Browse Networks in the Matlab Command Window Using surfNet

Author(s): Siu Hung Joshua Chan, Department of Chemical Engineering, The Pennsylvania State University

Reviewer(s):

INTRODUCTION

In this tutorial, we will demonstrate how to browse a COBRA model in verbal format in the Matlab command window through an initial call and interactive mouse clicking.

MATERIALS

EQUIPMENT SETUP

Start CobraToolbox

```
% initCobraToolbox;
```

```
     _____   _____   _____   _____     _____      |
    /  ___| /  _  \ |  _  \ |  _  \   / ___ \     |   COnstraint-Based Reconstruction and Analysis
    | |     | | | | | |_| | | |_| |  | |___| |    |   The COBRA Toolbox - 2017
    | |     | | | | |  _  { |  _  /  |  ___  |    |
    | |___  | |_| | | |_| | | | \ \  | |   | |    |   Documentation:
    \_____| \_____/ |_____/ |_|  \_\ |_|   |_|    |   http://opencobra.github.io/cobratoolbox
                                                   |

  > Checking if git is installed ...  Done.
  > Checking if the repository is tracked using git ...  Done.
  > Checking if curl is installed ...  Done.
  > Checking if remote can be reached ...  Done.
  > Initializing and updating submodules ... Done.
  > Adding all the files of The COBRA Toolbox ...  Done.
  > Define CB map output... set to svg.
  > Retrieving models ...    Done.
  > TranslateSBML is installed and working properly.
  > Configuring solver environment variables ...
    - [-*--] ILOG_CPLEX_PATH: /Users/sxc554/Applications/IBM/ILOG/CPLEX_Studio127/cplex/matlab/x86-64_osx
    - [*---] GUROBI_PATH: /Library/gurobi700/mac64/matlab
    - [----] TOMLAB_PATH :  --> set this path manually after installing the solver ( see instructions )
    - [-*--] MOSEK_PATH: /Users/sxc554/mosek/7/toolbox/r2013aom
    Done.
  > Checking available solvers and solver interfaces ... Done.
  > Setting default solvers ... Done.
  > Saving the MATLAB path ... Done.
    - The MATLAB path was saved in the default location.

  > Summary of available solvers and solver interfaces

    Support     LP   MILP     QP   MIQP    NLP
  ------------------------------------------------------------------------
  cplex_direct  full              0      0      0      0      -
  dqqMinos      full              1      -      -      -      -
  glpk          full              1      1      -      -      -
  gurobi        full              1      1      1      1      -
  ibm_cplex     full              0      0      0      -      -
  matlab        full              1      -      -      -      1
  mosek         full              1      1      1      -      -
  pdco          full              1      -      1      -      -
  quadMinos     full              1      -      -      -      1
  tomlab_cplex  full              0      0      0      0      -
  qpng          experimental      -      -      1      -      -
  tomlab_snopt  experimental      -      -      -      -      0
  gurobi_mex    legacy            0      0      0      0      -
  lindo_old     legacy            0      -      -      -      -
  lindo_legacy  legacy            0      -      -      -      -
  lp_solve      legacy            1      -      -      -      -
  opti          legacy            0      0      0      0      0
  ------------------------------------------------------------------------
  Total         -                 8      3      4      1      2

  + Legend: - = not applicable, 0 = solver not compatible or not installed, 1 = solver installed.


  > You can solve LP problems using: 'dqqMinos' - 'glpk' - 'gurobi' - 'matlab' - 'mosek' - 'pdco' - 'quadMinos' - 'lp_solve'
  > You can solve MILP problems using: 'glpk' - 'gurobi' - 'mosek'
  > You can solve QP problems using: 'gurobi' - 'mosek' - 'pdco' - 'qpng'
  > You can solve MIQP problems using: 'gurobi'
  > You can solve NLP problems using: 'matlab' - 'quadMinos'

  > Checking for available updates ...
  --> You cannot update your fork using updateCobraToolbox(). [b1da0e @ add-tutorial-browseNetwork].
      Please use the MATLAB.devTools (https://github.com/opencobra/MATLAB.devTools).
```

PROCEDURE

Load the *E. coli* iJO1366 model as an example model.

```
modelFileName = 'iJO1366.mat';
modelDirectory = getDistributedModelFolder(modelFileName); %Look up the folder for the distributed Models.
modelFileName= [modelDirectory filesep modelFileName]; % Get the full path. Necessary to be sure, that the right model is loaded
iJO1366 = readCbModel(modelFileName);
```

Browse a network

Browse the network by starting from an initial metabolite, e.g., D-glucose in the extracellular compartment.

```
surfNet(iJO1366, 'glc__D_e')
```

```
Met #1195  glc__D_e, D-Glucose, C6H12O6
Consuming reactions:
  #164  EX_glc__D_e, Bd: -10 / 1000, D-Glucose exchange
  glc__D_e <=>
  #1355  GLCtex_copy1, Bd: -1000 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e <=> glc__D_p
  #1356  GLCtex_copy2, Bd: 0 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e -> glc__D_p
Producing reactions: none

Show previous steps...
```

All reactions producing or consuming '**glc__D_e**' will have their reaction indices (#xxx), ids (.rxns), bounds (.lb/.ub), names (.rxnNames) and formulae printed on the command window. All reactions and the participating metabolites are hyperlinked. For example, **click** on the reaction '**GLCtex_copy1**'. (This is equivalent to run the following command.)

```
% called by clicking 'GLCtex_copy1'
surfNet([], 'GLCtex_copy1', 0, 'none', 0, 1, [], 0)
```

```
Rxn #1355  GLCtex_copy1, Bd: -1000 / 1000, Glucose transport via diffusion (extracellular to periplasm)
glc__D_e <=> glc__D_p
  id     Met       Stoich     metNames, metFormulas
Reactant:
 #1195   glc__D_e  -1         D-Glucose, C6H12O6
Product:
 #1587   glc__D_p  1          D-Glucose, C6H12O6

Show previous steps...
```

Details for the metabolites will appear, e.g., indeices, ids, stoichiometric coefficients, names and chemical formulae. By iteratively clicking on the reactions and metabolites that you are interested in, you can browse through the metabolic network.

