Sensitivity of a flux balance analysis solution with respect to input data

Author(s): Ronan M.T. Fleming, Leiden University

Reviewer(s):

INTRODUCTION

Consider an FBA problem

$$\begin{aligned} \max \quad & c^T v \\ \text{s.t.} \quad & Sv = b \\ & l \leq v \leq u \end{aligned}$$

The local sensitivity of the optimal objective value $\mathscr{L}^\star = c^T v^\star$ with respect to a changes in the input data $\{b, l, u\}$ is given by

$$\frac{\partial \mathscr{L}^\star}{\partial b} = y^\star$$

$$\frac{\partial \mathscr{L}^\star}{\partial l} = -w_l^\star$$

$$\frac{\partial \mathscr{L}^\star}{\partial u} = w_u^\star$$

where $y^\star$ is a vector of shadow prices and $w = w_l - w_u$ is a vector of reduced costs. That is, a shadow price is the partial derivative of the optimal value of the objective function with respect to b_i. It indicates how much net production, or net consumption, of each metabolite increases (positive), or decreases (negative), the optimal value of the objective. The reduced costs, $-w_l$ and w_u are the partial derivative of the optimal value of the objective function with respect to the lower and upper bounds on a reaction, respectively. They indicate how much relaxation, or tightening, of each bound increases, or decreases, the optimal objective, respectively. In the COBRA Toolbox, shadow prices and reduced costs are calculated by `optimizeCbModel`. When using the function

```
FBAsolution = optimizeCbModel(model,'max');
```

the shadow prices and reduced costs are given by `FBAsolution.y` and `FBAsolution.w`, respectively.

For a more complete theoretical description, see: cobratoolbox/tutorials/intro_sensitivityAnalysis.pdf

MATERIALS - EQUIPMENT SETUP

Please ensure that all the required dependencies (e.g. , `git` and `curl`) of The COBRA Toolbox have been properly installed by following the installation guide here. Please ensure that the COBRA Toolbox has been initialised (tutorial_initialize.mlx) and verify that the pre-packaged LP and QP solvers are functional (tutorial_verify.mlx).

PROCEDURE

Load E. coli core model

The most direct way to load a model into The COBRA Toolbox is to use the `readCbModel` function. For example, to load a model from a MAT-file, you can simply use the filename (with or without file extension).

```
fileName = 'ecoli_core_model.mat';
if ~exist('modelOri','var')
modelOri = readCbModel(fileName);
end
%backward compatibility with primer requires relaxation of upper bound on
%ATPM
modelOri = changeRxnBounds(modelOri,'ATPM',1000,'u');
model = modelOri;
%setp the matlab e.coli metabolic map parameters
outputFormatOK = changeCbMapOutput('matlab');
map=readCbMap('ecoli_core_map');
options.zeroFluxWidth = 0.1;
options.rxnDirMultiplier = 10;
```

model

1x1 struct with 28 fields

Field	Value	Size
S	72x95 sparse do...	72x95
mets	72x1 cell	72x1
b	72x1 double	72x1
csense	72x1 char	72x1
rxns	95x1 cell	95x1
lb	95x1 double	95x1
ub	95x1 double	95x1
c	95x1 double	95x1
osenseStr	'max'	1x3
genes	137x1 cell	137x1
rules	95x1 cell	95x1
metCharges	72x1 int32	72x1
metFormulas	72x1 cell	72x1
metNames	72x1 cell	72x1
metInChlString	72x1 cell	72x1
metKEGGID	72x1 cell	72x1
metChEBIID	72x1 cell	72x1
metPubChemID	72x1 cell	72x1
grRules	95x1 cell	95x1
rxnGeneMat	95x137 sparse d...	95x137
rxnConfidence...	95x1 double	95x1
rxnNames	95x1 cell	95x1
rxnNotes	95x1 cell	95x1
rxnECNumbers	95x1 cell	95x1
rxnReferences	95x1 cell	95x1
subSystems	95x1 cell	95x1

The meaning of each field in a standard model is defined in the standard COBRA model field definition.

In general, the following fields should always be present:

- **S**, the stoichiometric matrix
- **mets**, the identifiers of the metabolites
- **b**, Accumulation (positive) or depletion (negative) of the corresponding metabolites. 0 Indicates no concentration change.
- **csense**, indicator whether the b vector is a lower bound ('G'), upper bound ('L'), or hard constraint 'E' for the metabolites.
- **rxns**, the identifiers of the reactions
- **lb**, the lower bounds of the reactions
- **ub**, the upper bounds of the reactions
- **c**, the linear objective
- **genes**, the list of genes in your model
- **rules**, the Gene-protein-reaction rules in a computer readable format present in your model.
- **osenseStr**, the objective sense either `'max'` for maximisation or `'min'` for minimisation

Sensitivity Analysis

In the E. coli core model, when maximising ATP production, what is the shadow price of cytosolic protons?

