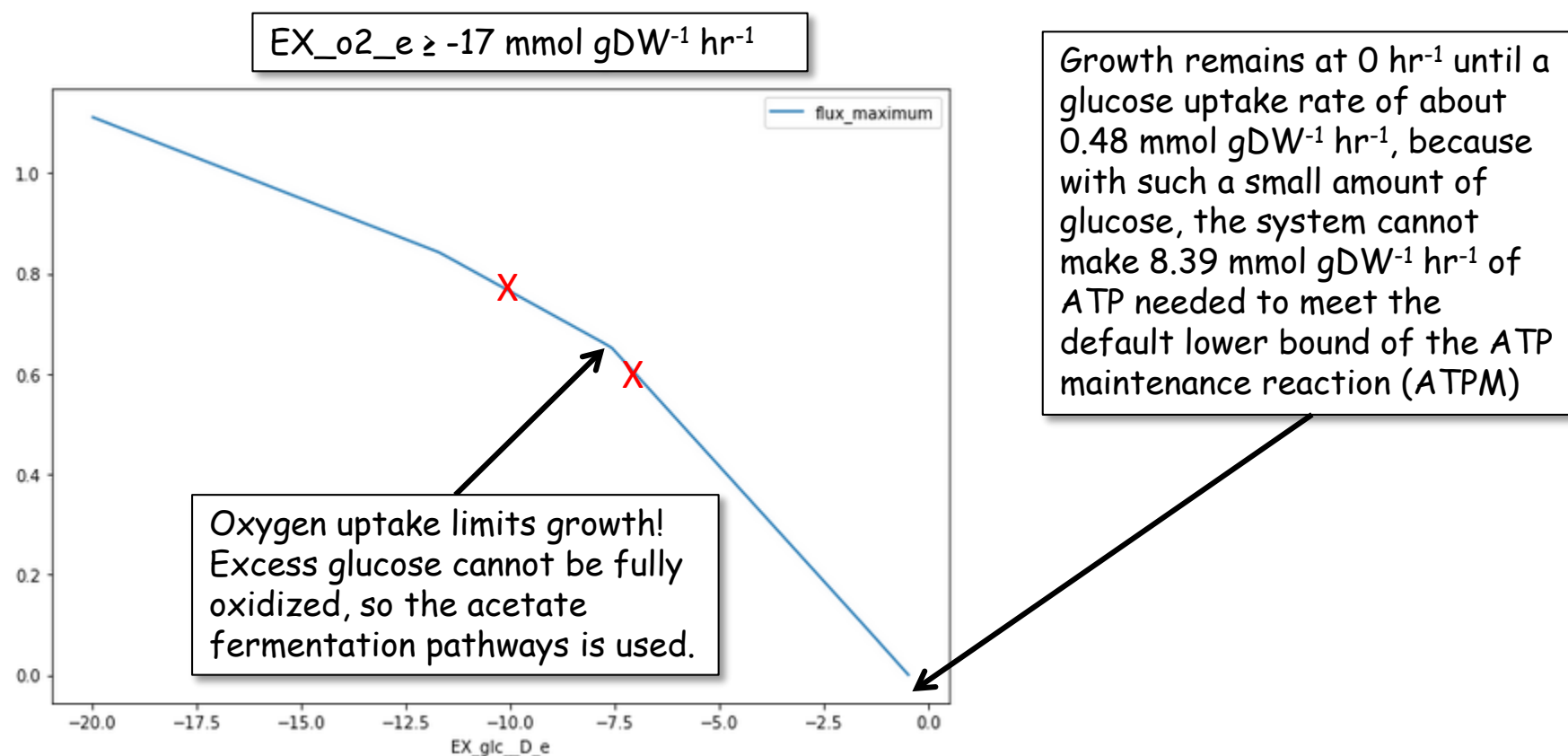


Robustness_Examples.ipynb



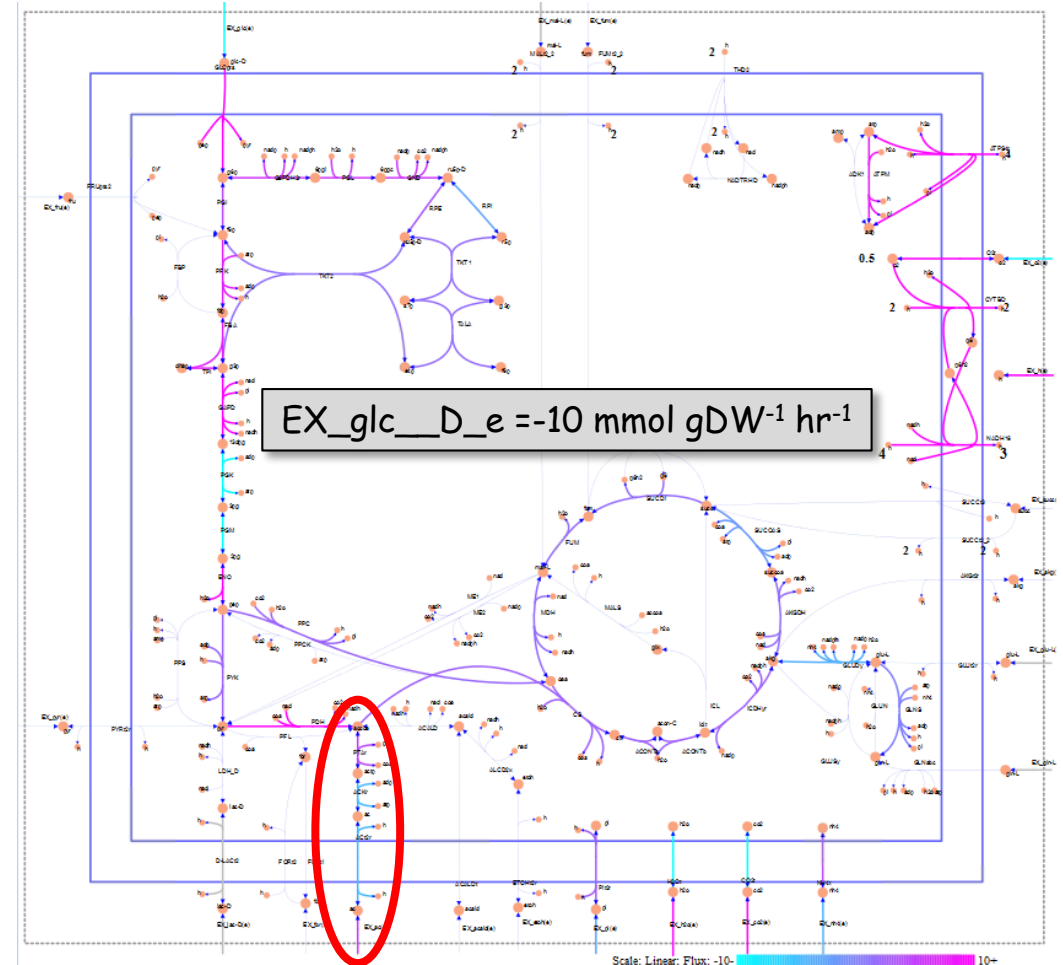
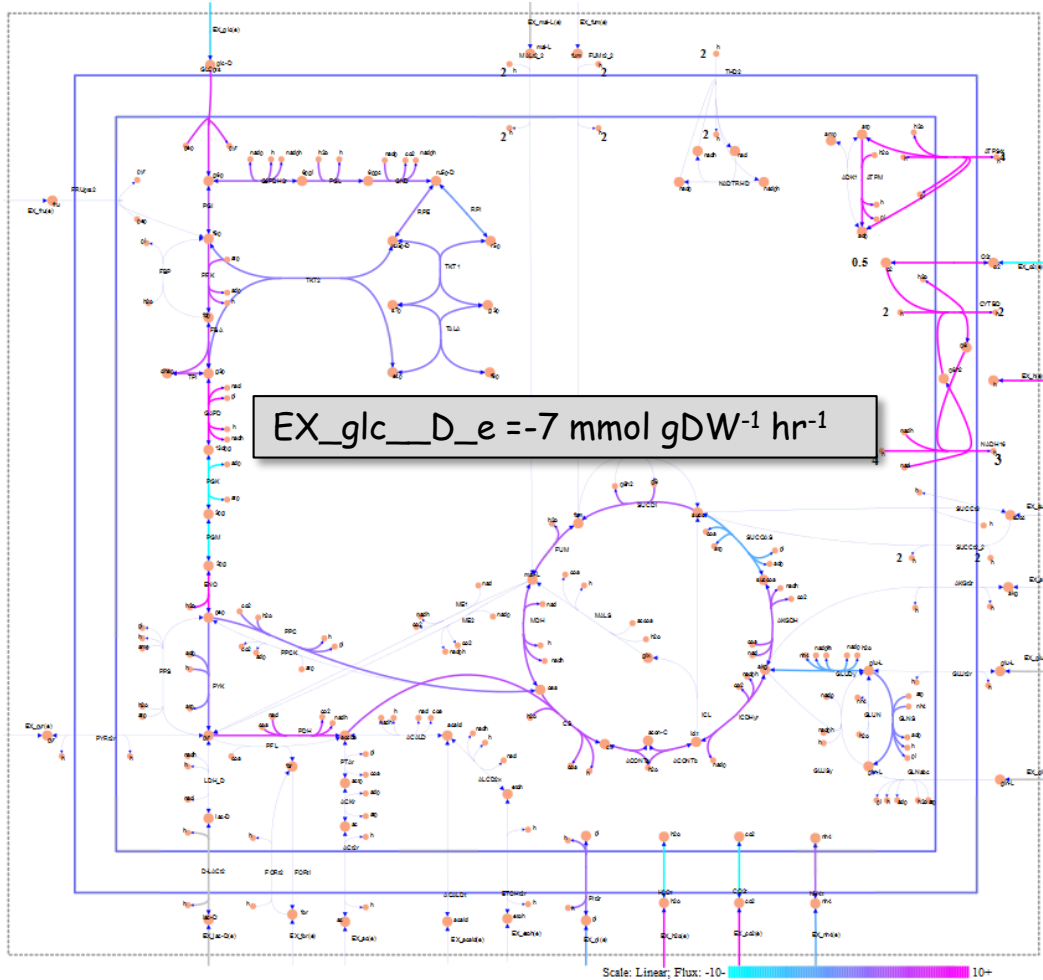
Robustness Analysis Example