Now, say you have gone through a series of metabolites and reactions (glc__D_e, GLCtex_copy1, glc__D_p, GLCptspp, g6p_c):

Click glc__D_p:

```
% called by clicking 'glc__D_p'
surfNet([], 'glc__D_p', 0, 'none', 0, 1, [], 0)
```

```
Met #1587  glc__D_p, D-Glucose, C6H12O6
Consuming reactions:
  #1336  GLCDpp, Bd: 0 / 1000, Glucose dehydrogenase (ubiquinone-8 as acceptor) (periplasm)
  q8_c + glc__D_p + h2o_p -> q8h2_c + glcn_p + h_p
  #1352  GLCabcpp, Bd: 0 / 1000, D-glucose transport via ABC system (periplasm)
  atp_c + h2o_c + glc__D_p -> adp_c + glc__D_c + h_c + pi_c
  #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
  pep_c + glc__D_p -> g6p_c + pyr_c
  #1354  GLCt2pp, Bd: 0 / 1000, D-glucose transport in via proton symport (periplasm)
  glc__D_p + h_p -> glc__D_c + h_c
Producing reactions:
  #1252  G1PPpp, Bd: 0 / 1000, Glucose-1-phosphatase
  g1p_p + h2o_p -> glc__D_p + pi_p
  #1355  GLCtex_copy1, Bd: -1000 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e <=> glc__D_p
  #1356  GLCtex_copy2, Bd: 0 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e -> glc__D_p
  #1607  LACZpp, Bd: 0 / 1000, B-galactosidase
  h2o_p + lcts_p -> gal_p + glc__D_p
  #2463  TREHpp, Bd: 0 / 1000, Alpha,alpha-trehalase (periplasm)
  h2o_p + tre_p -> 2 glc__D_p

Show previous steps...
```

Click GLCptspp:

```
% called by clicking 'GLCptspp'
surfNet([], 'GLCptspp', 0, 'none', 0, 1, [], 0)
```

```
Rxn #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
pep_c + glc__D_p -> g6p_c + pyr_c
  id     Met       Stoich     metNames, metFormulas
Reactant:
 #784       pep_c  -1         Phosphoenolpyruvate, C3H2O6P
 #1587   glc__D_p  -1         D-Glucose, C6H12O6
Product:
 #508       g6p_c  1          D-Glucose 6-phosphate, C6H11O9P
 #853       pyr_c  1          Pyruvate, C3H3O3

Show previous steps...
```

Click g6p_c:

```
% called by clicking 'g6p_c'
surfNet([], 'g6p_c', 0, 'none', 0, 1, [], 0)
```

```
Met #508  g6p_c, D-Glucose 6-phosphate, C6H11O9P
Consuming reactions:
  #1283  G6PDH2r, Bd: -1000 / 1000, Glucose 6-phosphate dehydrogenase
  g6p_c + nadp_c <=> 6pgl_c + h_c + nadph_c
  #1284  G6PP, Bd: 0 / 1000, Glucose-6-phosphate phosphatase
  g6p_c + h2o_c -> glc__D_c + pi_c
  #2077  PGI, Bd: -1000 / 1000, Glucose-6-phosphate isomerase
  g6p_c <=> f6p_c
  #2461  TRE6PS, Bd: 0 / 1000, Alpha,alpha-trehalose-phosphate synthase (UDP-forming)
  g6p_c + udpg_c -> h_c + tre6p_c + udp_c
Producing reactions:
  #477  AB6PGH, Bd: 0 / 1000, Arbutin 6-phosphate glucohydrolase
  arbt6p_c + h2o_c -> g6p_c + hqn_c
  #1214  FFSD, Bd: 0 / 1000, Beta-fructofuranosidase
  h2o_c + suc6p_c -> fru_c + g6p_c
  #1231  FRULYSDG, Bd: -1000 / 1000, Fructoselysine phosphate deglycase
  frulysp_c + h2o_c <=> g6p_c + lys__L_c
  #1285  G6Pt6_2pp, Bd: 0 / 1000, Glucose-6-phosphate transport via phosphate antiport (periplasm)
  2 pi_c + g6p_p -> g6p_c + 2 pi_p
  #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
  pep_c + glc__D_p -> g6p_c + pyr_c
  #1500  HEX1, Bd: 0 / 1000, Hexokinase (D-glucose:ATP)
  atp_c + glc__D_c -> adp_c + g6p_c + h_c
  #2082  PGMT, Bd: -1000 / 1000, Phosphoglucomutase
  g1p_c <=> g6p_c
  #2459  TRE6PH, Bd: 0 / 1000, Trehalose-6-phosphate hydrolase
  h2o_c + tre6p_c -> g6p_c + glc__D_c

Show previous steps...
```

In each click, there is also a button '**Show previous steps...**' at the bottom. Clicking on it will show the metabolites and reactions that you have visited in order. This is equivalent to calling:

```
% called by clicking 'Show previous steps...'
surfNet([], [], 0, 'none', 0, 1, [], 0, struct('showPrev', true))
```

```
glc__D_e>>GLCtex_copy1>>glc__D_p>>GLCptspp>>g6p_c>>
```

You can go back to any of the intermediate metabolites/reactions by clicking the hyperlinked `mets/rxns` shown.

Call options

Shown below are various call options for including flux vectors and customizing display. All call options are preserved during the interactive browsing by mouse clicking.