Hint: `FBAsolution.y`

```
model = modelOri;
model = changeRxnBounds(model,'EX_glc(e)',-1,'l');
model = changeRxnBounds(model,'EX_o2(e)',-1000,'l');
model = changeRxnBounds(model,'ATPM',0,'l');
model = changeObjective(model,'ATPM');
printConstraints(model,-1000,1000)
```

```
MinConstraints:
EX_glc(e)    -1
maxConstraints:
```

```
FBAsolution_maxATP = optimizeCbModel(model,'max');
```

Check the optimal value of the objective

```
FBAsolution_maxATP.f
```

```
ans = 17.5000
```

The shadow price of cytosolic protons (h[c]) is -0.25.

```
ind=strcmp(model.mets,'h[c]');
FBAsolution_maxATP.y(ind)
```

```
ans = -0.2500
```

```
printFluxVector(model, FBAsolution_maxATP.v, 1)
```

```
ACONTa                            2
ACONTb                            2
AKGDH                             2
ATPM                           17.5
ATPS4r                         13.5
CO2t                             -6
CS                                2
CYTBD                            12
ENO                               2
EX_co2(e)                         6
EX_glc(e)                        -1
EX_h2o(e)                         6
EX_o2(e)                         -6
FBA                               1
FUM                               2
GAPD                              2
GLCpts                            1
H2Ot                             -6
ICDHyr                            2
MDH                               2
NADH16                           10
NADTRHD                           2
O2t                               6
PDH                               2
PFK                               1
PGI                               1
PGK                              -2
PGM                              -2
PYK                               1
SUCDi                             2
SUCOAS                           -2
TPI                               1
```

What is your biochemical interpretation of this change in objective in the current context?

Hint: printFluxVector, drawFlux

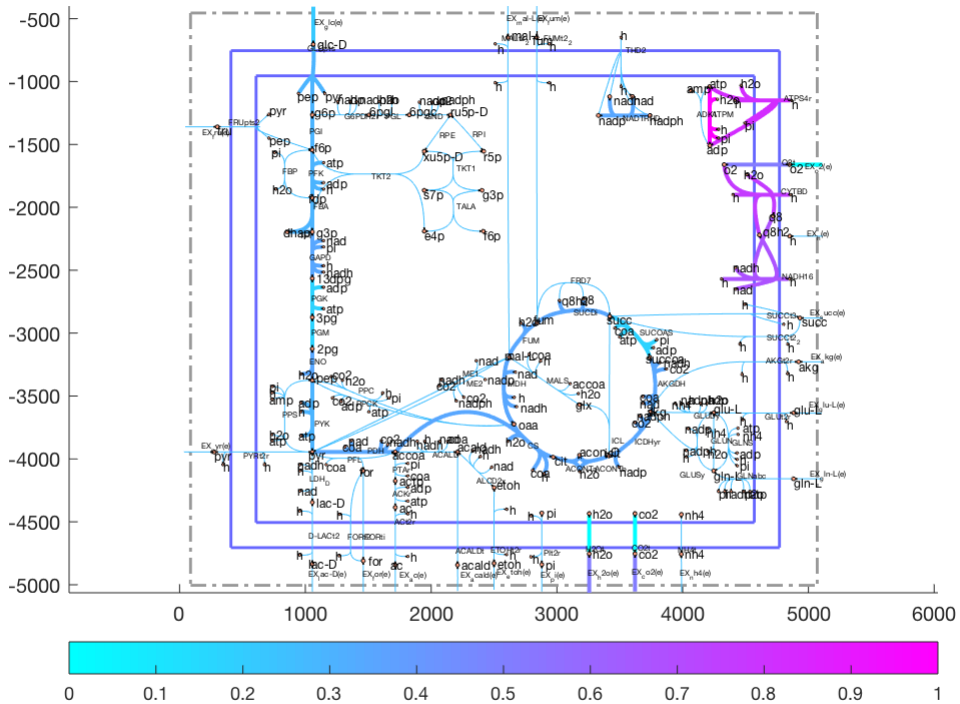
This is a unique solution (see Example 3).

```
dv = FBAsolution_maxATP_forceH.v-FBAsolution_maxATP.v;
dv(abs(dv)<1e-5)=0;
printFluxVector(model, dv, 1)
```

```
ATPM                              1
ATPS4r                            1
EX_h(e)                          -4
```