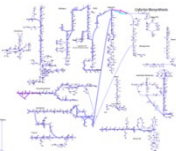
Impact of Increasing Glucose



Robustness_Examples.ipynb

Robustness Analysis Example Maps

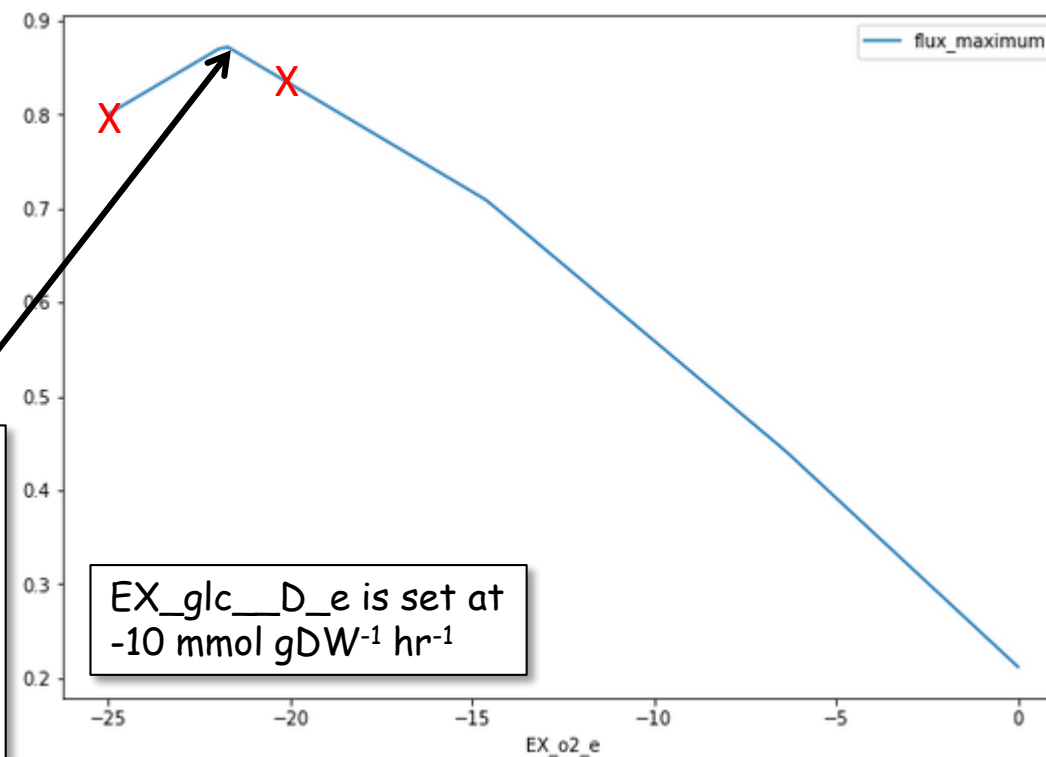




Robustness Analysis Example

Impact of Increasing Oxygen

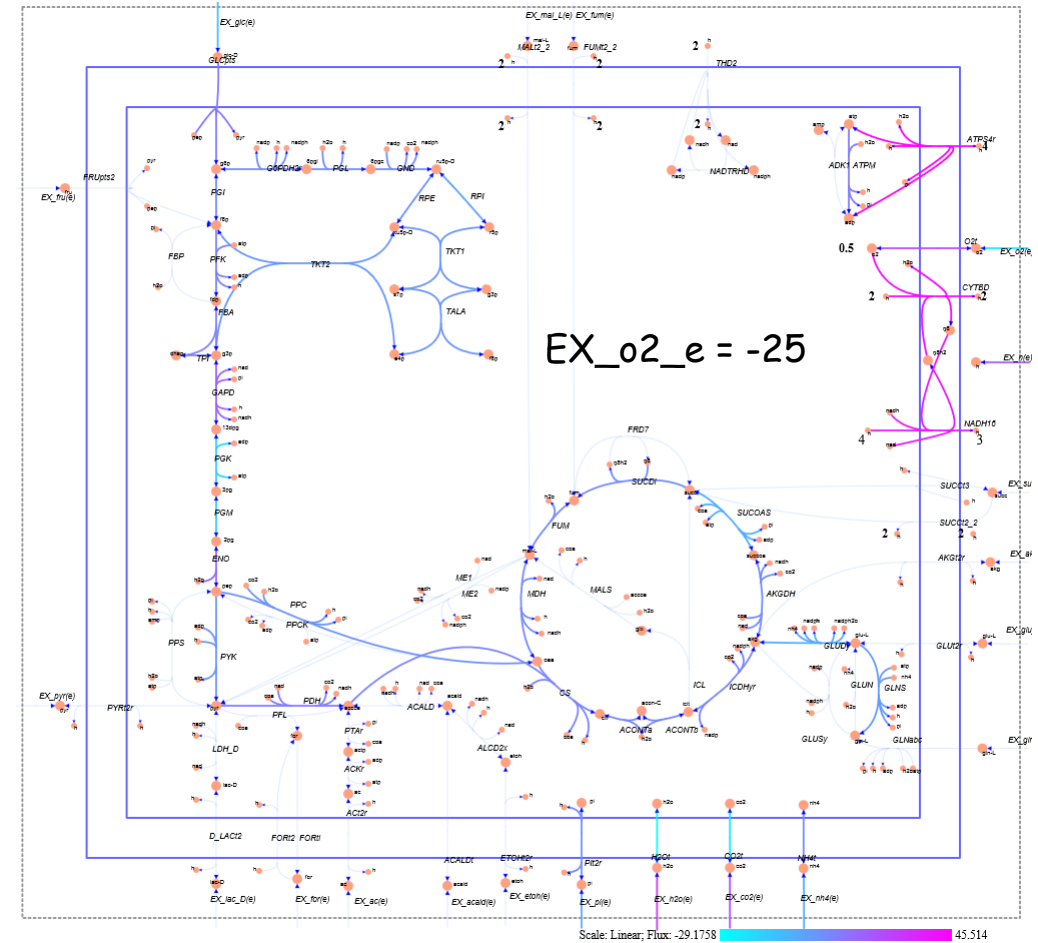
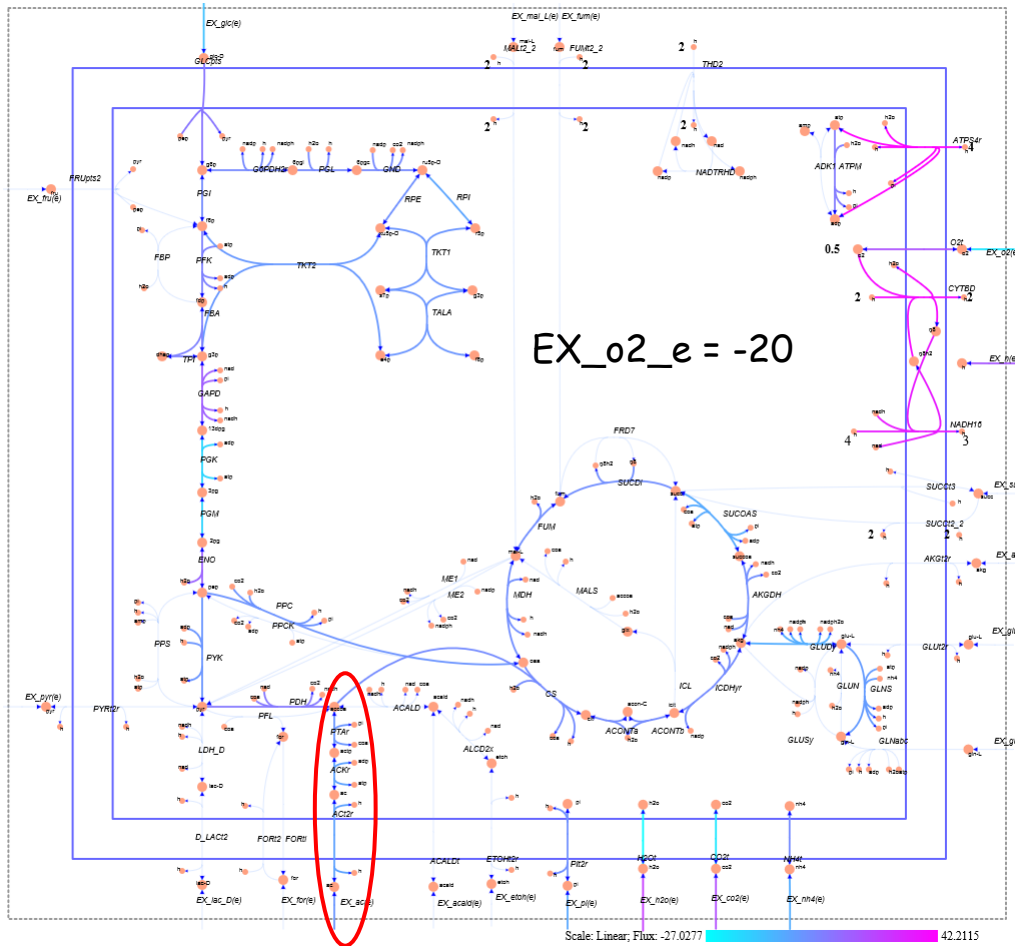
At an oxygen uptake of about 21.80 mmol gDW⁻¹ hr⁻¹, growth actually begins to decrease as oxygen uptake increases. This is because glucose becomes limiting at this point, and glucose that would have been used to produce biomass must instead be used to reduce excess oxygen.



Robustness_Examples.ipynb

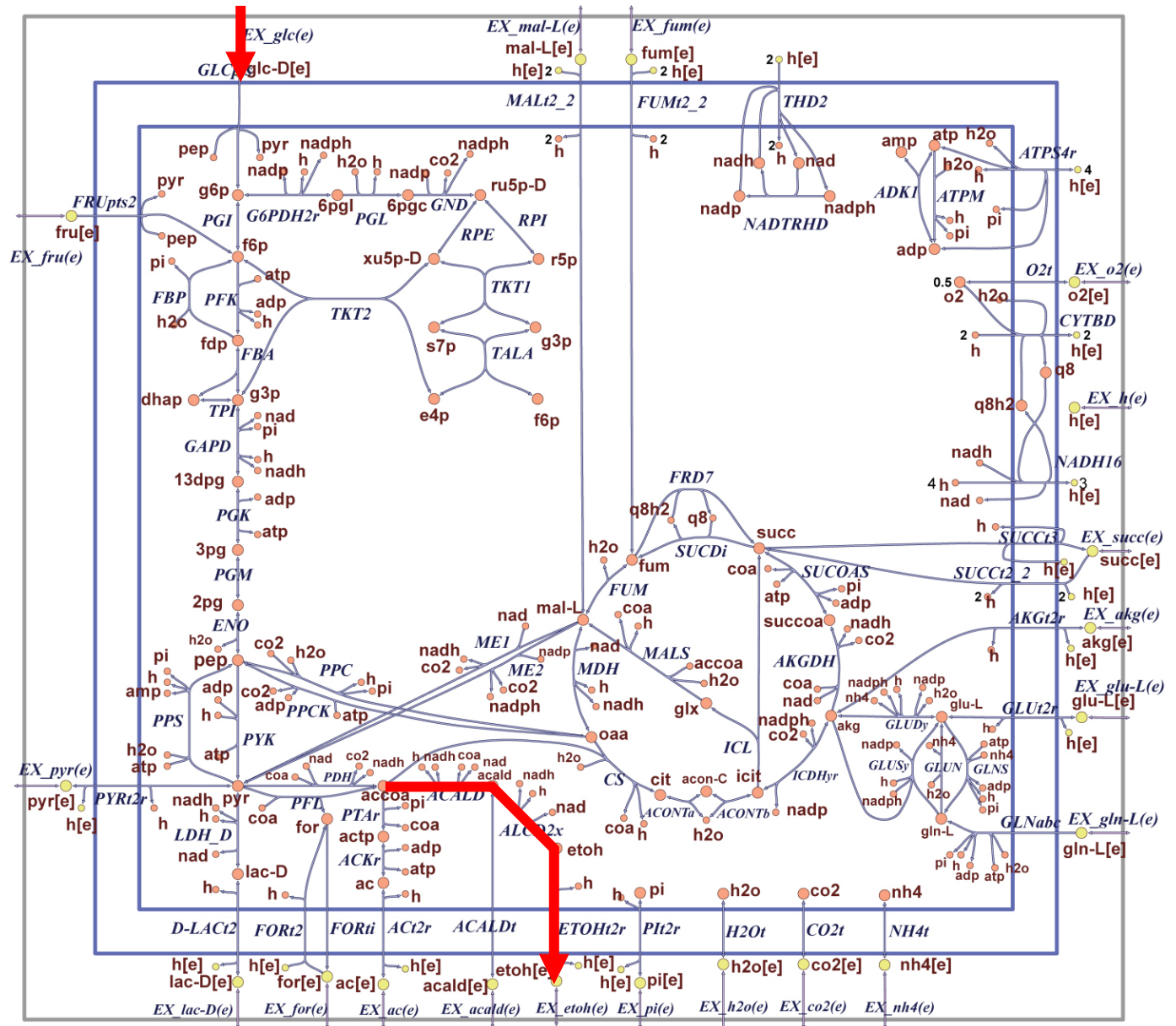
Robustness Analysis Example Maps

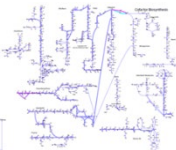
Impact of Increasing Oxygen



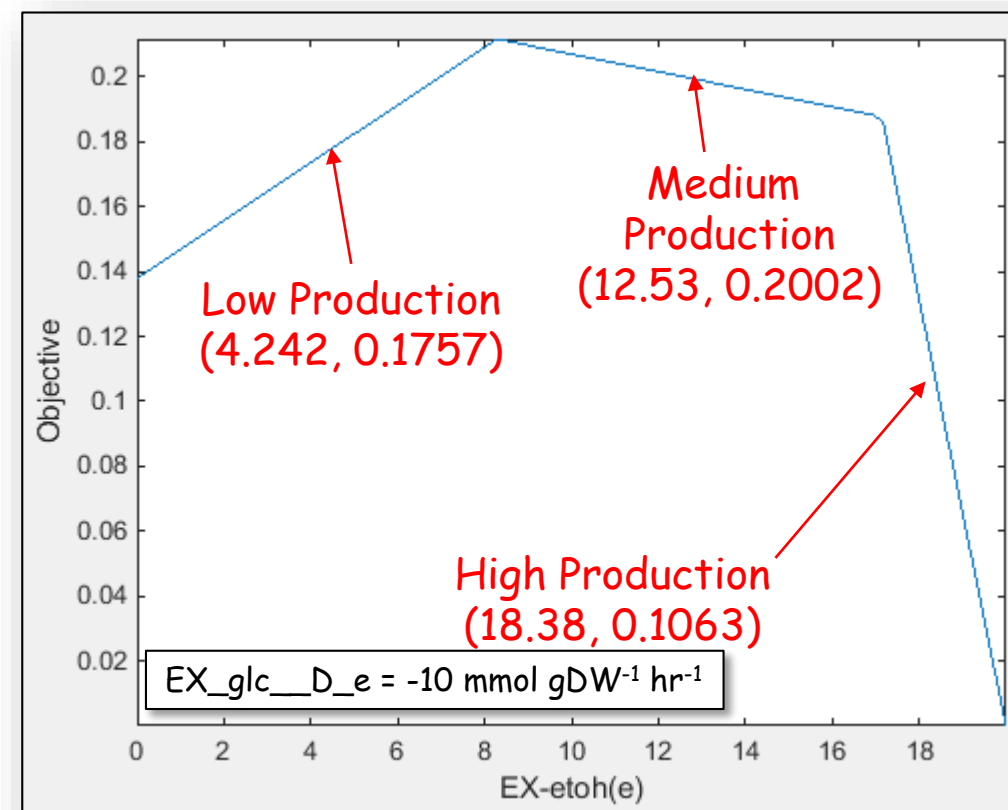
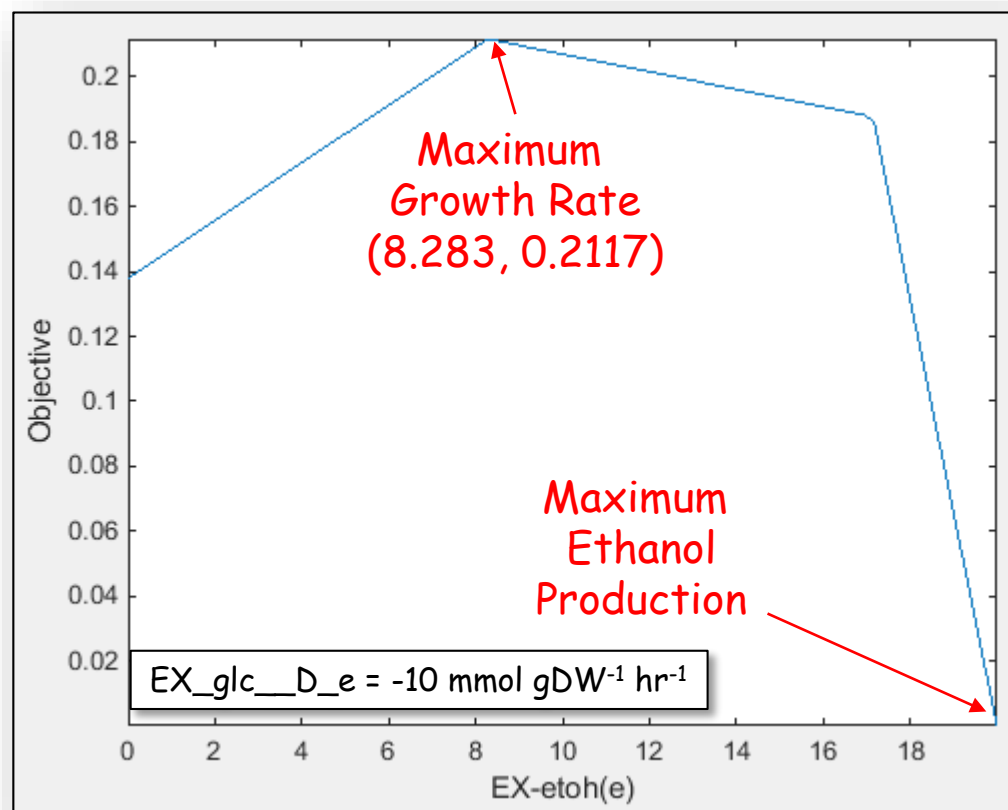
Anaerobic Ethanol Production

Orth, J. D., I. Thiele, et al. (2010). "What is flux balance analysis?" Nature biotechnology 28(3): 245-248.