Show objective reactions

Omit the '`metrxn`' (2nd) argument to print objective reactions:

```
surfNet(iJO1366)
```

```
Rxn #8  BIOMASS_Ec_iJO1366_core_53p95M, Bd: 0 / 1000, E. coli biomass objective function (iJO1366) - core - with 53.95 GAM estimate
0.000223 10fthf_c + 2.6e-05 2fe2s_c + 0.000223 2ohph_c + 0.00026 4fe4s_c + 0.51369 ala__L_c + 0.000223 amet_c +
  0.29579 arg__L_c + 0.24105 asn__L_c + 0.24105 asp__L_c + 54.1248 atp_c + 0.000122 bmocogdp_c + 2e-06 btn_c +
  0.005205 ca2_c + 0.005205 cl_c + 0.000576 coa_c + 2.5e-05 cobalt2_c + 0.13351 ctp_c + 0.000709 cu2_c +
  0.09158 cys__L_c + 0.026166 datp_c + 0.027017 dctp_c + 0.027017 dgtp_c + 0.026166 dttp_c + 0.000223 fad_c +
  0.006715 fe2_c + 0.007808 fe3_c + 0.26316 gln__L_c + 0.26316 glu__L_c + 0.61264 gly_c + 0.2151 gtp_c +
  48.6015 h2o_c + 0.094738 his__L_c + 0.29053 ile__L_c + 0.19519 k_c + 0.45053 leu__L_c + 0.34316 lys__L_c +
  0.15369 met__L_c + 0.008675 mg2_c + 0.000223 mlthf_c + 0.000691 mn2_c + 7e-06 mobd_c + 0.001831 nad_c +
  0.000447 nadp_c + 0.013013 nh4_c + 0.000323 ni2_c + 0.017868 pe160_c + 0.054154 pe161_c + 0.18527 phe__L_c +
  0.000223 pheme_c + 0.22106 pro__L_c + 0.000223 pydx5p_c + 0.000223 ribflv_c + 0.21579 ser__L_c + 0.000223 sheme_c
  + 0.004338 so4_c + 0.000223 thf_c + 0.000223 thmpp_c + 0.25369 thr__L_c + 0.056843 trp__L_c + 0.1379 tyr__L_c +
  5.5e-05 udcpdp_c + 0.1441 utp_c + 0.42316 val__L_c + 0.000341 zn2_c + 0.019456 kdo2lipid4_e +
  0.013894 murein5px4p_p + 0.045946 pe160_p + 0.02106 pe161_p -> 53.95 adp_c + 53.95 h_c + 53.9457 pi_c +
  0.7739 ppi_c
  id     Met            Stoich     metNames, metFormulas
Reactant:
 #1          10fthf_c  -0.000223  10-Formyltetrahydrofolate, C20H21N7O7
 #69          2fe2s_c  -0.000026  [2Fe-2S] iron-sulfur cluster, S2Fe2
 #82          2ohph_c  -0.000223  2-Octaprenyl-6-hydroxyphenol, C46H70O2
 #167         4fe4s_c  -0.00026   [4Fe-4S] iron-sulfur cluster, S4Fe4
 #255        ala__L_c  -0.513689  L-Alanine, C3H7NO2
 #265          amet_c  -0.000223  S-Adenosyl-L-methionine, C15H23N6O5S
 #294        arg__L_c  -0.295792  L-Arginine, C6H15N4O2
 #298        asn__L_c  -0.241055  L-Asparagine, C4H8N2O3
 #302        asp__L_c  -0.241055  L-Aspartate, C4H6NO4
 #307           atp_c  -54.124831  ATP, C10H12N5O13P3
 #314      bmocogdp_c  -0.000122  Bis-molybdopterin guanine dinucleotide, C40H44N20O27P4S4Mo
 #317           btn_c  -0.000002  Biotin, C10H15N2O3S
 #326           ca2_c  -0.005205  Calcium, Ca
 #355            cl_c  -0.005205  Chloride, Cl
 #358           coa_c  -0.000576  Coenzyme A, C21H32N7O16P3S
 #359       cobalt2_c  -0.000025  Co2+, Co
 #377           ctp_c  -0.133508  CTP, C9H12N3O14P3
 #379           cu2_c  -0.000709  Cu2+, Cu
 #383        cys__L_c  -0.09158   L-Cysteine, C3H7NO2S
 #392          datp_c  -0.026166  DATP, C10H12N5O12P3
 #401          dctp_c  -0.027017  DCTP, C9H12N3O13P3
 #412          dgtp_c  -0.027017  DGTP, C10H12N5O13P3
 #451          dttp_c  -0.026166  DTTP, C10H13N2O14P3
 #468           fad_c  -0.000223  Flavin adenine dinucleotide oxidized, C27H31N9O15P2
 #474           fe2_c  -0.006715  Fe2+, Fe
 #475           fe3_c  -0.007808  Fe3+, Fe
```