The flux map for optimal ATP production is shown below.

```
drawFlux(map, model, FBAsolution_maxATP.v, options);
```



ATP production is constrained by cytoplasmic proton balancing. Cytoplasmic protons are produced by various metabolic reactions and also enter into the cell, from the extracellular compartment, via the ATP synthase reaction (ATPS4r). At steady-state, an equal number of protons must be pumped out of the cytoplasm by the electron transport chain reactions or by excreting metabolites with symporters. Setting model.b(i) = 4, where i corresponds to cytoplasmic protons, h[c], removes 4 extra units of cytoplasmic protons from the system allowing 4 extra extracellular protons to enter the system that then enter the cell via the ATP synthase reaction, generating one extra unit of ATP. This increases the maximum rate of ATP synthesis by one unit, thereby increasing the ATP yield from glucose by 1 mol ATP/mol glucose.

Perturb the model in such a way as to increase the optimal rate of ATP hydrolysis ('ATPM') by exactly one unit. How does this compare with the theoretical prediction?

Hint: change model.b

Remove 4 units of cytoplasmic protons from the system, but changing model.b(i) to 4, where i corresponds to the index for cytoplasmic protons, and calculate the difference in the value of the optimal objective. The answer should be 1.

```
ind=strcmp(model.mets,'h[c]');
model.b(ind) = 4;
FBAsolution_maxATP_forceH = optimizeCbModel(model,'max');
FBAsolution_maxATP_forceH.f - FBAsolution_maxATP.f
```

```
ans = 1
```

In the E. coli core model, when maximising ATP production, what is the reduced cost of glucose exchange?

Hint: FBAsolution.rcost

```
rcost = FBAsolution_maxATP.rcost;
rcost(abs(rcost)<1e-4)=0;
flux=FBAsolution_maxATP.v;
printFluxVector(model, [model.lb,flux,model.ub,rcost], 1)
```

```
ACALD                          -1000           0        1000           0
ACALDt                         -1000           0        1000           0
ACKr                           -1000   1.202e-32        1000           0
ACONTa                         -1000           2        1000           0
ACONTb                         -1000           2        1000           0
ACt2r                          -1000          -0        1000           0
ADK1                           -1000           0        1000           0
AKGDH                              0           2        1000           0
AKGt2r                         -1000          -0        1000           0
ALCD2x                         -1000           0        1000           0
ATPM                               0        17.5        1000           0
ATPS4r                         -1000        13.5        1000           0
Biomass_Ecoli_core_N(w/GAM)-Nmet2               0           0        1000       188.3
CO2t                           -1000          -6        1000           0
CS                                 0           2        1000           0
CYTBD                              0          12        1000           0
D-LACt2                        -1000          -0        1000           0
ENO                            -1000           2        1000           0
ETOHt2r                        -1000          -0        1000           0
EX_ac(e)                           0           0        1000        4.25
EX_acald(e)                        0           0        1000         6.5
EX_akg(e)                          0           0        1000       11.75
EX_co2(e)                      -1000           6        1000           0
EX_etoh(e)                         0           0        1000         7.5
EX_for(e)                          0          -0        1000           0
EX_fru(e)                          0           0        1000        17.5
EX_fum(e)                          0           0        1000        8.75
EX_glc(e)                         -1          -1        1000        17.5
EX_gln-L(e)                        0           0        1000       13.25
EX_glu-L(e)                        0           0        1000          13
EX_h2o(e)                      -1000           6        1000           0
EX_h(e)                        -1000   1.449e-14        1000           0
EX_lac-D(e)                        0           0        1000        7.75
EX_mal-L(e)                        0           0        1000        8.75
EX_nh4(e)                      -1000          -0        1000           0
EX_o2(e)                       -1000          -6        1000           0
EX_pi(e)                       -1000  -1.593e-16        1000           0
EX_pyr(e)                          0           0        1000         6.5
EX_succ(e)                         0           0        1000          10
FBA                            -1000           1        1000           0
FBP                                0           0        1000           1
FORt2                              0           0        1000        0.25
FORti                              0           0        1000           0
FRD7                               0           0        1000           0
FRUpts2                            0           0        1000           0
FUM                            -1000           2        1000           0
FUMt2_2                            0           0        1000           0
G6PDH2r                        -1000           0        1000           0
GAPD                           -1000           2        1000           0
GLCpts                             0           1        1000           0
GLNS                               0           0        1000           0
GLNabc                             0           0        1000           0
GLUDy                          -1000           0        1000           0
GLUN                               0           0        1000           1
GLUSy                              0           0        1000           1
```

```
GLUt2r                          -1000          -0        1000           0
GND                                 0           0        1000      0.4167
H2Ot                            -1000          -6        1000           0
ICDHyr                          -1000           2        1000           0
ICL                                 0           0        1000           0
LDH_D                           -1000           0        1000           0
MALS                                0           0        1000           0
MALt2_2                             0           0        1000           0
MDH                             -1000           2        1000           0
ME1                                 0           0        1000           1
ME2                                 0           0        1000           1
NADH16                              0          10        1000           0
NADTRHD                             0           2        1000           0
NH4t                            -1000           0        1000           0
O2t                             -1000           6        1000           0
PDH                                 0           2        1000           0
PFK                                 0           1        1000           0
PFL                                 0           0        1000         1.5
PGI                             -1000           1        1000           0
PGK                             -1000          -2        1000           0
PGL                                 0           0        1000           0
PGM                             -1000          -2        1000           0
PIt2r                           -1000   1.593e-16        1000           0
PPC                                 0  -3.403e-16        1000           0
PPCK                                0           0        1000           1
PPS                                 0           0        1000           1
PTAr                            -1000  -1.202e-32        1000           0
PYK                                 0           1        1000           0
PYRt2r                          -1000          -0        1000           0
RPE                             -1000           0        1000           0
RPI                             -1000           0        1000           0
SUCCt2_2                            0           0        1000        0.75
SUCCt3                              0           0        1000           0
SUCDi                               0           2        1000           0
SUCOAS                          -1000          -2        1000           0
TALA                            -1000           0        1000           0
THD2                                0           0        1000         0.5
TKT1                            -1000           0        1000           0
TKT2                            -1000           0        1000           0
TPI                             -1000           1        1000           0
```

```
ind=strcmp(model.rxns,'EX_glc(e)');
FBAsolution_maxATP.rcost(ind)
```

```
ans = 17.5000
```