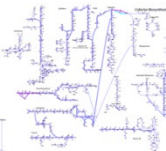




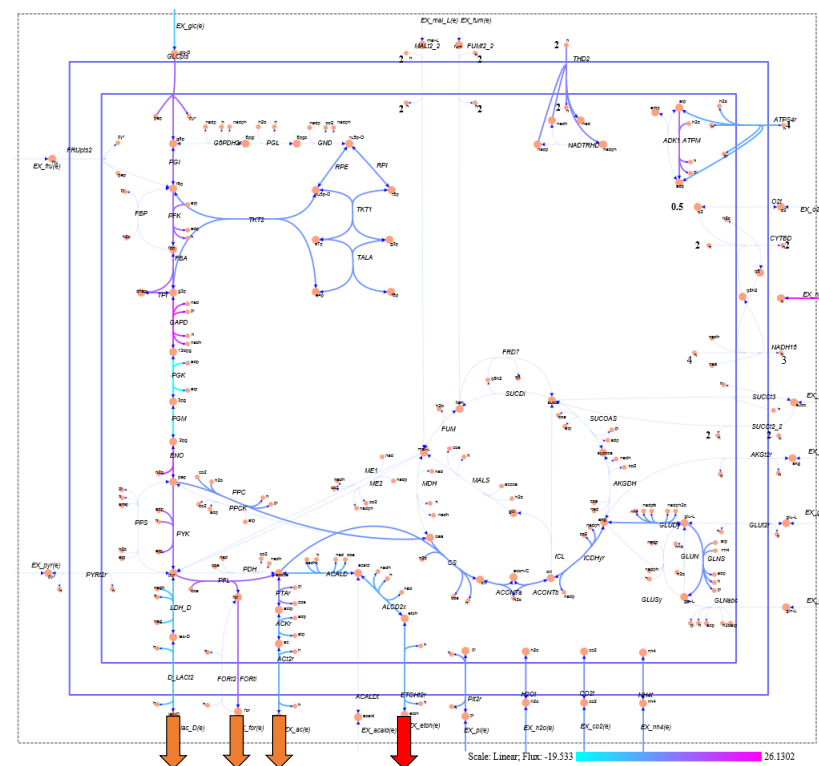
Maximum Anaerobic Ethanol Production



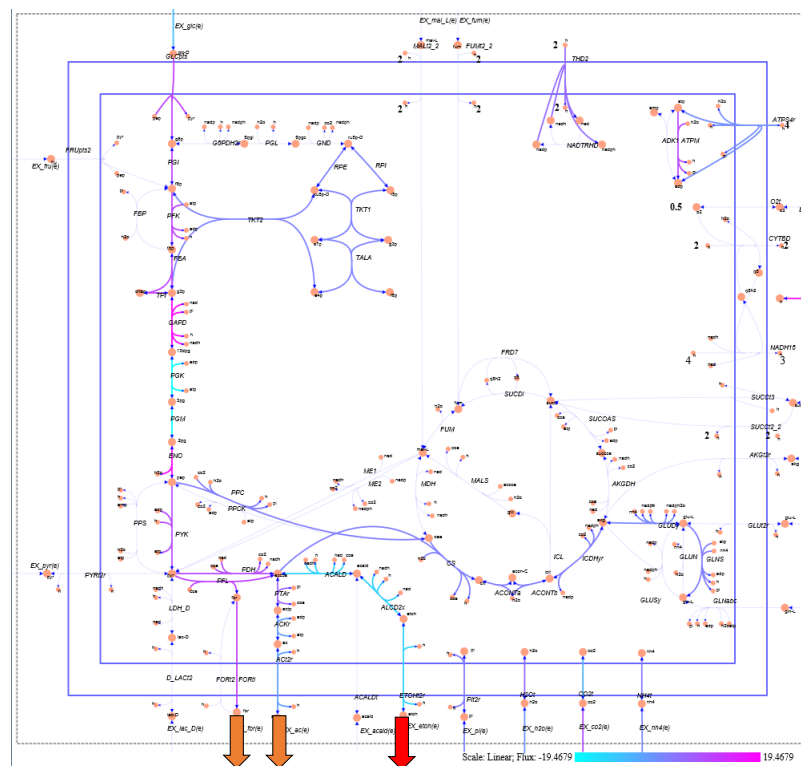
Robustness_Examples.ipynb



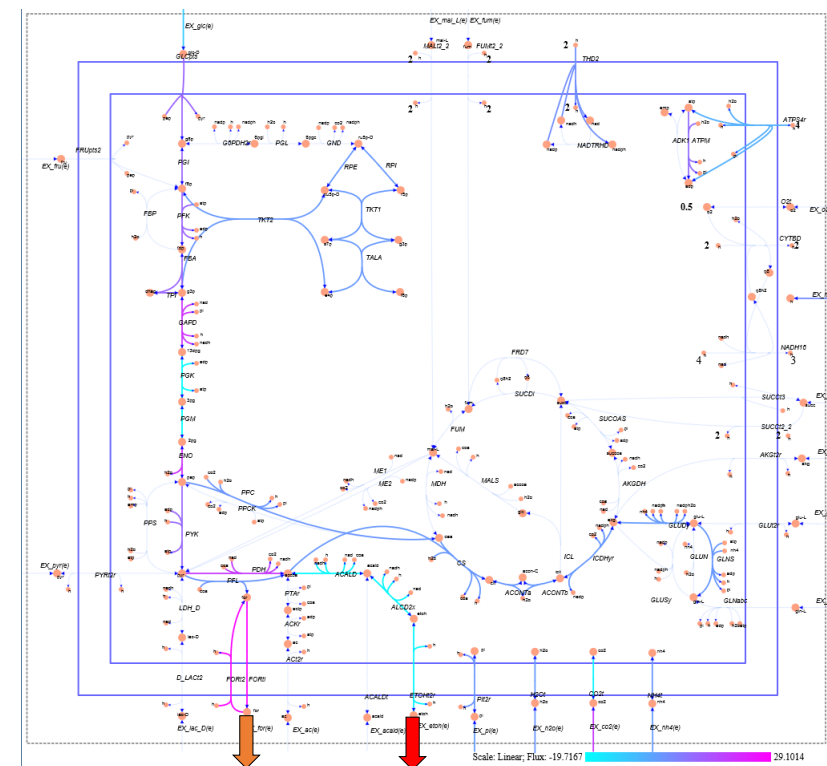
Ethanol Production Phenotype Maps



Low Production



Medium Production



High Production

Robustness Analysis Examples

```
In [1]: import cobra.test
import numpy as np
import pandas as pd
from cobrapy_bigg_client import client
import matplotlib.pyplot as plt
pd.set_option('display.max_rows', 500)
```

```
In [2]: model_orig = client.download_model('e_coli_core', save=False) # Loading the model to the simulation
model_orig.solver = 'glpk'
```

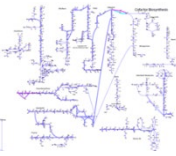
Set parameter Username
Academic license - for non-commercial use only - expires 2022-10-10

Glucose Robustness Analysis

```
In [3]: from cobra.flux_analysis import production_envelope

model = model_orig.copy()
model.reactions.EX_o2_e.lower_bound = -20
model.reactions.EX_glc__D_e.lower_bound = -30
RA_glc = production_envelope(
    model, ["EX_glc__D_e"], objective="BIOMASS_Ecoli_core_w_GAM", carbon_sources="EX_glc__D_e", points=100)
RA_glc.round(5)
```

Robustness_Example.ipynb

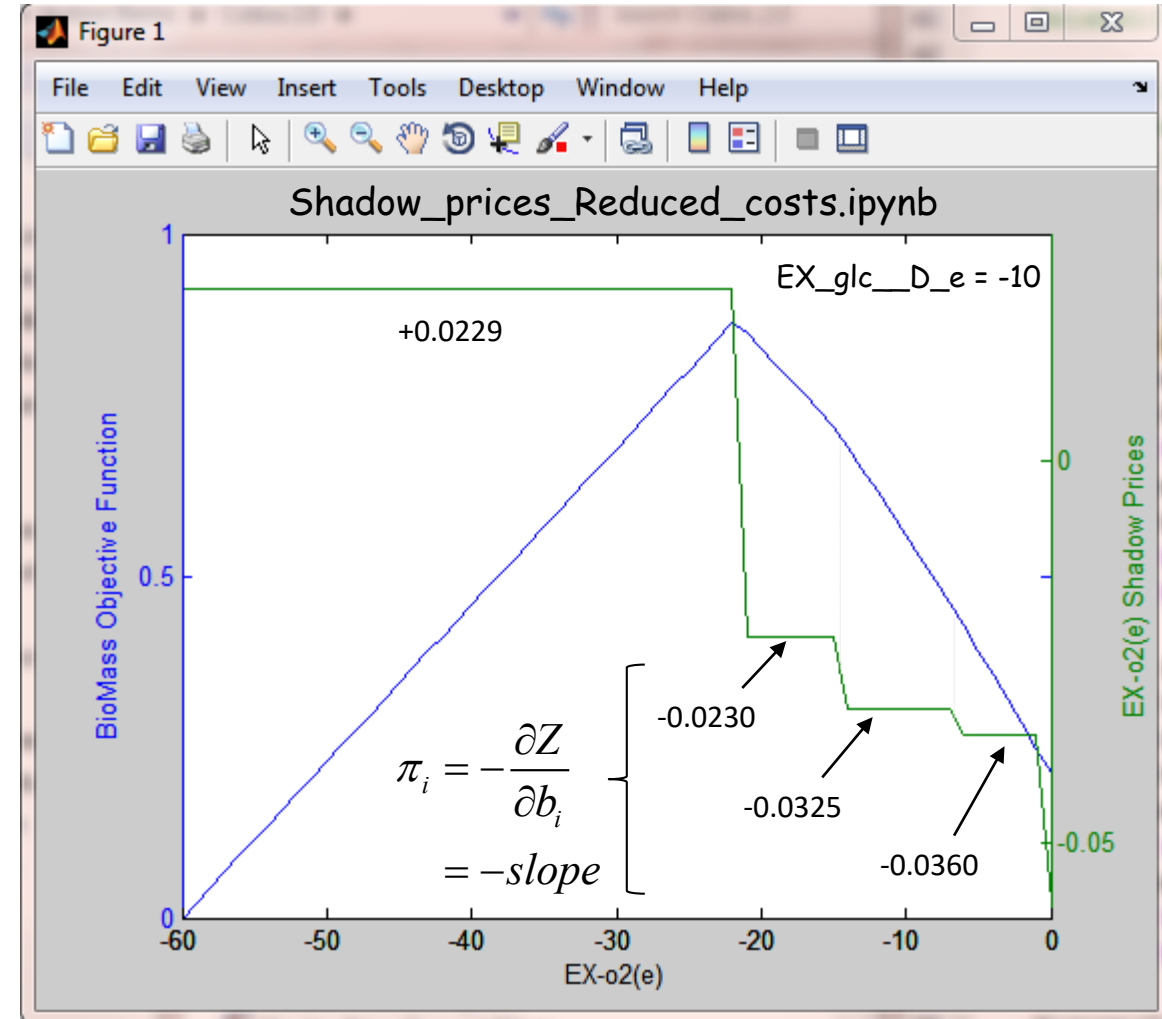


Lesson Outline

- Robustness Analysis
- • Shadow Prices
- Reduced Costs
- Phenotype Phase Plane Analysis
- Production Envelopes

Shadow Prices: Metabolites

- Shadow prices, π_i , are the derivative of the objective function, Z , with respect to the flux, b_i , of a **metabolite**.
- The shadow prices define the incremental change in the objective function if a constraining flux is incrementally changed.
- The sensitivity of an FBA solution is indicated by shadow prices. *They indicate how much the addition of a given unit of metabolite flux will increase or decrease the objective.*
- In the COBRApy tools, shadow prices can be calculated by `model.optimize()`. They can be accessed by `solution.shadow_prices` or `model.metabolites.get_by_id('x').shadow_price` where 'x' is the metabolite name.



Shadow Prices

Since

$$\pi_i = -\frac{\partial Z}{\partial b_i} \Rightarrow \pi_i \approx -\frac{\Delta Z_{GR}}{\Delta b_{o2[e]}} \Rightarrow \Delta Z_{GR} \approx -\pi_i \Delta b_{o2[e]}$$

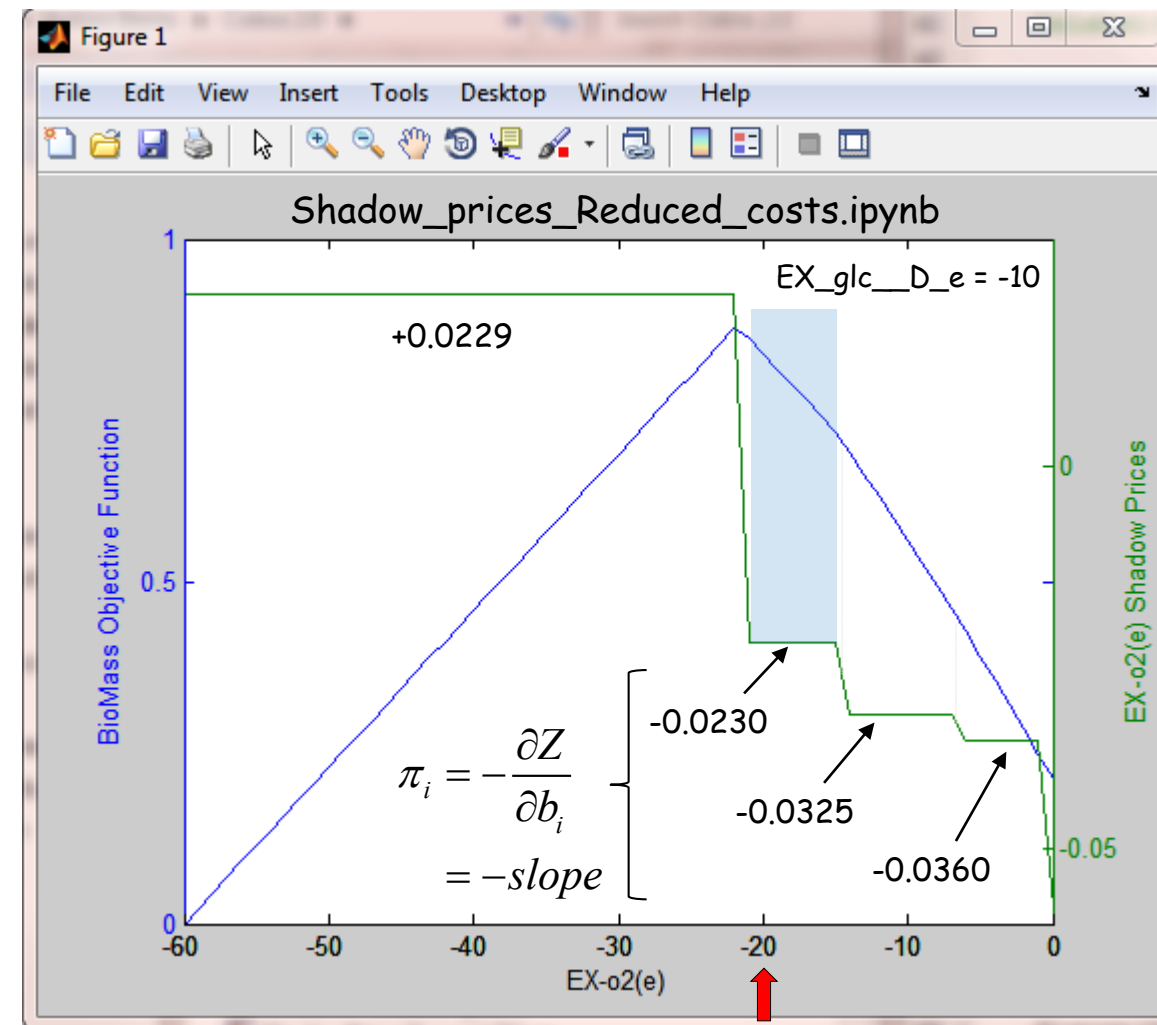
1 unit of flux of $b_{o2[e]}$

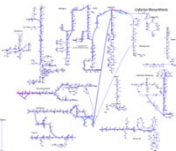
$$\Rightarrow \Delta Z_{GR} \approx -\pi_i$$

If the objective is set to maximize cell growth rate (Z_{GR}), and the shadow price (π_i), of oxygen (b_{o2_e}) is -0.0230, it means that an additional flux unit of oxygen, $o2_e$, within the EX_ $o2_e$ uptake region of -15 and -21, will increase the growth rate by 0.0230.

Each steady state solution (phenotype) will have different shadow prices. This example is based on EX_glc__D_e = -10

Some solvers return positive shadow prices!

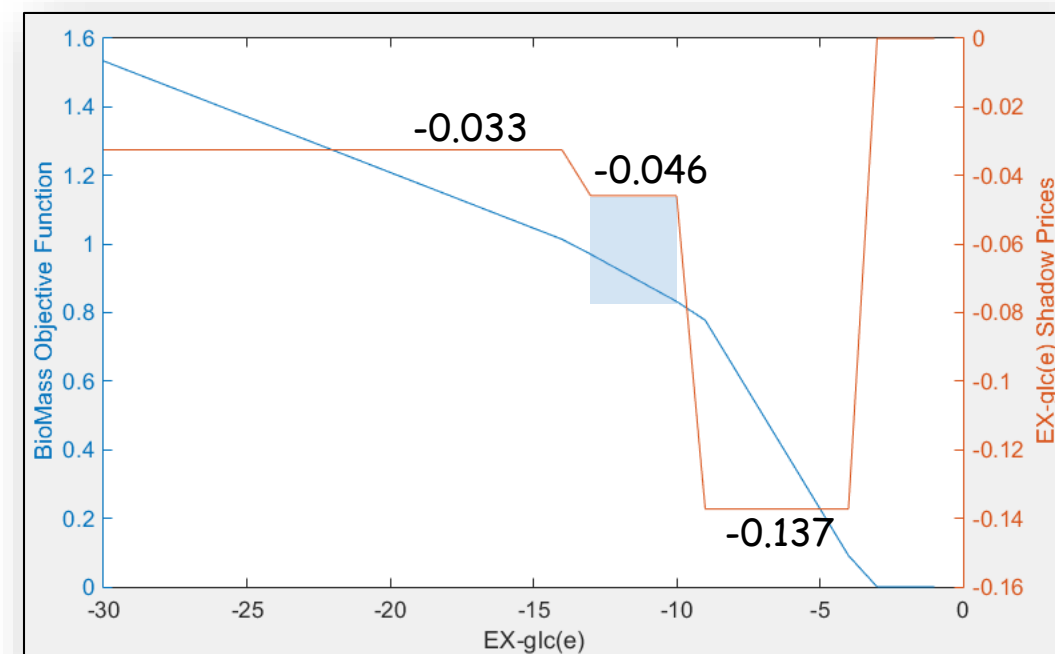




Glucose Shadow Prices: Aerobic Growth Rate Example

(Shadow_prices_Reduced_costs.ipynb)

If the objective is set to maximize cell growth rate (Z_{GR}), and the shadow price ($\pi_{glc_D_e}$), of glucose is -0.046 when $EX_o2_e = -20$, it means that an additional molecule of glucose will increase the growth rate by 0.046.