Call with a list of mets/rxns

The 'metrxn' arguement can be a string of id for a metabolite or reaction. It can also be a cell array of ids, e.g.,

```
surfNet(iJO1366, {'glc__D_p'; 'GLCptspp'; 'g6p_c'})
```

```
Met #1587  glc__D_p, D-Glucose, C6H12O6
Consuming reactions:
  #1336  GLCDpp, Bd: 0 / 1000, Glucose dehydrogenase (ubiquinone-8 as acceptor) (periplasm)
  q8_c + glc__D_p + h2o_p -> q8h2_c + glcn_p + h_p
  #1352  GLCabcpp, Bd: 0 / 1000, D-glucose transport via ABC system (periplasm)
  atp_c + h2o_c + glc__D_p -> adp_c + glc__D_c + h_c + pi_c
  #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
  pep_c + glc__D_p -> g6p_c + pyr_c
  #1354  GLCt2pp, Bd: 0 / 1000, D-glucose transport in via proton symport (periplasm)
  glc__D_p + h_p -> glc__D_c + h_c
Producing reactions:
  #1252  G1PPpp, Bd: 0 / 1000, Glucose-1-phosphatase
  g1p_p + h2o_p -> glc__D_p + pi_p
  #1355  GLCtex_copy1, Bd: -1000 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e <=> glc__D_p
  #1356  GLCtex_copy2, Bd: 0 / 1000, Glucose transport via diffusion (extracellular to periplasm)
  glc__D_e -> glc__D_p
  #1607  LACZpp, Bd: 0 / 1000, B-galactosidase
  h2o_p + lcts_p -> gal_p + glc__D_p
  #2463  TREHpp, Bd: 0 / 1000, Alpha,alpha-trehalase (periplasm)
  h2o_p + tre_p -> 2 glc__D_p

Rxn #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
pep_c + glc__D_p -> g6p_c + pyr_c
   id     Met       Stoich     metNames, metFormulas
Reactant:
 #784       pep_c  -1         Phosphoenolpyruvate, C3H2O6P
 #1587   glc__D_p  -1         D-Glucose, C6H12O6
Product:
 #508       g6p_c  1          D-Glucose 6-phosphate, C6H11O9P
 #853       pyr_c  1          Pyruvate, C3H3O3

Met #508  g6p_c, D-Glucose 6-phosphate, C6H11O9P
Consuming reactions:
  #1283  G6PDH2r, Bd: -1000 / 1000, Glucose 6-phosphate dehydrogenase
  g6p_c + nadp_c <=> 6pgl_c + h_c + nadph_c
  #1284  G6PP, Bd: 0 / 1000, Glucose-6-phosphate phosphatase
  g6p_c + h2o_c -> glc__D_c + pi_c
  #2077  PGI, Bd: -1000 / 1000, Glucose-6-phosphate isomerase
  g6p_c <=> f6p_c
  #2461  TRE6PS, Bd: 0 / 1000, Alpha,alpha-trehalose-phosphate synthase (UDP-forming)
  g6p_c + udpg_c -> h_c + tre6p_c + udp_c
Producing reactions:
  #477  AB6PGH, Bd: 0 / 1000, Arbutin 6-phosphate glucohydrolase
  arbt6p_c + h2o_c -> g6p_c + hqn_c
  #1214  FFSD, Bd: 0 / 1000, Beta-fructofuranosidase
  h2o_c + suc6p_c -> fru_c + g6p_c
  #1231  FRULYSDG, Bd: -1000 / 1000, Fructoselysine phosphate deglycase
  frulysp_c + h2o_c <=> g6p_c + lys__L_c
  #1285  G6Pt6_2pp, Bd: 0 / 1000, Glucose-6-phosphate transport via phosphate antiport (periplasm)
  2 pi_c + g6p_p -> g6p_c + 2 pi_p
  #1353  GLCptspp, Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
  pep_c + glc__D_p -> g6p_c + pyr_c
  #1500  HEX1, Bd: 0 / 1000, Hexokinase (D-glucose:ATP)
  atp_c + glc__D_c -> adp_c + g6p_c + h_c
```

Show metabolite names in reaction formulae

Some models may use generic ids for `mets/rxns`. In this case, call `surfNet()` with the '`metNameFlag`' (3rd) arguement turned on to show the names for metabolites (`.metNames`) in the reaction formulae, e.g.,

```
surfNet(iJO1366, 'fgam_c', 1)
```

```
Met #484  fgam_c, N2-Formyl-N1-(5-phospho-D-ribosyl)glycinamide, C8H13N2O9P
Consuming reactions:
  #2207  PRFGS, Bd: 0 / 1000, Phosphoribosylformylglycinamidine synthase
  ATP + N2-Formyl-N1-(5-phospho-D-ribosyl)glycinamide + L-Glutamine + H2O -> ADP + 2-(Formamido)-N1-(5-phospho-D-ribosyl)acetamidine
  + L-Glutamate + H+ + Phosphate
Producing reactions:
  #1316  GARFT, Bd: -1000 / 1000, Phosphoribosylglycinamide formyltransferase
  10-Formyltetrahydrofolate + N1-(5-Phospho-D-ribosyl)glycinamide <=> N2-Formyl-N1-(5-phospho-D-ribosyl)glycinamide + H+
  + 5,6,7,8-Tetrahydrofolate
  #1317  GART, Bd: 0 / 1000, GAR transformylase-T
  ATP + Formate + N1-(5-Phospho-D-ribosyl)glycinamide -> ADP + N2-Formyl-N1-(5-phospho-D-ribosyl)glycinamide + H+ + Phosphate


Show previous steps...
```

Hide reaction detials

Turn off the '`showMets`' (6th) arguement to suppress details for reactions, e.g.,

```
surfNet(iJO1366, iJO1366.rxns(1001:1010), [], [], [], 0)
```

```
Rxn #1001  DHPPDA2, Bd: 0 / 1000, Diaminohydroxyphosphoribosylaminopyrimidine deaminase (25drapp)
25drapp_c + h_c + h2o_c -> 5apru_c + nh4_c

Rxn #1002  DHPS2, Bd: 0 / 1000, Dihydropteroate synthase
4abz_c + 6hmhptpp_c -> dhpt_c + ppi_c

Rxn #1003  DHPTDCs2, Bd: 0 / 1000, 4,5-dihydroxy-2,3-pentanedione cyclization (spontaneous)
dhptd_c -> mdhdhf_c

Rxn #1004  DHPTDNR, Bd: 0 / 0, Dihydropteridine reductase
dhptdn_c + 3 h_c + nadph_c -> nadp_c + thptdn_c

Rxn #1005  DHPTDNRN, Bd: 0 / 0, Dihydropteridine reductase (NADH)
dhptdn_c + 3 h_c + nadh_c -> nad_c + thptdn_c

Rxn #1006  DHPTPE, Bd: -1000 / 1000, Dihydroneopterin triphosphate 2'-epimerase
ahdt_c <=> dhmptp_c

Rxn #1007  DHQS, Bd: 0 / 1000, 3-dehydroquinate synthase
2dda7p_c -> 3dhq_c + pi_c

Rxn #1008  DHQTi, Bd: 0 / 1000, 3-dehydroquinate dehydratase, irreversible
3dhq_c -> 3dhsk_c + h2o_c

Rxn #1009  DIMPtex, Bd: -1000 / 1000, DIMP transport via diffusion (extracellular to periplasm)
dimp_e <=> dimp_p

Rxn #1010  DINSt2pp, Bd: 0 / 1000, Deoxyinosine transport in via proton symport (periplasm)
din_p + h_p -> din_c + h_c

Show previous steps...
```