Display the change in the flux vector:

```
dv = FBAsolution_maxATP_moreGlc.v-FBAsolution_maxATP.v;
dv(abs(dv)<1e-4)=0;
printFluxVector(model, dv, 1)
```

```
ACONTa                           2
ACONTb                           2
AKGDH                            2
ATPM                          17.5
ATPS4r                        13.5
CO2t                            -6
CS                               2
CYTBD                           12
ENO                              2
EX_co2(e)                        6
```

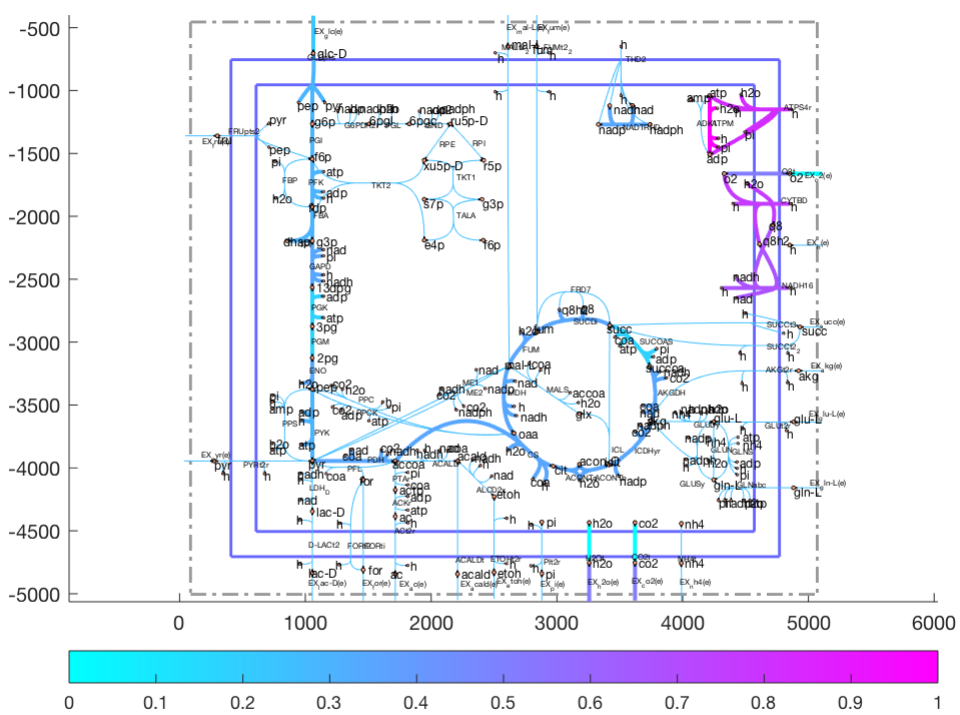
```
EX_glc(e)                    -1
EX_h2o(e)                     6
EX_o2(e)                     -6
FBA                           1
FUM                           2
GAPD                          2
GLCpts                        1
H2Ot                         -6
ICDHyr                        2
MDH                           2
NADH16                       10
NADTRHD                       2
O2t                           6
PDH                           2
PFK                           1
PGI                           1
PGK                          -2
PGM                          -2
PYK                           1
SUCDi                         2
SUCOAS                       -2
TPI                           1
```