Biomass ($EX_glc_D_e = -10$)

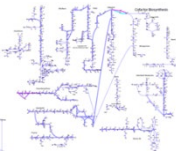
0.832

Biomass ($EX_glc_D_e = -11$)

0.878

Δ Biomass

0.046



Shadow Prices and Reduced Costs

```
In [1]: import cobra.test
import numpy as np
import pandas as pd
from cobrapy_bigg_client import client
import matplotlib.pyplot as plt
import escher
from escher import Builder
pd.set_option('display.max_rows', 500)
```

```
In [2]: model_orig = client.download_model('e_coli_core', save=False) # Loading the model to the simulation
model_orig.solver = 'glpk'
```

Set parameter Username

Academic license - for non-commercial use only - expires 2022-10-10

Shadow Prices

Shadow prices are calculated with the `model.optimize()` method.

```
In [3]: model = model_orig.copy()
solution = model.optimize()
solution.shadow_prices.round(5)
```

Shadow_prices_Reduced_costs.ipynb



Lesson Outline

- Robustness Analysis
- Shadow Prices
- • Reduced Costs
- Phenotype Phase Plane Analysis
- Production Envelopes

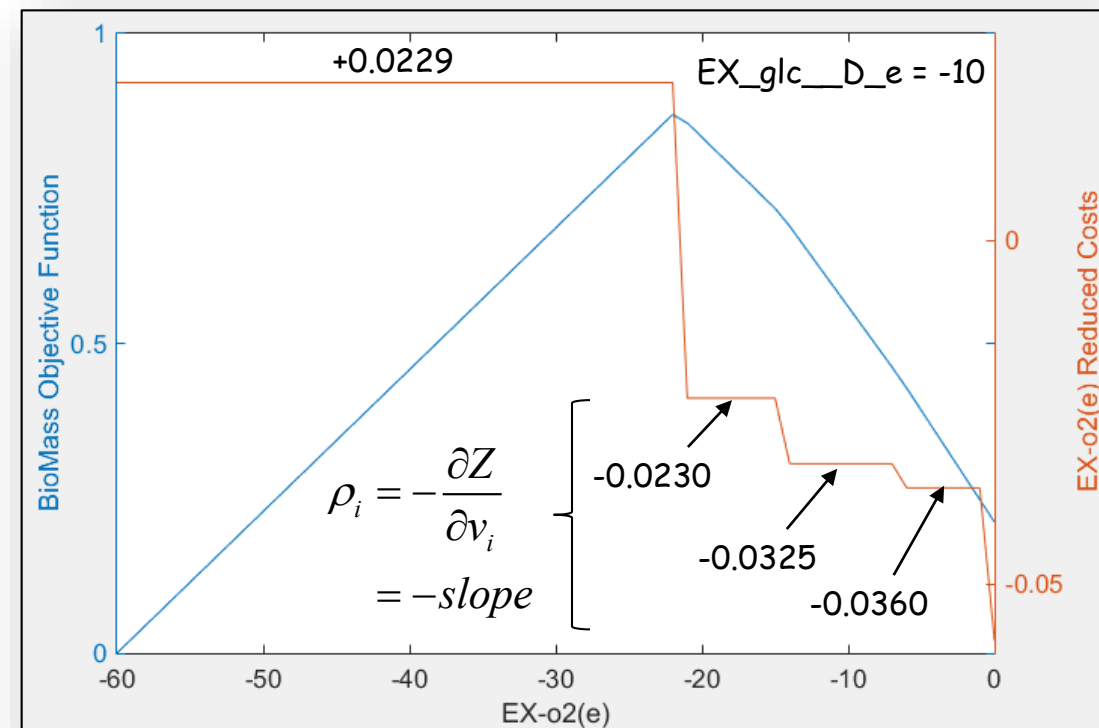
Reduced Costs: Reactions

- Reduced costs, ρ_i , are the derivatives of the objective function (Z) with respect to a reaction (v_i). Reduced costs indicate how much each particular **reaction** affects the objective.

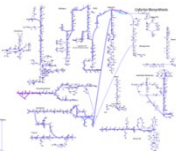
$$\rho_i = -\frac{\partial Z}{\partial v_i} \Rightarrow \rho_i \approx -\frac{\Delta Z}{\Delta v} \Rightarrow \Delta Z \approx -\rho_i \Delta v \Rightarrow \Delta Z \approx -\rho_i$$

1 unit of flux

- The reduced costs are associated with each flux (v_i) and signify the amount by which the objective function is decreased if v_i is increased. For instance, if the input flux of glucose shows a reduced cost of $-x$, it means that increasing that flux by one unit will increase of the objective function by x units.
- In the COBRA Toolbox, reduced costs can be calculated by `model.optimize()`. They can be accessed by `solution.reduced_costs` or `model.reactions.get_by_id('y').reduced_cost` where 'y' is the reaction name.
- Some solvers return positive reduced costs!



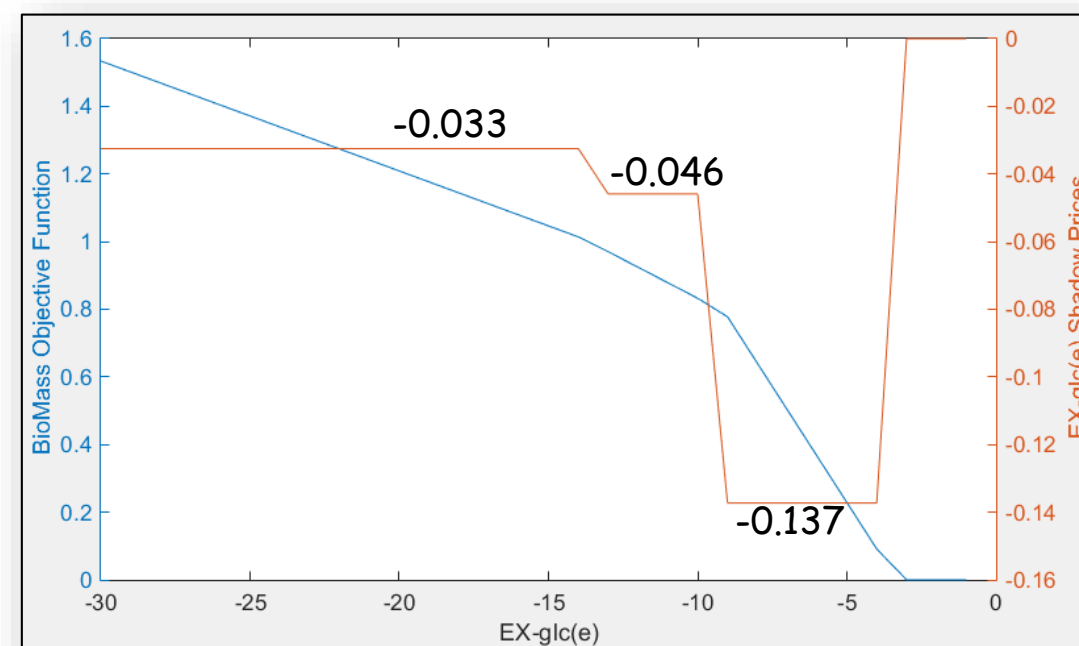
Shadow_prices_Reduced_costs.ipynb



Glucose Reduced Costs: Aerobic Growth Rate Example

(Shadow_prices_Reduced_costs.ipynb)

If the objective is set to maximize cell growth rate (Z_{GR}), and the reduced costs ($\rho_{EX_glc_D_e}$), of glucose ($EX_glc_D_e$) is -0.046, it means that an additional unit of glucose will increase the growth rate by 0.046.



Biomass ($EX_glc_D_e = -10$)

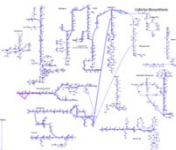
0. 832

Biomass ($EX_glc_D_e = -11$)

0. 878

Δ Biomass

0.046



Shadow Prices and Reduced Costs

```
In [1]: import cobra.test
import numpy as np
import pandas as pd
from cobrapy_bigg_client import client
import matplotlib.pyplot as plt
import escher
from escher import Builder
pd.set_option('display.max_rows', 500)
```

```
In [2]: model_orig = client.download_model('e_coli_core', save=False) # Loading the model to the simulation
model_orig.solver = 'glpk'
```

Set parameter Username

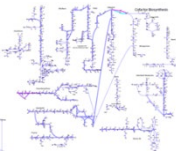
Academic license - for non-commercial use only - expires 2022-10-10

Shadow Prices

Shadow prices are calculated with the `model.optimize()` method.

```
In [3]: model = model_orig.copy()
solution = model.optimize()
solution.shadow_prices.round(5)
```

Shadow_prices_Reduced_costs.ipynb

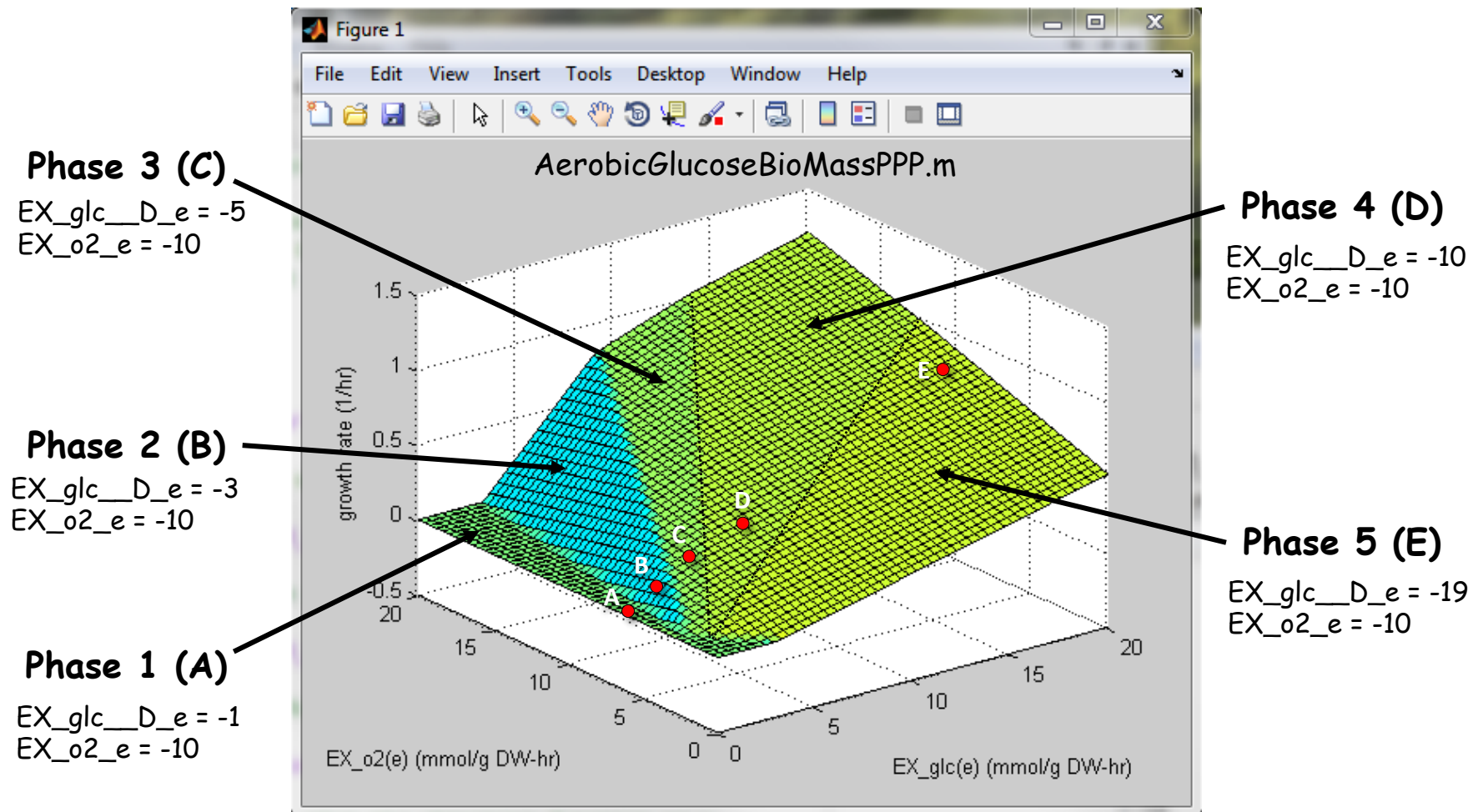


Lesson Outline

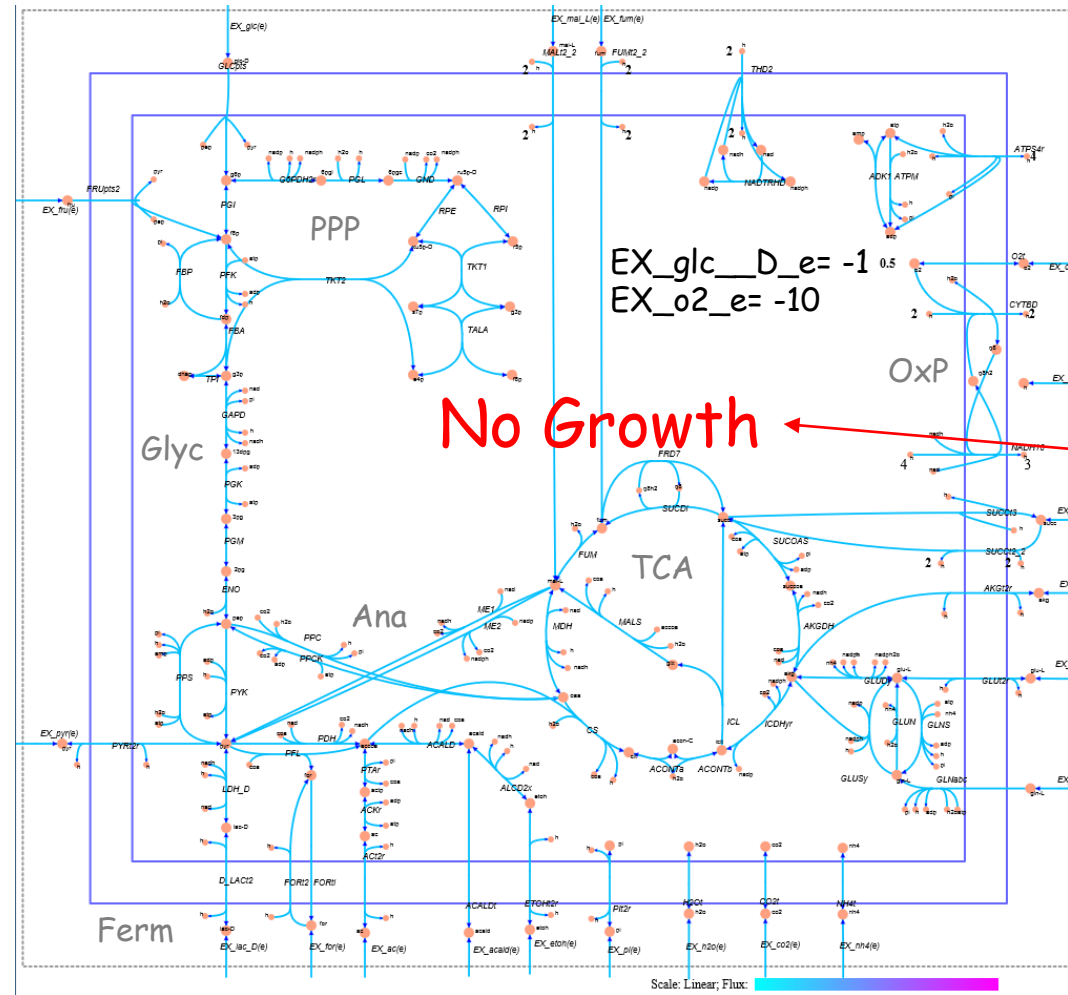
- Robustness Analysis
- Shadow Prices
- Reduced Costs
- • Phenotype Phase Plane Analysis
- Production Envelopes