Look at one or more flux distributions

First, get a flux distribution by optimizing the biomass production of the model (the standard flux balance analysis[1]). Then call surfNet with the flux distribution (4th argument) to look at how the flux through pyruvate is distributed:

```
s = optimizeCbModel(iJO1366, 'max', 'one');
surfNet(iJO1366, 'pyr_c', [], s.x)
```

```
Met #853  pyr_c, Pyruvate, C3H3O3
Consuming reactions with non-zero fluxes :
  #511  ACHBS (0.28541), Bd: 0 / 1000, 2-aceto-2-hydroxybutanoate synthase
  2obut_c + h_c + pyr_c -> 2ahbut_c + co2_c
  #513  ACLS (0.85886), Bd: 0 / 1000, Acetolactate synthase
  h_c + 2 pyr_c -> alac__S_c + co2_c
  #618  ALATA_L (-0.57111), Bd: -1000 / 1000, L-alanine transaminase
  akg_c + ala__L_c <=> glu__L_c + pyr_c
  #987  DHDPS (0.36441), Bd: 0 / 1000, Dihydrodipicolinate synthase
  aspsa_c + pyr_c -> 23dhdp_c + h_c + 2 h2o_c
  #1053  DXPS (0.00279), Bd: 0 / 1000, 1-deoxy-D-xylulose 5-phosphate synthase
  g3p_c + h_c + pyr_c -> co2_c + dxyl5p_c
  #2047  PDH (7.96919), Bd: 0 / 1000, Pyruvate dehydrogenase
  coa_c + nad_c + pyr_c -> accoa_c + co2_c + nadh_c
  #2171  POR5 (0.10684), Bd: -1000 / 1000, Pyruvate synthase
  coa_c + 2 flxso_c + pyr_c <=> accoa_c + co2_c + 2 flxr_c + h_c
  #2466  TRPAS2 (-0.05584), Bd: -1000 / 1000, Tryptophanase (L-tryptophan)
  h2o_c + trp__L_c <=> indole_c + nh4_c + pyr_c
Producing reactions with non-zero fluxes :
  #554  ADCL (0.00066), Bd: 0 / 1000, 4-aminobenzoate synthase
  4adcho_c -> 4abz_c + h_c + pyr_c
  #666  ANS (0.05584), Bd: 0 / 1000, Anthranilate synthase
  chor_c + gln__L_c -> anth_c + glu__L_c + h_c + pyr_c
  #813  CHRPL (0.00022), Bd: 0 / 1000, Chorismate pyruvate lyase
  chor_c -> 4hbz_c + pyr_c
  #908  CYSTL (0.1512), Bd: 0 / 1000, Cystathionine b-lyase
  cyst__L_c + h2o_c -> hcys__L_c + nh4_c + pyr_c
  #978  DHAPT (0.86538), Bd: 0 / 1000, Dihydroxyacetone phosphotransferase
  dha_c + pep_c -> dhap_c + pyr_c
  #1353  GLCptspp (10), Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
  pep_c + glc__D_p -> g6p_c + pyr_c

Show previous steps...
```

All reactions involving pyruvate with non-zero fluxes are printed. The flux values are in the parentheses following the reaction ids. Note that reactions stated as consuming or producing the metabolite have taken the directions of the fluxes into account. Therefore, supplying a different flux distribution or not supplying may give different display. By default, only reactions with non-zero fluxes are printed if a flux distribution is supplied. Turn the '`nonzeroFluxFlag`' (5th) argument off to show all reactions:

```
surfNet(iJO1366, 'pyr_c', [], s.x, 0)
```

```
Met #853  pyr_c, Pyruvate, C3H3O3
Consuming reactions:
  #511  ACHBS (0.28541), Bd: 0 / 1000, 2-aceto-2-hydroxybutanoate synthase
  2obut_c + h_c + pyr_c -> 2ahbut_c + co2_c
  #513  ACLS (0.85886), Bd: 0 / 1000, Acetolactate synthase
  h_c + 2 pyr_c -> alac__S_c + co2_c
  #618  ALATA_L (-0.57111), Bd: -1000 / 1000, L-alanine transaminase
  akg_c + ala__L_c <=> glu__L_c + pyr_c
  #987  DHDPS (0.36441), Bd: 0 / 1000, Dihydrodipicolinate synthase
  aspsa_c + pyr_c -> 23dhdp_c + h_c + 2 h2o_c
  #1053  DXPS (0.00279), Bd: 0 / 1000, 1-deoxy-D-xylulose 5-phosphate synthase
  g3p_c + h_c + pyr_c -> co2_c + dxyl5p_c
  #2047  PDH (7.96919), Bd: 0 / 1000, Pyruvate dehydrogenase
  coa_c + nad_c + pyr_c -> accoa_c + co2_c + nadh_c
  #2067  PFL (0), Bd: 0 / 1000, Pyruvate formate lyase
  coa_c + pyr_c -> accoa_c + for_c
  #2171  POR5 (0.10684), Bd: -1000 / 1000, Pyruvate synthase
  coa_c + 2 flxso_c + pyr_c <=> accoa_c + co2_c + 2 flxr_c + h_c
  #2172  POX (0), Bd: 0 / 1000, Pyruvate oxidase
  h2o_c + pyr_c + q8_c -> ac_c + co2_c + q8h2_c
  #2198  PPS (0), Bd: 0 / 1000, Phosphoenolpyruvate synthase
  atp_c + h2o_c + pyr_c -> amp_c + 2 h_c + pep_c + pi_c
  #2466  TRPAS2 (-0.05584), Bd: -1000 / 1000, Tryptophanase (L-tryptophan)
  h2o_c + trp__L_c <=> indole_c + nh4_c + pyr_c
Producing reactions:
  #507  ACGAptspp (0), Bd: 0 / 1000, N-Acetyl-D-glucosamine transport via PEP:Pyr PTS  (periplasm)
  pep_c + acgam_p -> acgam6p_c + pyr_c
  #516  ACMANAptspp (0), Bd: 0 / 1000, N-acetyl-D-mannosamine transport via PTS  (periplasm)
  pep_c + acmana_p -> acmanap_c + pyr_c
  #518  ACMUMptspp (0), Bd: 0 / 1000, N-acetylmuramate transport via PEP:Pyr PTS (periplasm)
  pep_c + acmum_p -> acmum6p_c + pyr_c
  #522  ACNML (0), Bd: 0 / 1000, N-Acetylneuraminate lyase
  acnam_c -> acmana_c + pyr_c
  #554  ADCL (0.00066), Bd: 0 / 1000, 4-aminobenzoate synthase
  4adcho_c -> 4abz_c + h_c + pyr_c
  #617  ALATA_D2 (0), Bd: 0 / 1000, D-alanine transaminase
  ala__D_c + pydx5p_c -> pyam5p_c + pyr_c
  #619  ALATA_L2 (0), Bd: 0 / 1000, Alanine transaminase
  ala__L_c + pydx5p_c -> pyam5p_c + pyr_c
  #666  ANS (0.05584), Bd: 0 / 1000, Anthranilate synthase
  chor_c + gln__L_c -> anth_c + glu__L_c + h_c + pyr_c
  #698  ARBTptspp (0), Bd: 0 / 1000, Arbutin transport via PEP:Pyr PTS (periplasm)
  pep_c + arbt_p -> arbt6p_c + pyr_c
  #716  ASCBptspp (0), Bd: 0 / 1000, L-ascorbate transport via PEP:Pyr PTS (periplasm)
```