What is your biochemical interpretation of this?

Hint: use drawFlux with a perturbed optimal reaction rate vector

The flux map for the perturbation to optimal ATP production is shown below. Note the reactions whose rates are substantially increasing, starting from glucose.

```
drawFlux(map, model, dv, options);
```



Perturb the model in such a way as to increase the optimal rate of ATP hydrolysis ('ATPM') by exactly 17.5 units. How does this compare with the theoretical prediction?

Hint: change model.lb

```
model = modelOri;
model = changeRxnBounds(model,'EX_glc(e)',-2,'l'); %note the change in the
lower bound from -1 to -2
model = changeRxnBounds(model,'EX_o2(e)',-1000,'l');
model = changeRxnBounds(model,'ATPM',0,'l');
model = changeObjective(model,'ATPM');
FBAsolution_maxATP_moreGlc = optimizeCbModel(model,'max');
```

By changing the lower bound on glucose exhange from -1 to -2, we see that the value of the objective increases by 17.5, which is equal to the reduced cost of glucose obtained from FBAsolution_maxATP.rcost:

```
FBAsolution_maxATP_moreGlc.f - FBAsolution_maxATP.f
```

```
ans = 17.5000
```

TROUBLESHOOTING

Note that, if an optimization problem is reformulated from a maximisation to a minimisation problem, then the signs of each of the dual variables is reversed.

TIMING

1 hr.

ANTICIPATED RESULTS

Understanding of how an optimal objective will change in response to changing the input data.

Acknowledgments

Part of this tutorial was originally written by Jeff Orth and Ines Thiele for the publication "What is flux balance analysis?"

REFERENCES

1. Orth. J., Thiele, I., Palsson, B.O., What is flux balance analysis? Nat Biotechnol. Mar; 28(3): 245–248 (2010).

2. Laurent Heirendt & Sylvain Arreckx, Thomas Pfau, Sebastian N. Mendoza, Anne Richelle, Almut Heinken, Hulda S. Haraldsdottir, Jacek Wachowiak, Sarah M. Keating, Vanja Vlasov, Stefania Magnusdottir, Chiam Yu Ng, German Preciat, Alise Zagare, Siu H.J. Chan, Maike K. Aurich, Catherine M. Clancy, Jennifer Modamio, John T. Sauls, Alberto Noronha, Aarash Bordbar, Benjamin Cousins, Diana C. El Assal, Luis V. Valcarcel, Inigo Apaolaza, Susan Ghaderi, Masoud Ahookhosh, Marouen Ben Guebila, Andrejs Kostromins, Nicolas Sompairac, Hoai M. Le, Ding Ma, Yuekai Sun, Lin Wang, James T. Yurkovich, Miguel A.P. Oliveira, Phan T. Vuong, Lemmer P. El Assal, Inna Kuperstein, Andrei Zinovyev, H. Scott Hinton, William A. Bryant, Francisco

J. Aragon Artacho, Francisco J. Planes, Egils Stalidzans, Alejandro Maass, Santosh Vempala, Michael Hucka, Michael A. Saunders, Costas D. Maranas, Nathan E. Lewis, Thomas Sauter, Bernhard Ø. Palsson, Ines Thiele, Ronan M.T. Fleming, **Creation and analysis of biochemical constraint-based models: the COBRA Toolbox v3.0**, Nature Protocols, volume 14, pages 639–702, 2019 doi.org/10.1038/s41596-018-0098-2.