Phenotype Phase Plane Analysis Example

Variables: EX_{o2_e} & $EX_{glc_D_e}$



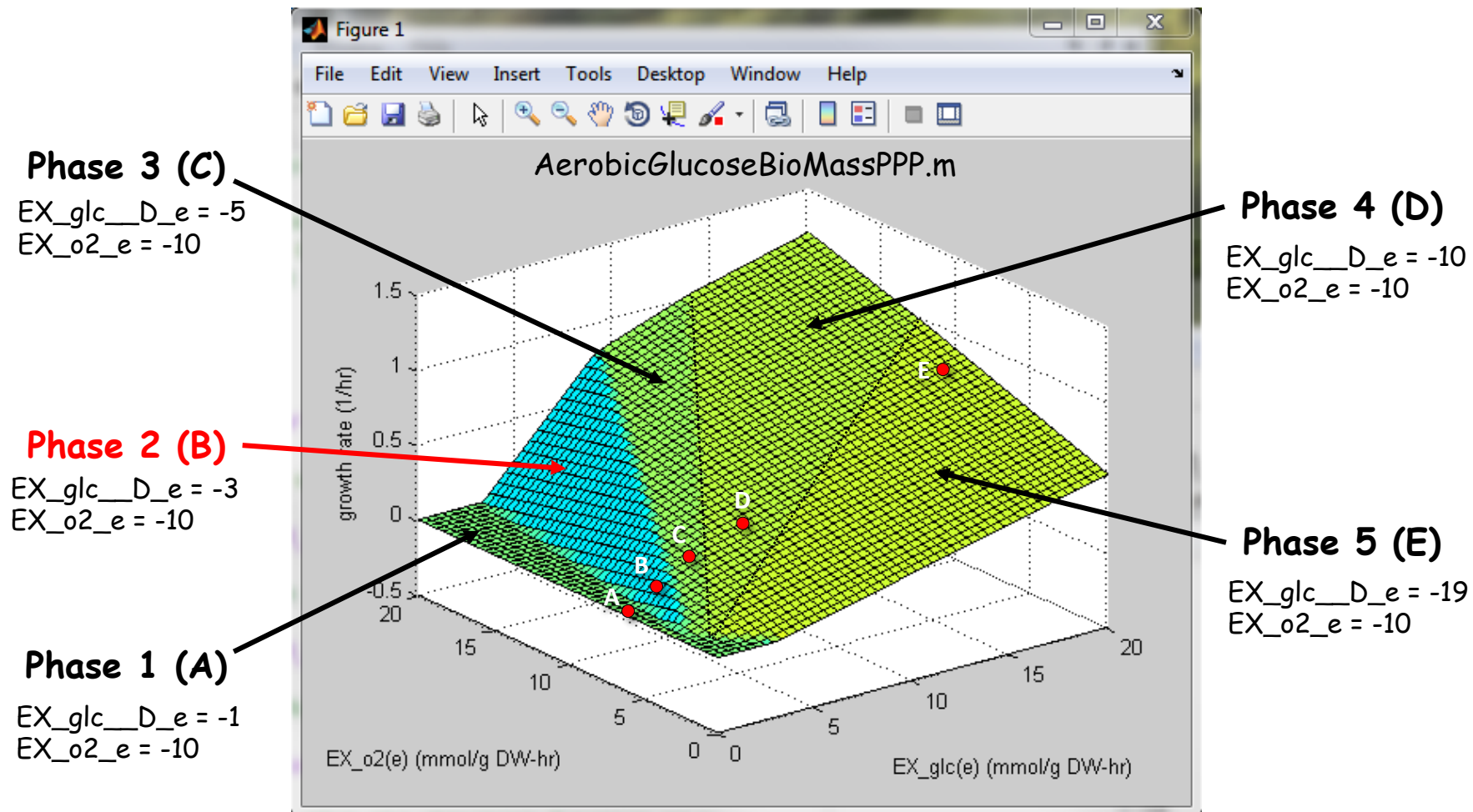
Phenotype Phase Plane Analysis Example (Phase 1)



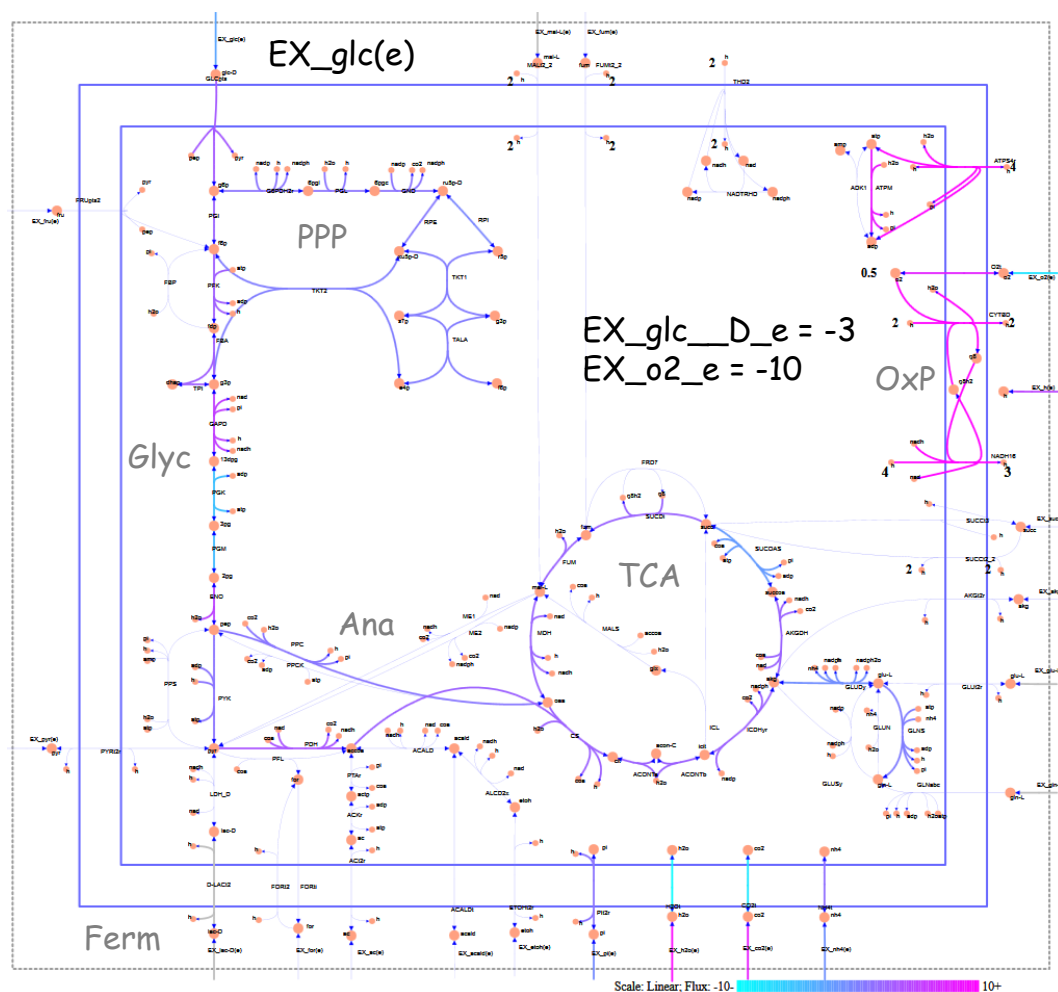
Phenotype_phase_plan_example.ipynb

Phenotype Phase Plane Analysis Example

Variables: EX_{o2_e} & $EX_{glc_D_e}$



Phenotype Phase Plane Analysis Example (Phase 2)



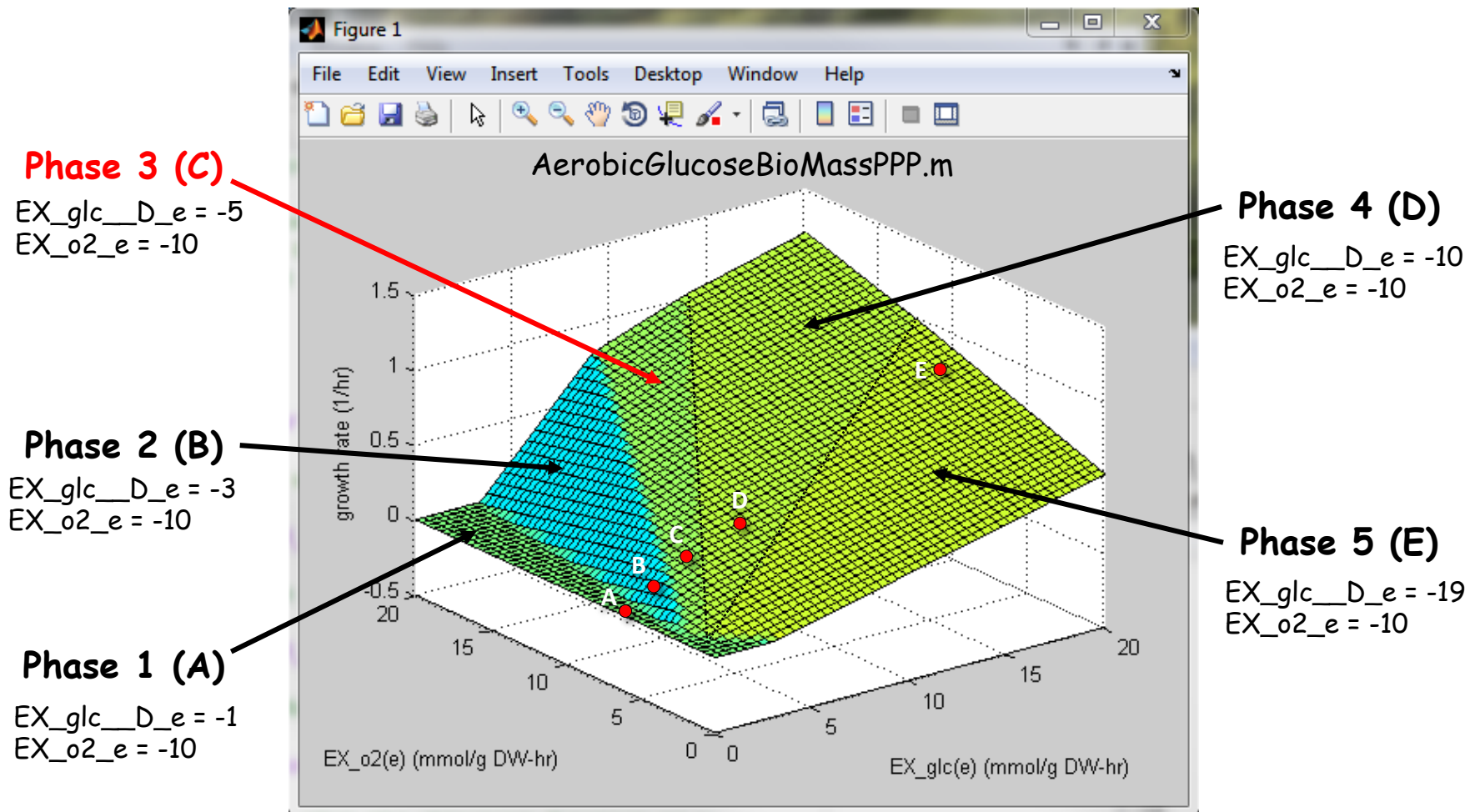
Phenotype_phase_plan_example.ipynb

ACONTa	3.41586	FBA	2.81795	PGI	2.96248
ACONTb	3.41586	FUM	3.2184	PGK	-5.51348
AKGDH	2.69395	GAPD	5.51348	PGM	-5.23969
ATPM	8.39	GLCpts	3	PI+2r	0.673268
ATPS4r	21.278	GLNS	9.47587	PYK	2.14468
Biomass	0.183018	GLUDy	-0.951162	RPE	-0.131553
CO2+	-10.2116	GLUN	9.42907	RPI	-0.131553
CS	3.41586	H2O+	-11.5448	SUCDi	3.2184
CYTBD	20	ICDHyr	2.89141	SUCOAS	-2.69395
ENO	5.23969	ICL	0.524456	TALA	-0.032741
EX_co2(e)	10.2116	MALS	0.524456	THD2	0.444093
EX_glc(e)	-3	MDH	3.74286	TKT1	-0.032741
EX_h2o(e)	11.5448	NADH16	16.7816	TKT2	-0.098811
EX_h(e)	3.67134	NH4+	0.997959	TPI	2.81795
EX_nh4(e)	-0.997959	O2+	10		
EX_o2(e)	-10	PDH	4.62623		
EX_pi(e)	-0.673268	PFK	2.81795		

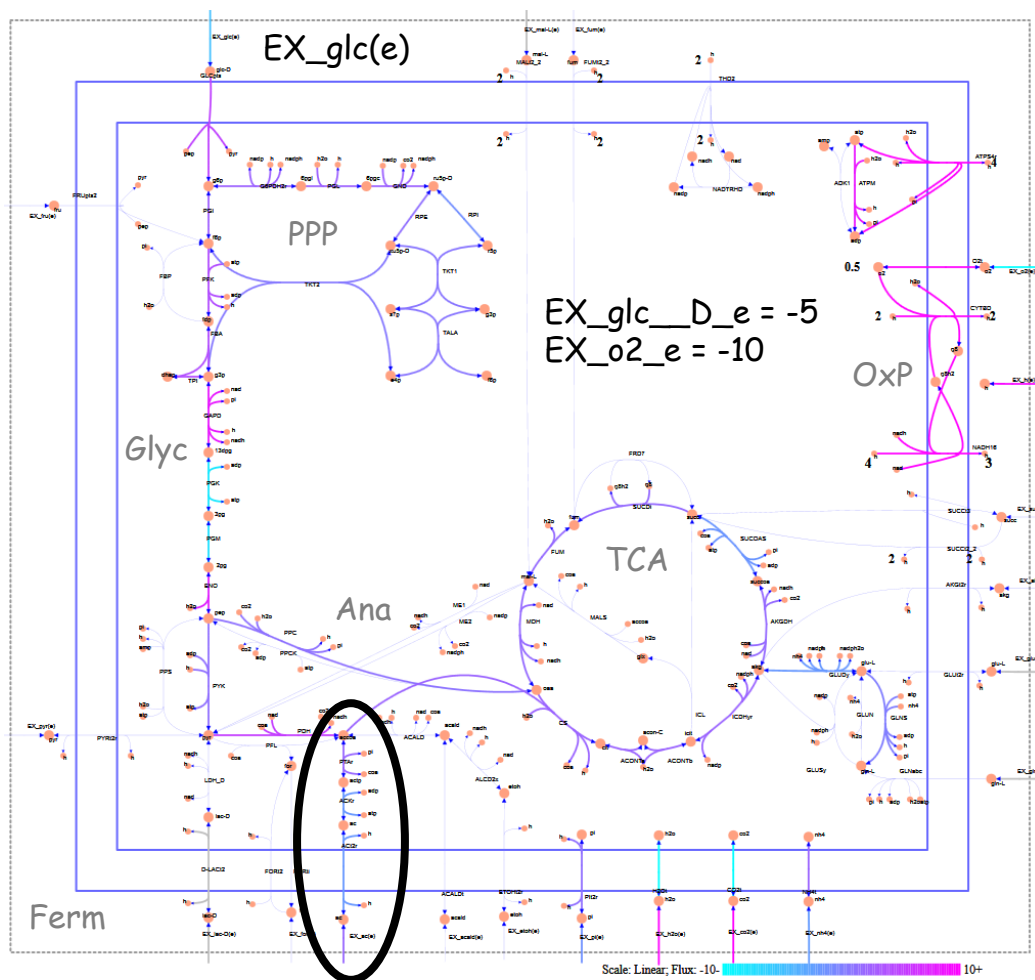
Growth is limited by excess oxygen; not enough glucose to reduce all the oxygen and produce biomass optimally