You can also compare multiple flux distributions by supplying them in a matrix format, each column being a flux distribution. For example, get another flux distribution maximizing the biomass production using D-fructose instead of glucose as substrate. Then call surfNet to look at reactions with different fluxes.

Original uptake rates:

```
printUptakeBound(iJO1366);
```

```
EX_ca2_e -1000
EX_cbl1_e -0.01
EX_cl_e -1000
EX_co2_e -1000
EX_cobalt2_e -1000
EX_cu2_e -1000
EX_fe2_e -1000
EX_fe3_e -1000
EX_glc__D_e -10
EX_h_e -1000
EX_h2o_e -1000
EX_k_e -1000
EX_mg2_e -1000
EX_mn2_e -1000
EX_mobd_e -1000
EX_na1_e -1000
EX_nh4_e -1000
EX_ni2_e -1000
EX_o2_e -1000
EX_pi_e -1000
EX_sel_e -1000
EX_slnt_e -1000
EX_so4_e -1000
EX_tungs_e -1000
EX_zn2_e -1000
```

Use fructose instead of glucose as substrate:

```
iJO1366 = changeRxnBounds(iJO1366, {'EX_glc__D_e'; 'EX_fru_e'},...
    [0; -10], {'L'; 'L'});
printUptakeBound(iJO1366);
```

```
EX_ca2_e -1000
EX_cbl1_e -0.01
EX_cl_e -1000
EX_co2_e -1000
EX_cobalt2_e -1000
EX_cu2_e -1000
EX_fe2_e -1000
EX_fe3_e -1000
EX_fru_e -10
EX_h_e -1000
EX_h2o_e -1000
EX_k_e -1000
EX_mg2_e -1000
EX_mn2_e -1000
EX_mobd_e -1000
EX_na1_e -1000
EX_nh4_e -1000
EX_ni2_e -1000
EX_o2_e -1000
EX_pi_e -1000
EX_sel_e -1000
EX_slnt_e -1000
EX_so4_e -1000
EX_tungs_e -1000
EX_zn2_e -1000
```

Run FBA again to get a flux distribution using fructose as substrate. Then look at reactions with different fluxes in the glucose and fructose cases using `surfNet`.

```
sFru = optimizeCbModel(iJO1366, 'max', 'one');  % FBA
fluxMatrix = [s.x, sFru.x];  % put two flux vectors in a matrix
% reactions with different fluxes
rxnDiff = abs(fluxMatrix(:, 1) - fluxMatrix(:, 2)) > 1e-6;
surfNet(iJO1366, iJO1366.rxns(rxnDiff), [], fluxMatrix, [], 0)
```

```
Rxn #139  EX_fru_e (0, -10), Bd: -10 / 1000, D-Fructose exchange
fru_e <=>

Rxn #164  EX_glc__D_e (-10, 0), Bd: 0 / 1000, D-Glucose exchange
glc__D_e ->

Rxn #623  ALAt2pp_copy2 (-0.00511, 0), Bd: -1000 / 1000, L-alanine transport in via proton symport (periplasm)
ala__L_p + h_p <=> ala__L_c + h_c

Rxn #624  ALAt4pp (0.00511, 0), Bd: 0 / 1000, L-alanine transport in via sodium symport (periplasm)
ala__L_p + na1_p -> ala__L_c + na1_c

Rxn #1230  FRUK (0, 5.75203), Bd: 0 / 1000, Fructose-1-phosphate kinase
atp_c + f1p_c -> adp_c + fdp_c + h_c

Rxn #1238  FRUpts2pp (0, 4.24797), Bd: 0 / 1000, Fructose transport via PEP:Pyr PTS (f6p generating) (periplasm)
pep_c + fru_p -> f6p_c + pyr_c

Rxn #1239  FRUptspp (0, 5.75203), Bd: 0 / 1000, D-fructose transport via PEP:Pyr PTS (periplasm)
pep_c + fru_p -> f1p_c + pyr_c

Rxn #1240  FRUtex (-0, 10), Bd: -1000 / 1000, D-fructose transport via diffusion (extracellular to periplasm)
fru_e <=> fru_p

Rxn #1353  GLCptspp (10, 0), Bd: 0 / 1000, D-glucose transport via PEP:Pyr PTS (periplasm)
pep_c + glc__D_p -> g6p_c + pyr_c

Rxn #1356  GLCtex_copy2 (10, 0), Bd: 0 / 1000, Glucose transport via diffusion (extracellular to periplasm)
glc__D_e -> glc__D_p

Rxn #1377  GLUt2rpp (0, -0.00511), Bd: -1000 / 1000, L-glutamate transport via proton symport, reversible (periplasm)
glu__L_p + h_p <=> glu__L_c + h_c

Rxn #1378  GLUt4pp (0, 0.00511), Bd: 0 / 1000, Na+/glutamate symport (periplasm)
glu__L_p + na1_p -> glu__L_c + na1_c

Rxn #1758  MDH (4.82506, 4.82528), Bd: -1000 / 1000, Malate dehydrogenase
mal__L_c + nad_c <=> h_c + nadh_c + oaa_c

Rxn #1837  MOX (0.0016, 0.00138), Bd: -1000 / 1000, Malate oxidase
mal__L_c + o2_c <=> h2o2_c + oaa_c

Rxn #2048  PDX5PO2 (0.00022, 0), Bd: 0 / 1000, Pyridoxine 5'-phosphate oxidase (anaerboic
nad_c + pdx5p_c -> h_c + nadh_c + pydx5p_c

Rxn #2049  PDX5POi (0, 0.00022), Bd: 0 / 1000, Pyridoxine 5'-phosphate oxidase
o2_c + pdx5p_c -> h2o2_c + pydx5p_c

Rxn #2064  PFK (5.75203, 0), Bd: 0 / 1000, Phosphofructokinase
atp_c + f6p_c -> adp_c + fdp_c + h_c

Rxn #2077  PGI (5.91807, -4.08193), Bd: -1000 / 1000, Glucose-6-phosphate isomerase
g6p_c <=> f6p_c

Show previous steps...
```

