Metabolic Modelling, Spring 2009, Exercises
24.3.2009

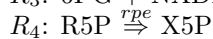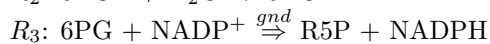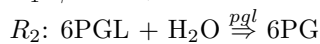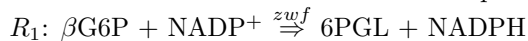*These exercises are to be completed during the exercise session using Matlab. If you manage to complete all the exercises befroe the end of the session, you are free to leave, after you have shown your solutions to the instructor.*

1. Stoichiometric matrix.

   Consider the metabolic model of E. coli pentose phosphate pathway:

   R_1: βG6P + NADP$^+$ $\overset{zwf}{\Rightarrow}$ 6PGL + NADPH

   R_2: 6PGL + H$_2$O $\overset{pgl}{\Rightarrow}$ 6PG

   R_3: 6PG + NADP$^+$ $\overset{gnd}{\Rightarrow}$ R5P + NADPH

   R_4: R5P $\overset{rpe}{\Rightarrow}$ X5P

   R_5: βG6P $\overset{gpi}{\Leftrightarrow}$ βF6P

   R_6: S7P + G3P $\overset{tkt}{\Leftrightarrow}$ R5P + X5P

   R_7: βF6P + E4P $\overset{tkt}{\Leftrightarrow}$ S7P + G3P

   R_8: βF6P + G3P $\overset{tkt}{\Leftrightarrow}$ E4P + X5P
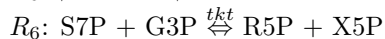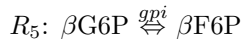
   (a) Enter the stoichiometric matrix of the system to MATLAB.

   (b) Assume the following rate vector for the reactions:

   $$v = (1, 1, 0, -1, -1, 0, 1, 1)$$

   Given these reaction rates and the stochimetric matrix, check and list the metabolites that

   - are accumulating ($\sum_j s_{ij} v_j > 0$),
   - remain in constant concentration,
   - are diminishing

   MATLAB Help > Getting started > Matrices and arrays

   MATLAB Functions by Category > Arithmetic operators

2. From the stoichiometric matrix of the metabolic system it is easy to obtain the adjacency matrices of the reaction graph ($R_{ij} = 1$ iff i and j are neighbors) and the substrate graph. Denote

   - S_{abs} — the matrix that is obtained by taking the absolute value of each item of S
   - $Q = S'_{abs} S_{abs}$ — product of the transpose of S_{abs} with S_{abs}

- $R = (Q > 0)$ — a matrix where R_{ij} is the indicator of $Q_{ij} > 0$

(a) Write a MATLAB function that computes the adjacency matrix of reaction graph R by the procedure described above. Test the function using the example metabolic system.

(b) Explain why the above procedure is correct.

(c) The adjacency matrix of the substrate graph can be obtained from the stoichiometric matrix by an analogous procedure. Derive this procedure and write a MATLAB script to compute the adjacency matrix of the substrate graph.

MATLAB Help > Getting Started > Programming > Scripts and functions

MATLAB Functions: abs()

3. Graph statistics: degree distributions and path length distributions

(a) Write a MATLAB function that computes from the adjacency matrix of a graph its node degree distribution

$$D = [d_1, d_2, \ldots, d_{max}],$$

where d_k denotes the number of nodes in the graph that have degree k. Test your function with the example metabolic system.

MATLAB Functions: sum(),max(),histc()

(b) Write a MATLAB function that computes from the adjacency matrix of a graph its path length distribution. Test your function with the example metabolic system.

MATLAB Functions: graphallshortestpaths(),histc()

4. The file http://www.cs.helsinki.fi/bioinformatiikka/mbi/courses/ 08-09/memo/stryer_generic_model.txt contains the stoichimetrix matrix of a generic (not organism specifc) model of central carbon metabolism.

- Download the stoichiometric matrix into a local file. Then load it into MATLAB either using the Import Data -facility or the MATLAB function *load()*. The file contains the metabolite and reaction names at the end of the file. You will probably need to edit these out in order to load the file correctly.

- Using the functions you have implemented, compute the adjacency matrices of the reaction graph and the substrate graph.

- Compute and plot the node degree distribution and path length distribution of the system.

MATLAB Functions: plot(), loglog()

5. (a) Write a MATLAB function that fits a scale free model $P(k) \approx k^{-\gamma}$ to the node degree distribution of the substrate graph.

Hint: use the MATLAB Function fminsearch() *to optimize the parameter γ. You will need to write a helper function $sse = modelerror(\gamma, k, y)$*

that takes as input the node degrees $k = 1 : kmax$ and returns the sum of squared errors between the model $x^{-\gamma}$ and the observed node degrees $y = (y_1, \ldots, y_{kmax})$. fminsearch *will use a function handle to* modelerror *to obtain the error for different values of γ during optimization.*

(b) Plot the fitted model together with the node degree distribution. Does the scale-free model fit the data well?

MATLAB Functions: loglog()