Phenotype Phase Plane Analysis Example

Variables: EX_{o2_e} & $EX_{glc_D_e}$



Phenotype Phase Plane Analysis Example (Phase 3)



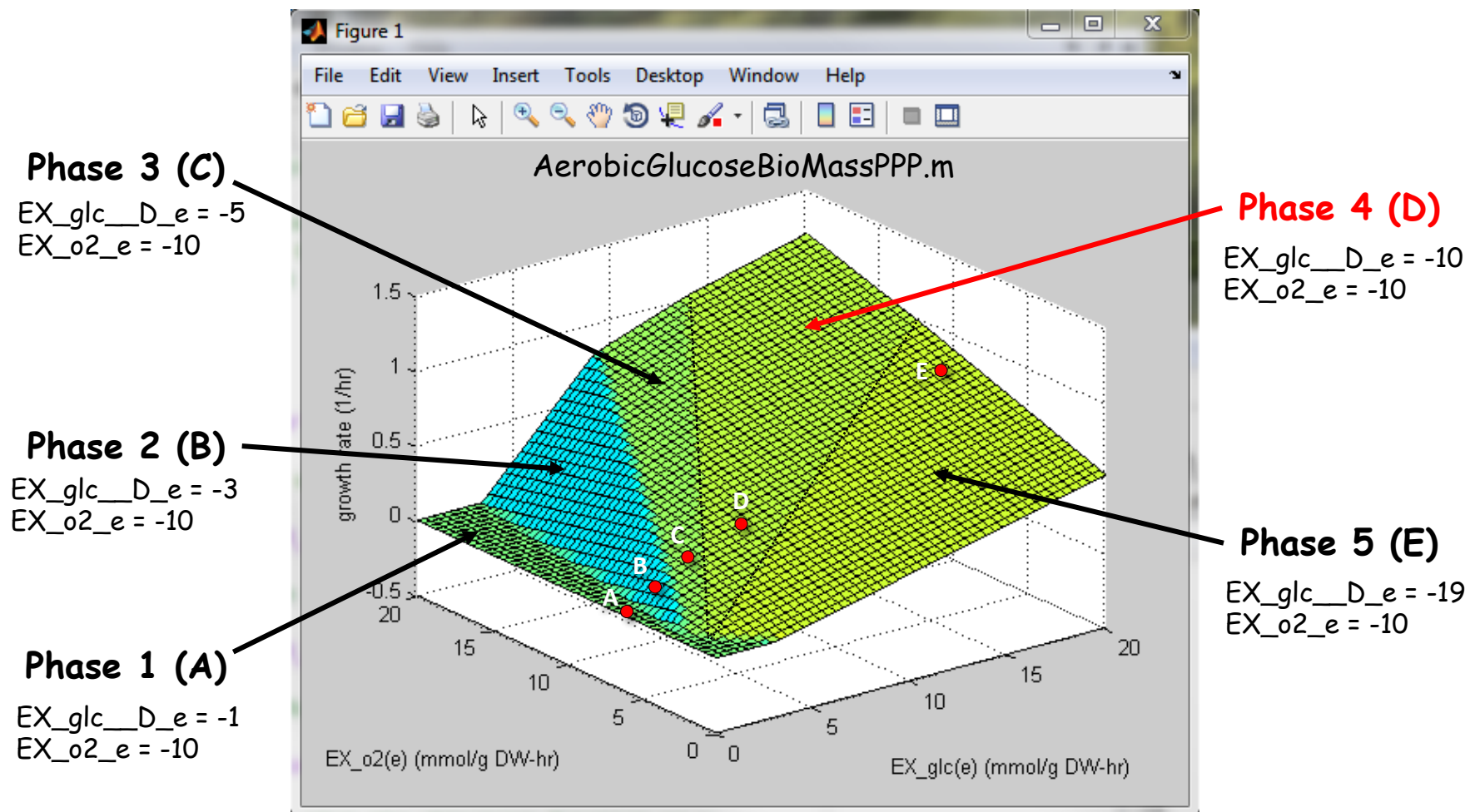
Phenotype_phase_plan_example.ipynb

ACKr	-1.83668	EX_nh4(e)	-2.03666	PDH	5.34252
ACONTa	2.106	EX_o2(e)	-10	PFK	3.84494
ACONTb	2.106	EX_pi(e)	-1.37402	PGI	2.57284
ACT2r	-1.83668	FBA	3.84494	PGK	-8.22357
AKGDH	1.70302	FUM	1.70302	PGL	2.35059
ATPM	8.39	G6PDH2r	2.35059	PGM	-7.66481
ATPS4r	21.5061	GAPD	8.22357	PI+2r	1.37402
Biomass	0.373508	GLCpts	5	PPC	1.07032
CO2†	-10.4318	GLNS	0.095506	PTAr	1.83668
CS	2.106	GLUDy	-1.94116	PYK	1.40059
CYTBD	20	GND	2.35059	RPE	1.29858
ENO	7.66481	H2O†	-13.1526	RPI	-1.05201
EX_ac(e)	1.83668	ICDHyr	2.106	SUCDi	1.70302
EX_co2(e)	10.4318	MDH	1.70302	SUCOAS	-1.70302
EX_glc(e)	-5	NADH16	18.297	TALA	0.71671
EX_h2o(e)	13.1526	NH4†	2.03666	TKT1	0.71671
EX_h(e)	9.32925	O2†	10	TKT2	0.581873
				TPI	3.84494

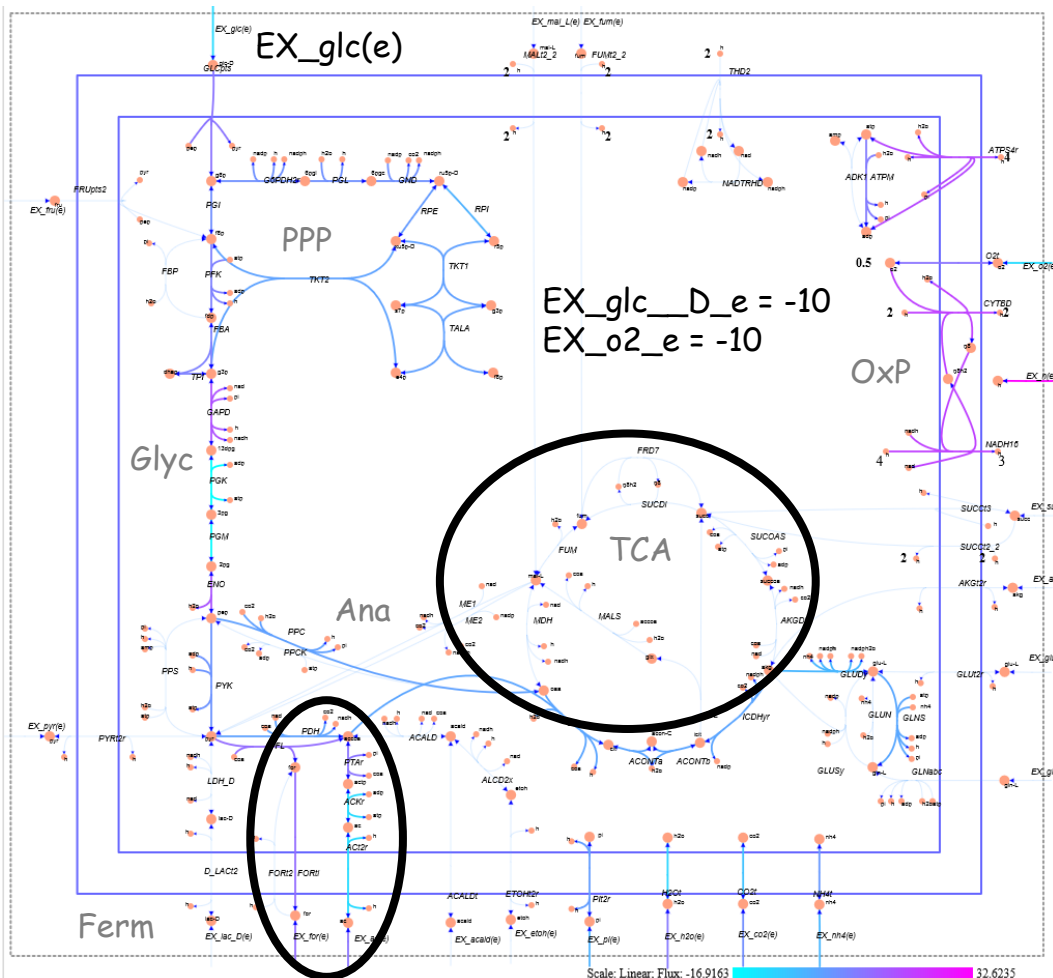
Not enough oxygen to fully oxidize glucose;
acetate produced through fermentation

Phenotype Phase Plane Analysis Example

Variables: EX_{o2_e} & $EX_{glc_D_e}$



Phenotype Phase Plane Analysis Example (Phase 4)



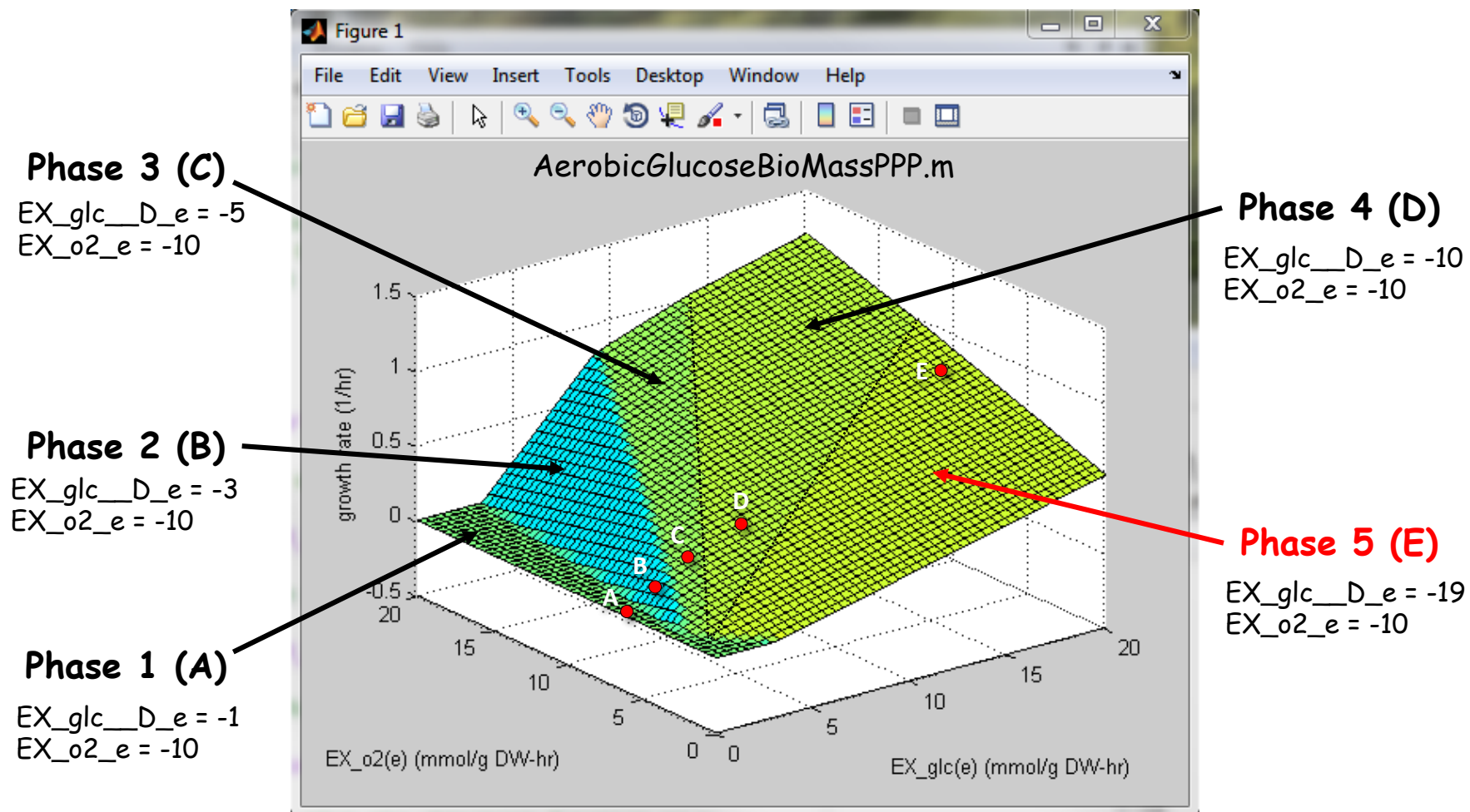
Phenotype_phase_plan_example.ipynb

ACKr	-9.90568	EX_o2(e)	-10	PGI	5.09263
ACONTa	0.60316	EX_pi(e)	-2.05658	PGK	-16.9163
ACONTb	0.60316	FBA	7.84632	PGL	4.79277
ACT2r	-9.90568	FORti	11.5033	PGM	-16.0799
ATPM	8.39	G6PDH2r	4.79277	PIt2r	2.05658
ATPS4r	18.8064	GAPD	16.9163	PPC	1.60202
Biomass	0.559051	GLCpts	10	PTAr	9.90568
CO2t	-4.89467	GLNS	0.142949	PYK	4.18773
CS	0.60316	GLUDy	-2.90544	RPE	2.79333
CYTBD	20	GND	4.79277	RPI	-1.99944
ENO	16.0799	H2O+	-8.96702	TALA	1.49758
EX_ac(e)	9.90568	ICDHyr	0.60316	TKT1	1.49758
EX_co2(e)	4.89467	NADH16	20	TKT2	1.29576
EX_for(e)	11.5033	NH4+	3.04839	TPI	7.84632
EX_glc(e)	-10	O2+	10		
EX_h2o(e)	8.96702	PDH	1.10076		
EX_h(e)	32.6235	PFK	7.84632		
EX_nh4(e)	-3.04839	PFL	11.5033		