Customize model data to be displayed

Customize the fields for metabolites and reactions to be printed by supplying the '`field2print`' (7th) argument. It is defaulted to be:

```
{{'metNames','metFormulas'}, {'rxnNames','lb','ub'}}
```

The first cell contains the metabolite-related fields to be printed and the second cell contains the reaction-related fields to be printed. It can also be inputted as a single cell array of strings, as long as from the size (equal to `#mets` or `#rxns`) or from the name of the field (starting with '`met`' or '`rxn`'), the fields are recognizable to be met- or rxn-related. For example, show the `grRules` for rxns but omit the bounds and show the constraint sense (`csense`) associated with each metabolite. Note the difference from the original call:

```
surfNet(iJO1366, 'fdp_c', [], [], [], [],...
```

```
Met #473  fdp_c, D-Fructose 1,6-bisphosphate, C6H10O12P2, csense: E
Consuming reactions with non-zero fluxes :
  #1151  FBA (5.75203, 5.75203), Fructose-bisphosphate aldolase, grRules: b2097 or b1773 or b2925
  fdp_c <=> dhap_c + g3p_c
Producing reactions with non-zero fluxes :
  #1230  FRUK (0, 5.75203), Fructose-1-phosphate kinase, grRules: b2168
  atp_c + f1p_c -> adp_c + fdp_c + h_c
  #2064  PFK (5.75203, 0), Phosphofructokinase, grRules: b3916 or b1723
  atp_c + f6p_c -> adp_c + fdp_c + h_c

Show previous steps...
```

```
    {'metNames', 'metFormulas', 'rxnNames', 'grRules', 'csense'})
surfNet(iJO1366, 'fdp_c')
```

```
Met #473  fdp_c, D-Fructose 1,6-bisphosphate, C6H10O12P2
Consuming reactions:
  #1151  FBA, Bd: -1000 / 1000, Fructose-bisphosphate aldolase
  fdp_c <=> dhap_c + g3p_c
  #1153  FBP, Bd: 0 / 1000, Fructose-bisphosphatase
  fdp_c + h2o_c -> f6p_c + pi_c
Producing reactions:
  #1230  FRUK, Bd: 0 / 1000, Fructose-1-phosphate kinase
  atp_c + f1p_c -> adp_c + fdp_c + h_c
  #2064  PFK, Bd: 0 / 1000, Phosphofructokinase
  atp_c + f6p_c -> adp_c + fdp_c + h_c

Show previous steps...
```

The last argument (8th) 'nCharBreak' sets the number of characters printed per line. By default, it is equal to the width of the Matlab command window. Note the difference:

Characters per line = width of the command window (default):

```
surfNet(iJO1366, [], [], [], [], 0)
```

```
Rxn #8  BIOMASS_Ec_iJO1366_core_53p95M, Bd: 0 / 1000, E. coli biomass objective function (iJO1366) - core - with 53.95 GAM estimate
0.000223 10fthf_c + 2.6e-05 2fe2s_c + 0.000223 2ohph_c + 0.00026 4fe4s_c + 0.51369 ala__L_c + 0.000223 amet_c +
  0.29579 arg__L_c + 0.24105 asn__L_c + 0.24105 asp__L_c + 54.1248 atp_c + 0.000122 bmocogdp_c + 2e-06 btn_c +
  0.005205 ca2_c + 0.005205 cl_c + 0.000576 coa_c + 2.5e-05 cobalt2_c + 0.13351 ctp_c + 0.000709 cu2_c +
  0.09158 cys__L_c + 0.026166 datp_c + 0.027017 dctp_c + 0.027017 dgtp_c + 0.026166 dttp_c + 0.000223 fad_c +
  0.006715 fe2_c + 0.007808 fe3_c + 0.26316 gln__L_c + 0.26316 glu__L_c + 0.61264 gly_c + 0.2151 gtp_c +
  48.6015 h2o_c + 0.094738 his__L_c + 0.29053 ile__L_c + 0.19519 k_c + 0.45053 leu__L_c + 0.34316 lys__L_c +
  0.15369 met__L_c + 0.008675 mg2_c + 0.000223 mlthf_c + 0.000691 mn2_c + 7e-06 mobd_c + 0.001831 nad_c +
  0.000447 nadp_c + 0.013013 nh4_c + 0.000323 ni2_c + 0.017868 pe160_c + 0.054154 pe161_c + 0.18527 phe__L_c +
  0.000223 pheme_c + 0.22106 pro__L_c + 0.000223 pydx5p_c + 0.000223 ribflv_c + 0.21579 ser__L_c + 0.000223 sheme_c
  + 0.004338 so4_c + 0.000223 thf_c + 0.000223 thmpp_c + 0.25369 thr__L_c + 0.056843 trp__L_c + 0.1379 tyr__L_c +
  5.5e-05 udcpdp_c + 0.1441 utp_c + 0.42316 val__L_c + 0.000341 zn2_c + 0.019456 kdo2lipid4_e +
  0.013894 murein5px4p_p + 0.045946 pe160_p + 0.02106 pe161_p -> 53.95 adp_c + 53.95 h_c + 53.9457 pi_c +
  0.7739 ppi_c

Show previous steps...
```