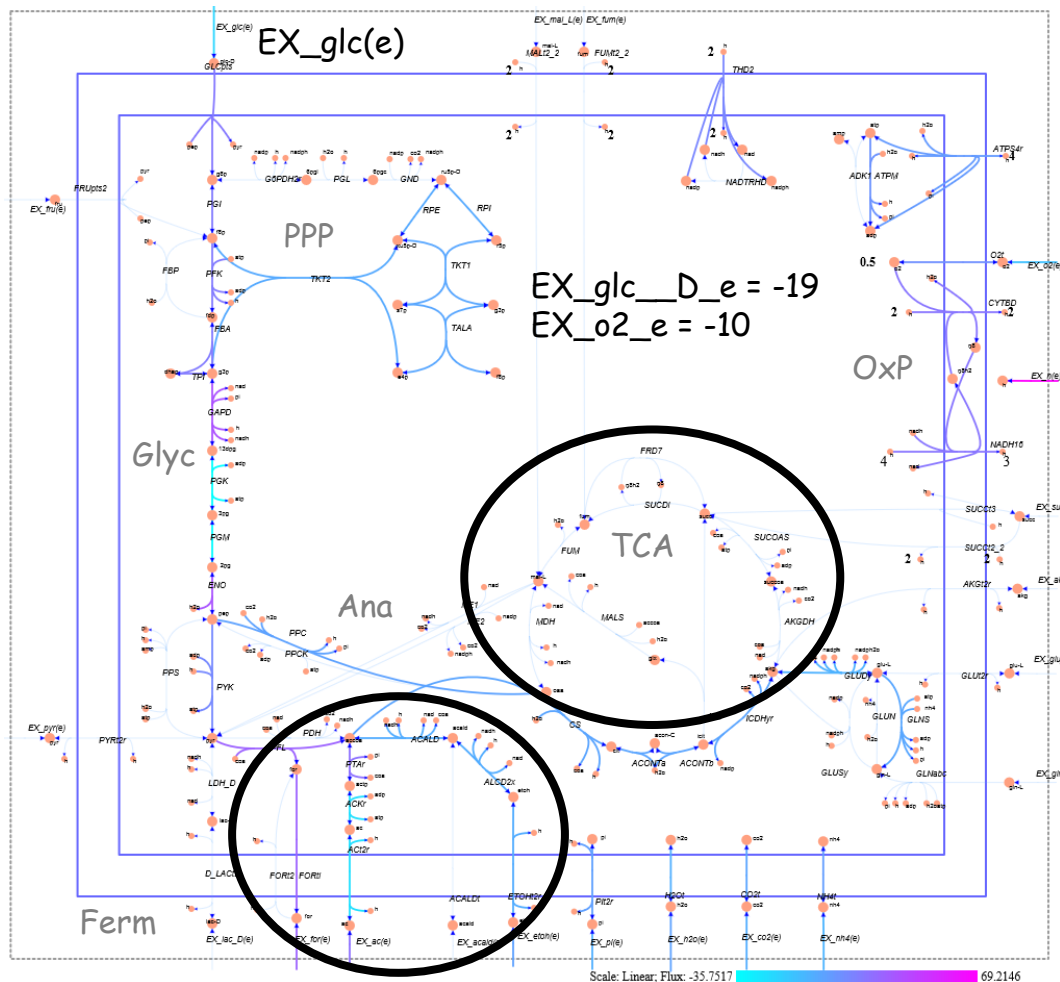
Not enough oxygen to fully oxidize glucose;
acetate and formate are produced and secreted

Phenotype Phase Plane Analysis Example

Variables: EX_{o2_e} & $EX_{glc_D_e}$



Phenotype Phase Plane Analysis Example (Phase 5)



Phenotype_phase_plan_example.ipynb

ACALD	-2.12502	EX_glc(e)	-19	PFL	29.2279
ACKr	-23.0206	EX_h2o(e)	2.52474	PGI	18.8266
ACONTa	0.912497	EX_h(e)	69.2146	PGK	-35.7517
ACONTb	0.912497	EX_nh4(e)	-4.61179	PGM	-34.4864
ACt2r	-23.0206	EX_o2(e)	-10	PIt2r	3.11132
ALCD2x	-2.12502	EX_pi(e)	-3.11132	PPC	2.42363
ATPM	8.39	FBA	18.1587	PTAr	23.0206
ATPS4r	5.95415	FORti	29.2279	PYK	12.6238
Biomass	0.845766	GAPD	35.7517	RPE	-0.607936
CO2t	1.51113	GLCpts	19	RPI	-0.607936
CS	0.912497	GLNS	0.216262	TALA	-0.151307
CYTBD	20	GLUDy	-4.39553	THD2	14.5016
ENO	34.4864	H2O+	-2.52474	TKT1	-0.151307
ETOHt2r	-2.12502	ICDHyr	0.912497	TKT2	-0.456629
EX_ac(e)	23.0206	NADH16	20	TPI	18.1587
EX_co2(e)	-1.51113	NH4+	4.61179		
EX_etoh(e)	2.12502	O2+	10		
EX_for(e)	29.2279	PFK	18.1587		

Not enough oxygen to fully oxidize glucose; acetate, formate and ethanol are produced and secreted.

Phenotype Phase Plane Analysis Example

Variables: EX_o2(e) & EX_glc(e)

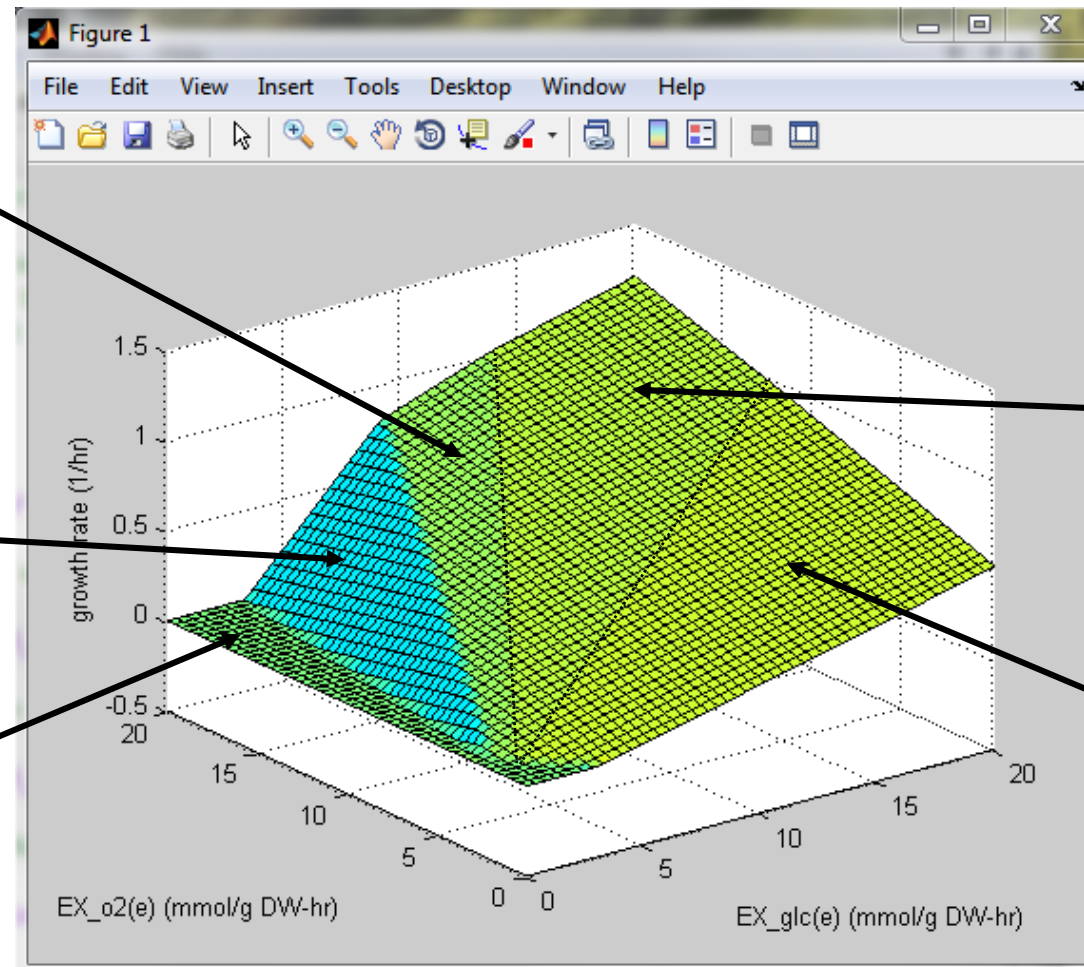
Phase 3
Not enough oxygen to fully oxidize glucose; acetate produced through fermentation

Phase 2
Growth is limited by excess oxygen; not enough glucose to reduce all the oxygen and produce biomass optimally

Phase 1
No growth; not enough glucose

Phase 4
Not enough oxygen to fully oxidize glucose; acetate and formate are produced and secreted

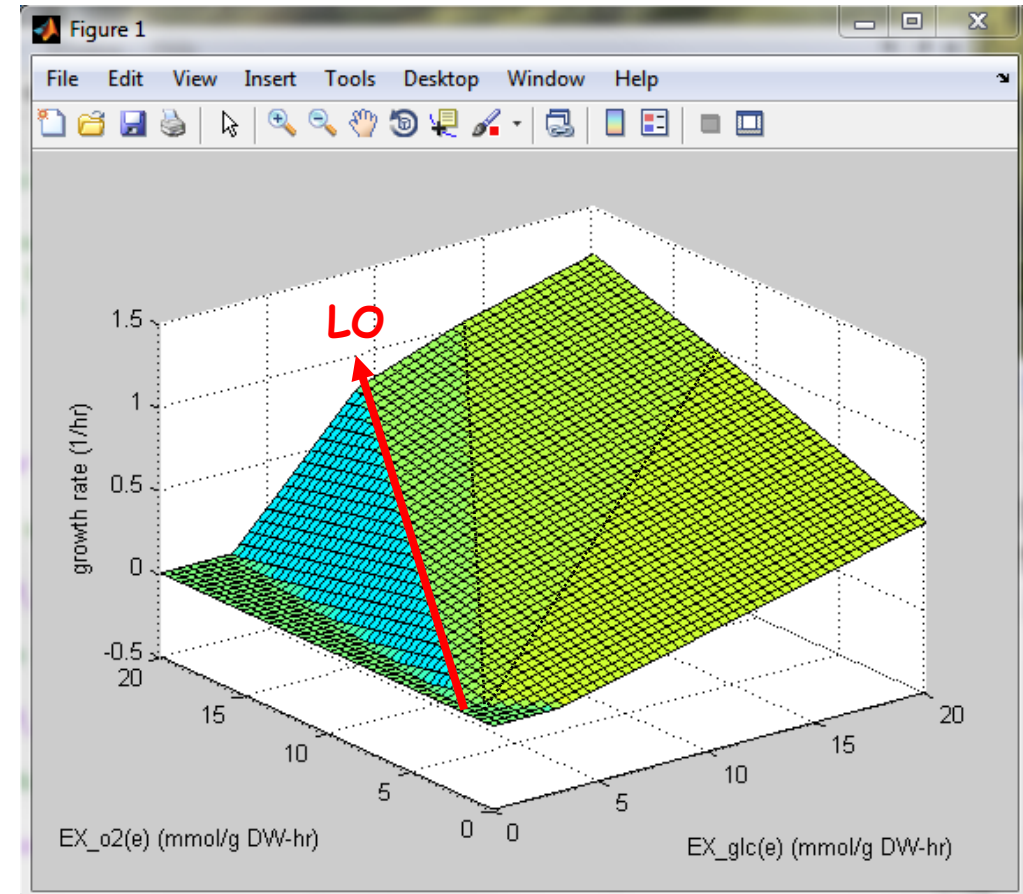
Phase 5
Not enough oxygen to fully oxidize glucose; acetate, formate and ethanol are produced and secreted.



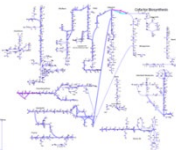
Orth, J. D., I. Thiele, et al. (2010). "What is flux balance analysis?" Supplementary Tutorial, Nature biotechnology 28(3): 245-248. (Supplementary Examples 4 & 5)

Line of Optimality

- The line of optimality (LO) is defined as a line representing the optimal relation between the two metabolic fluxes used to create a phenotype phase plane.
- The line of optimality is determined by specifying an uptake rate of the substrate along the x-axis and then allowing any value for the flux along the y-axis. Linear Programming can then be used to calculate the optimal value of the objective as a function of the y-axis flux. Once the objective is determined, the corresponding flux value for the y-axis is used to plot the line of optimality (LO).
- The LO defines the optimal utilization of the metabolic pathways without limitations on the availability of the substrates.



Edwards, J. S., R. U. Ibarra, et al. (2001). "In silico predictions of Escherichia coli metabolic capabilities are consistent with experimental data." Nat Biotechnol 19(2): 125-130.



Phenotype Phase Plane Analysis Example

```
In [1]: import cobra.test
import numpy as np
import pandas as pd
from cobrapy_bigg_client import client
import matplotlib.pyplot as plt
import escher
from escher import Builder
pd.set_option('display.max_rows', 500)
```

```
In [2]: model_orig = client.download_model('e_coli_core', save=False) # Loading the model to the simulation
model_orig.solver = 'glpk'
```

Set parameter Username

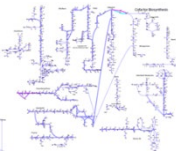
Academic license - for non-commercial use only - expires 2022-10-10

Phase 1 (EX_glc__D_e = -1; EX_o2_e = -10)

```
In [3]: model_1 = model_orig.copy()

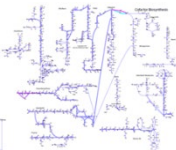
model_1.reactions.EX_glc__D_e.bounds = [-1,-1]
model_1.reactions.EX_o2_e.bounds = [-10,-10]
solution1 = model_1.optimize()
solution1_frame = solution1.to_frame()
solution1_frame.sort_index().round(5)
solution1_frame.fluxes.round(5)
```

Phenotype_phase_plane_example.ipynb




Lesson Outline

- Robustness Analysis
- Shadow Prices
- Reduced Costs
- Phenotype Phase Plane Analysis
- • Production Envelopes

cameo Watch

23

build error

 codecov 64%

Navigation

[Installation](#)[Tutorials](#)[Contributing](#)[Development](#)[API Docs](#)

Welcome to cameo!

chat on gitter


pypi v0.13.6

license APACHE2

build error

coverage 67%

DOI 10.5281/zenodo.2575046

Shipping faster with ZenHub  launch binder

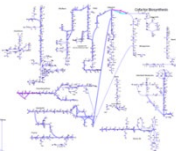
What is cameo?

Cameo is a high-level python library developed to aid the strain design process in metabolic engineering projects. The library provides a modular framework of simulation and strain design methods that targets developers that want to develop new design algorithms and custom analysis workflows. Furthermore, it exposes a high-level API to users that just want to compute promising strain designs.

Curious? Head over to try.cameo.bio and give it a try.

Please cite <https://doi.org/10.1021/acssynbio.7b00423> if you've used cameo in a scientific publication.