40 characters per line:

```
surfNet(iJO1366, [], [], [], [], 0, [], 40)
```

```
Rxn #8  BIOMASS_Ec_iJO1366_core_53p95M, Bd: 0 / 1000, E. coli biomass objective function (iJO1366) - core - with 53.95 GAM estimate
0.000223 10fthf_c + 2.6e-05 2fe2s_c +
  0.000223 2ohph_c + 0.00026 4fe4s_c +
  0.51369 ala__L_c + 0.000223 amet_c +
  0.29579 arg__L_c + 0.24105 asn__L_c +
  0.24105 asp__L_c + 54.1248 atp_c +
  0.000122 bmocogdp_c + 2e-06 btn_c +
  0.005205 ca2_c + 0.005205 cl_c +
  0.000576 coa_c + 2.5e-05 cobalt2_c +
  0.13351 ctp_c + 0.000709 cu2_c +
  0.09158 cys__L_c + 0.026166 datp_c +
  0.027017 dctp_c + 0.027017 dgtp_c +
  0.026166 dttp_c + 0.000223 fad_c +
  0.006715 fe2_c + 0.007808 fe3_c +
  0.26316 gln__L_c + 0.26316 glu__L_c +
  0.61264 gly_c + 0.2151 gtp_c +
  48.6015 h2o_c + 0.094738 his__L_c +
  0.29053 ile__L_c + 0.19519 k_c +
  0.45053 leu__L_c + 0.34316 lys__L_c +
  0.15369 met__L_c + 0.008675 mg2_c +
  0.000223 mlthf_c + 0.000691 mn2_c +
  7e-06 mobd_c + 0.001831 nad_c +
  0.000447 nadp_c + 0.013013 nh4_c +
  0.000323 ni2_c + 0.017868 pe160_c +
  0.054154 pe161_c + 0.18527 phe__L_c +
  0.000223 pheme_c + 0.22106 pro__L_c +
  0.000223 pydx5p_c + 0.000223 ribflv_c
  + 0.21579 ser__L_c + 0.000223 sheme_c
  + 0.004338 so4_c + 0.000223 thf_c +
  0.000223 thmpp_c + 0.25369 thr__L_c +
  0.056843 trp__L_c + 0.1379 tyr__L_c +
  5.5e-05 udcpdp_c + 0.1441 utp_c +
  0.42316 val__L_c + 0.000341 zn2_c +
  0.019456 kdo2lipid4_e +
  0.013894 murein5px4p_p +
  0.045946 pe160_p + 0.02106 pe161_p ->
  53.95 adp_c + 53.95 h_c + 53.9457 pi_c
  + 0.7739 ppi_c

Show previous steps...
```

80 characters per line:

```
surfNet(iJO1366, [], [], [], [], 0, [], 80)
```

```
Rxn #8  BIOMASS_Ec_iJO1366_core_53p95M, Bd: 0 / 1000, E. coli biomass objective function (iJO1366) - core - with 53.95 GAM estimate
0.000223 10fthf_c + 2.6e-05 2fe2s_c + 0.000223 2ohph_c + 0.00026 4fe4s_c +
  0.51369 ala__L_c + 0.000223 amet_c + 0.29579 arg__L_c + 0.24105 asn__L_c +
  0.24105 asp__L_c + 54.1248 atp_c + 0.000122 bmocogdp_c + 2e-06 btn_c +
  0.005205 ca2_c + 0.005205 cl_c + 0.000576 coa_c + 2.5e-05 cobalt2_c +
  0.13351 ctp_c + 0.000709 cu2_c + 0.09158 cys__L_c + 0.026166 datp_c +
  0.027017 dctp_c + 0.027017 dgtp_c + 0.026166 dttp_c + 0.000223 fad_c +
  0.006715 fe2_c + 0.007808 fe3_c + 0.26316 gln__L_c + 0.26316 glu__L_c +
  0.61264 gly_c + 0.2151 gtp_c + 48.6015 h2o_c + 0.094738 his__L_c +
  0.29053 ile__L_c + 0.19519 k_c + 0.45053 leu__L_c + 0.34316 lys__L_c +
  0.15369 met__L_c + 0.008675 mg2_c + 0.000223 mlthf_c + 0.000691 mn2_c +
  7e-06 mobd_c + 0.001831 nad_c + 0.000447 nadp_c + 0.013013 nh4_c +
  0.000323 ni2_c + 0.017868 pe160_c + 0.054154 pe161_c + 0.18527 phe__L_c +
  0.000223 pheme_c + 0.22106 pro__L_c + 0.000223 pydx5p_c + 0.000223 ribflv_c +
  0.21579 ser__L_c + 0.000223 sheme_c + 0.004338 so4_c + 0.000223 thf_c +
  0.000223 thmpp_c + 0.25369 thr__L_c + 0.056843 trp__L_c + 0.1379 tyr__L_c +
  5.5e-05 udcpdp_c + 0.1441 utp_c + 0.42316 val__L_c + 0.000341 zn2_c +
  0.019456 kdo2lipid4_e + 0.013894 murein5px4p_p + 0.045946 pe160_p +
  0.02106 pe161_p -> 53.95 adp_c + 53.95 h_c + 53.9457 pi_c + 0.7739 ppi_c

Show previous steps...
```

REFERENCES

[1] Orth, J. D., Thiele I., and Palsson, B. Ø. What is flux balance analysis? *Nat. Biotechnol., 28*(3), 245–248 (2010).