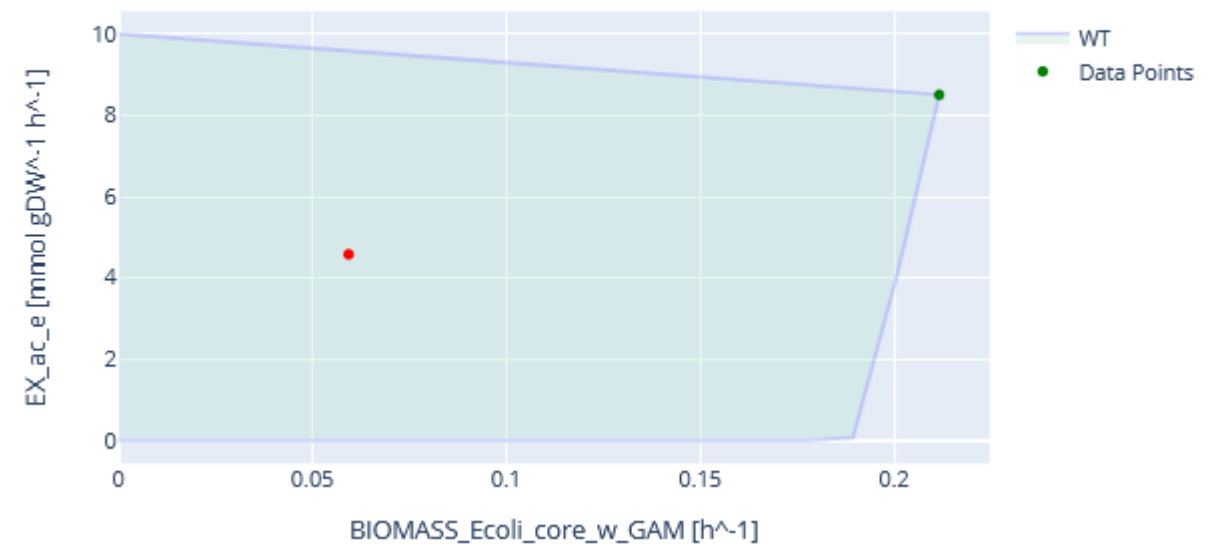
<https://cameo.bio/index.html>

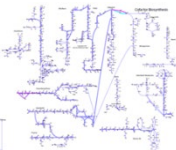


Production Envelopes

- Production envelopes are part of phenotypic phase plane analysis which is a modeling technique that was developed to do a theoretical assessment of the limits of cell operation based on their stoichiometric constraints [Edwards *et al.* 2001].
- A production envelope is a representation that illustrates the tradeoffs between production of the desired product and cell growth.

Phenotypic Phase Plane (flux)



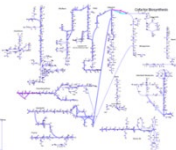


Production Envelope Data

BIOMASS_Ecoli_core_w_GAM	objective_lower_bound	objective_upper_bound	c_yield_lower_bound	c_yield_upper_bound	mass_yield_lower_bound	mass_yield_upper_bound
0	0	10	0	0.333333	0	0.327739
0.01114	0	9.921241	0	0.330708	0	0.325157
0.02228	0	9.842483	0	0.328083	0	0.322576
0.03342	0	9.763724	0	0.325457	0	0.319995
0.044561	0	9.684965	0	0.322832	0	0.317414
0.055701	0	9.606207	0	0.320207	0	0.314832
0.066841	0	9.527448	0	0.317582	0	0.312251
0.077981	0	9.448689	0	0.314956	0	0.30967
0.089121	0	9.369931	0	0.312331	0	0.307089
0.100261	0	9.291172	0	0.309706	0	0.304507
0.111402	0	9.212413	0	0.30708	0	0.301926
0.122542	0	9.133655	0	0.304455	0	0.299345
0.133682	0	9.054896	0	0.30183	0	0.296764
0.144822	0	8.976137	0	0.299205	0	0.294183
0.155962	0	8.897379	0	0.296579	0	0.291601
0.167102	0	8.81862	0	0.293954	0	0.28902
0.178242	0	8.739861	0	0.291329	0	0.286439
0.189383	0.073591	8.661103	0.002453	0.288703	0.002412	0.283858
0.200523	3.986155	8.582344	0.132872	0.286078	0.130642	0.281276
0.211663	8.503585	8.503585	0.283453	0.283453	0.278695	0.278695

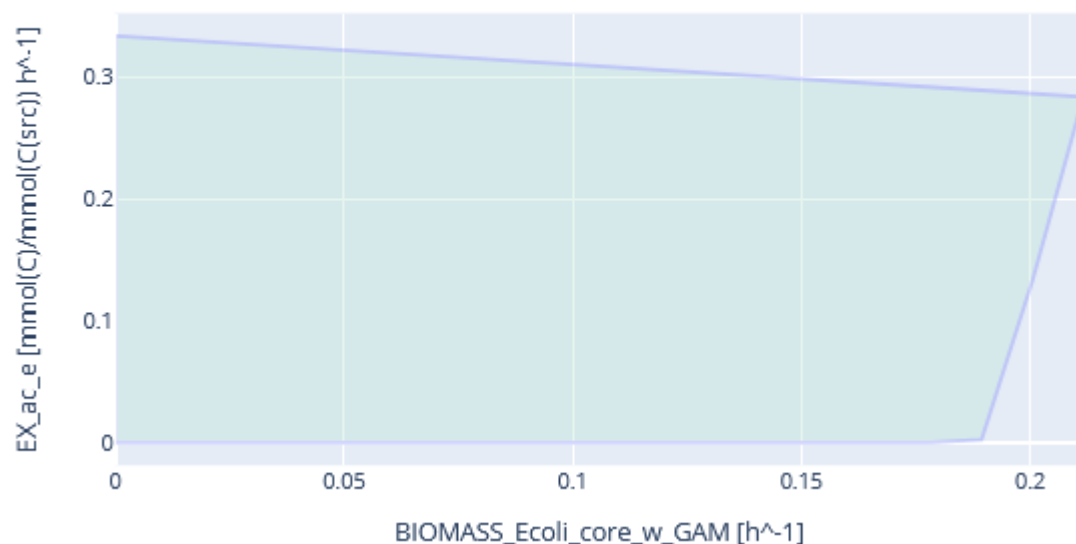
carbon_yield: if carbon source is defined and the product is a single metabolite (mol carbon product per mol carbon feeding source)

mass_yield: if carbon source is defined and the product is a single metabolite (gram product per 1 g of feeding source)

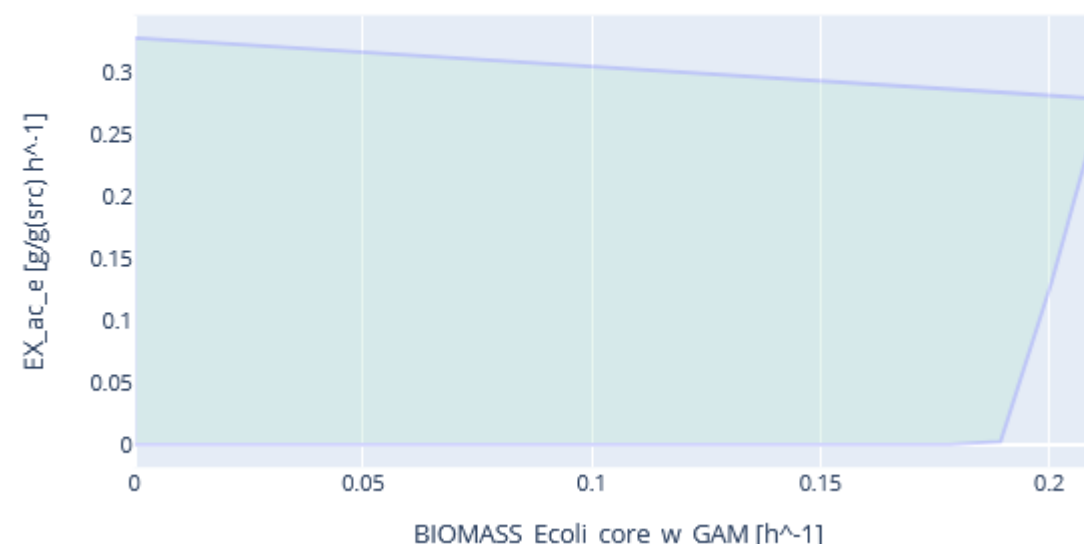


Production Envelope Categories

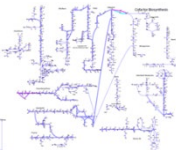
Phenotypic Phase Plane (carbon yield, src=EX_glc_D_e)



Phenotypic Phase Plane (mass yield, src=EX_glc_D_e)



Can plot any of the categories calculated by this method: objective_lower_bound, objective_upper_bound, c_yield_lower_bound, c_yield_upper_bound, mass_yield_lower_bound, and mass_yield_upper_bound.



Production Envelopes

Production envelopes are part of phenotypic phase plane analysis which is a modeling technique that was developed to do a theoretical accessment of the limits of cell operation based on their stoichiometric constraints [Edwards *et al.* 2001].

A production envelope is a representation that illustrates the tradeoffs between production of the desired product and cell growth.

This tutorial is a modified version of the Cameo tutorial on phenotype phase planes at <https://cameo.bio/04-analyze-models.html#phenotypic-phase-plane>

Setting the environment

```
In [1]: from cameo import models
        from cameo.visualization.plotting.with_plotly import PlotlyPlotter
        #from cameo import flux_variability_analysis
        from cameo import phenotypic_phase_plane

        model = models.bigg.e_coli_core.copy()
        plotter = PlotlyPlotter()

        Set parameter Username
        Academic license - for non-commercial use only - expires 2022-10-10
        Read LP format model from file C:\Users\hinton\AppData\Local\Temp\tmpg4bnmixz.lp
        Reading time = 0.01 seconds
        : 72 rows, 190 columns, 720 nonzeros
```

Production_envelopes.ipynb



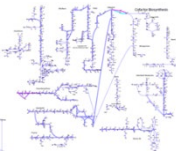
Lesson Outline

- Robustness Analysis
- Shadow Prices
- Reduced Costs
- Phenotype Phase Plane Analysis
- Production Envelopes



Reflective Questions

1. What is the purpose of robustness analysis?
2. What do the kinks in the robustness analysis represent?
3. What are shadow prices?
4. How are shadow prices related to the objective function?
5. What are reduced costs?
6. How are reduced costs related to the objective function?
7. What is the difference between shadow prices and reduced costs?
8. What should you use to predict the impact of a metabolite on the objective function, shadow prices or reduced costs?
9. What should you use to predict the impact of a reaction on the objective function, shadow prices or reduced costs?
10. What is the purpose of phenotype phase plane analysis?
11. How is robustness analysis related to phenotype phase plane analysis?
12. What is the relationship between shadow prices and phenotype phase plane analysis?
13. What is the line of optimality?
14. How do the different phases created in the phenotype phase plane analysis connect to the physiology of a cell?
15. What is Cameo?
16. What is the purpose of a production envelope?
17. What does carbon yield mean?
18. What does mass yield mean?



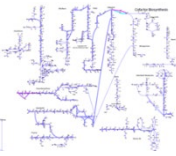
References

Robustness Analysis

- [Edwards, J. S. and B. O. Palsson \(2000\). "Robustness analysis of the Escherichia coli metabolic network." Biotechnology progress 16\(6\): 927-939.](#)
- [Edwards, J. S., R. U. Ibarra, et al. \(2001\). "In silico predictions of Escherichia coli metabolic capabilities are consistent with experimental data." Nat Biotechnol 19\(2\): 125-130.](#)
- [Orth, J. D., I. Thiele, et al. \(2010\). "What is flux balance analysis?" Nature biotechnology 28\(3\): 245-248. \(Supplementary Tutorial\)](#)
- [Price, N. D., J. A. Papin, et al. \(2003\). "Genome-scale microbial in silico models: the constraints-based approach." Trends in biotechnology 21\(4\): 162-169](#)

Phenotype Phase Plane Analysis

- [Bell, S. L. and B. O. Palsson \(2005\). "Phenotype phase plane analysis using interior point methods." Computers & Chemical Engineering 29\(3\): 481-486.](#)
- [Edwards, J. S., R. Ramakrishna, et al. \(2002\). "Characterizing the metabolic phenotype: a phenotype phase plane analysis." Biotechnology and bioengineering 77\(1\): 27-36.](#)
- [Ibarra, R. U., J. S. Edwards, et al. \(2002\). "Escherichia coli K-12 undergoes adaptive evolution to achieve in silico predicted optimal growth." Nature 420\(6912\): 186-189](#)
- [Kauffman, K. J., J. D. Pajerowski, et al. \(2002\). "Description and analysis of metabolic connectivity and dynamics in the human red blood cell." Biophysical journal 83\(2\): 646-662.](#)
- [Edwards, J. S., R. U. Ibarra, et al. \(2001\). "In silico predictions of Escherichia coli metabolic capabilities are consistent with experimental data." Nat Biotechnol 19\(2\): 125-130.](#)
- [Edwards, J. S. and B. O. Palsson \(2000\). "Robustness analysis of the Escherichia coli metabolic network." Biotechnology progress 16\(6\): 927-939.](#)
- [Orth, J. D., I. Thiele, et al. \(2010\). "What is flux balance analysis?" Nature biotechnology 28\(3\): 245-248. \(Supplementary Tutorial\)](#)
- [Price, N. D., J. A. Papin, et al. \(2003\). "Genome-scale microbial in silico models: the constraints-based approach." Trends in biotechnology 21\(4\): 162-169](#)



COBRApy Production Envelopes

The production envelope can be used to analyze a model's ability to produce a given compound conditional on the fluxes for another set of reactions, such as the uptake rates. The model is alternately optimized with respect to minimizing and maximizing the objective and the obtained fluxes are recorded. Ranges to compute production are set to the effective bounds, i.e., the minimum / maximum fluxes that can be obtained given current reaction bounds.

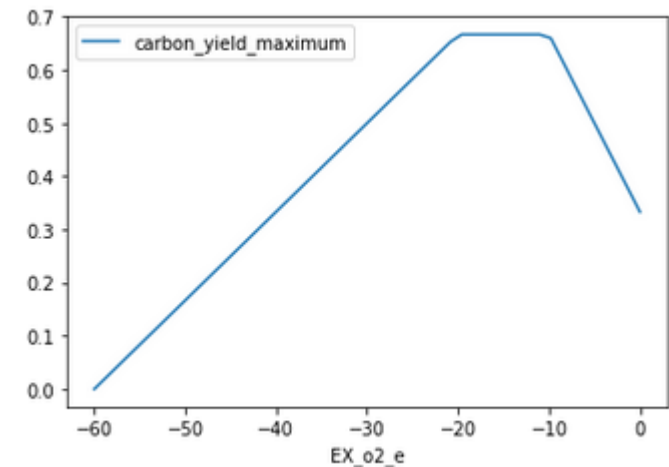
Parameters:

- **model** (cobra.Model) - The model to compute the production envelope for.
- **reactions** (list or string) - A list of reactions, reaction identifiers or a single reaction.
- **objective** (string, dict, model.solver.interface.Objective, optional) - The objective (reaction) to use for the production envelope. Use the model's current objective if left missing.
- **carbon_sources** (list or string, optional) - One or more reactions or reaction identifiers that are the source of carbon for computing carbon (mol carbon in output over mol carbon in input) and mass yield (gram product over gram output). Only objectives with a carbon containing input and output metabolite is supported. Will identify active carbon sources in the medium if none are specified.
- **points** (int, optional) - The number of points to calculate production for.
- **threshold** (float, optional) - A cut-off under which flux values will be considered to be zero (default model.tolerance).

Returns: A data frame with one row per evaluated point and

- **reaction id** : one column per input reaction indicating the flux at each given point,
- **carbon_source**: identifiers of carbon exchange reactions A column for the maximum and minimum each for the following types:
 - **flux**: the objective flux
 - **carbon_yield**: if carbon source is defined and the product is a single metabolite (mol carbon product per mol carbon feeding source)
 - **mass_yield**: if carbon source is defined and the product is a single metabolite (gram product per 1 g of feeding source)

Return Type: pandas.DataFrame



https://cobrapy.readthedocs.io/en/stable/autoapi/cobra/flux_analysis/phenotype_phase_plane/index.html