Metabolic Systems Analysis Workshop

A Step-by-Step Guide to Elementary Flux Mode Analysis

Ross Carlson
Reed Taffs, James Folsom, Kristopher Hunt

Department of Chemical and Biological Engineering
Center for Biofilm Engineering
Montana State University, Bozeman MT
rossc@erc.montana.edu

List of Electronic Files:

pdf:
EFMA_Workshop (presentation files)
EFM algorithm
EFMA_Workbook_Manual
EFMA_Workbook_Manual (completed version)

Carlson et al 2016, Extremophiles. 20:261-274.
Hunt et al 2014. Bioinformatics. 30(11): 1569-1578.
Taffs et al., 2009, BMC Systems Biology. 3:114.
Carlson, 2009, Bioinformatics. 25: 90-97 *and supplemental material.*
Carlson, 2007, Bioinformatics. 23: 1258-1264 and supplemental material.
Carlson et al., 2005, Appl. Environ. Microbiol. 71: 713-720.
Carlson and Srienc, 2004a, Biotechnol. Bioeng. 85(1): 1-19.
Carlson and Srienc, 2004b, Biotechnol. Bioeng. 86(2): 149-162.
Carlson et al., 2002, Biotechnol. Bioeng. 79(2): 121-134.

MS Excel:
EcMatrix (completed)
EcPHB (completed)
EFMA_Workbook_2 (exercise book and completed version)
EFMA_Workbook_3 (exercise book and completed version)
Gfp (completed protein analysis file)
EFMA_Workshop_Aids

Other:
Doubletool.exe (elementary mode analysis program v. 4.9.2)
Meta352_double.exe (elementary mode analysis program v. 3.5.2)
Ec200.txt (Escherichia coli metabolic model input file)
Ec200out.txt & EcMatrix.txt (output files for model Ec200.txt, completed folder)
EcPHB.txt (non-growth E. coli model with PHB synthesis, completed folder)
EcPHBout.txt. & EcPHBMatrix.txt (output files for model EcPHB.txt, completed folder)
FAEc07 folder (contains FluxAnalyzer/CellNetAnalyzer files for Carlson, 2007 model)
Gfp.txt
EFMA_model_diagrams.ppt (ppt files of model illustrations)
SC01.txt (Saccharomyces cerevisiae metabolic model input file)
SCnoX.txt (S. cerevisiae metabolic model input file- no biomass reaction)
AHC.txt (photoautotroph/heterotroph microbial community model input file)
Metatool2efmtool.py (Python script for converting METATOOL input file into an EFMtool compatible file)
Instructions_for_metatool2efmtool (instructions for using Python script above)

Note: the 'Completed folder' folder contains the 'completed versions' of the exercises.

Section 1.0 -Elementary Flux Mode analysis software

METATOOL v 4.9, the elementary flux mode program used with this workshop, was downloaded from: http://pinguin.biologie.uni-jena.de/bioinformatik/networks/metatool/metatool5.0/metatool5.0.html. This website contains software documentation as well as a number of sample metabolic network models. A pdf file written by Schuster, Dandekar, and Fell describing the algorithm used by the METATOOL program has been included in the workshop material (EFM_algorithm.pdf). Some relevant references to the elementary flux mode program METATOOL include:

Von Kamp, A. and Schuster, S. (2006) Metatool 5.0: fast and flexible elementary modes analysis. Bioinformatics 22(15):1930-1931.

Schuster, S., Dandekar, T. and Fell, D. (1999) Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering. TIBTECH, 17, 53-60.

Schuster, S. and Hilgetag, C. (1994) On elementary flux modes in biochemical reaction systems at steady state. J. Biol. Syst. 2, 165-182.

Schuster, S., Hilgetag, C., Woods, J. H. and Fell, D. A. (1996) Elementary modes of functioning in biochemical networks. In: Computation in Cellular and Molecular Biological Systems (Cuthbertson, R., Holcombe,M. and Paton, R., eds), pp. 151-165, World Scientific, Singapore.

Pfeiffer, T., Sanchez-Valdenebro, I.,Nuno, J. C., Montero F. and Schuster, S. (1999) METATOOL: For Studying Metabolic Networks, Bioinformatics 15(3): 251-257.

The models used in this workshop are designed to run on the METATOOL v4.9 executable program named 'doubletool.exe'. The output files from these simulations are easily manipulated with common spreadsheet software like MS Excel. For larger-scale simulations, it is advisable to use other elementary flux mode programs like 'FluxAnalyzer' or its successor 'CellNetAnalyzer'. (http://www.mpi-magdeburg.mpg.de/projects/cna/cna.html). These elementary flux mode programs operate on a MATLAB platform and are better designed to handle large datasets. In addition, MATLAB is very efficient at matrix manipulations and can handle datasets too large for MS Excel. Useful references for these programs include:

Klamt, S., Saez-Rodriguez, J. and Gilles, E.D. (2007) Structural and functional analysis of cellular networks with CellNetAnalyzer. BMC Systems Biology, 1:2. <freely available at BMC Systems Biology>

Klamt, S., Saez-Rodriguez, J., Lindquist, J., Simeoni, L. and Gilles, E.D. (2006) A methodology for the structural and functional analysis of signaling and regulatory networks. BMC Bioinformatics, 7:56. <freely available at BMC Bioinformatics>

Klamt, S., Gagneur, J. and von Kamp, A. (2005) Algorithmic approaches for computing elementary modes in large biochemical reaction networks. IEE Proceedings Systems Biology, 152(4), 249-255.

Klamt, S., Stelling, J., Ginkel, M. and Gilles, E.D. (2003) FluxAnalyzer: exploring structure, pathways, and flux distributions in metabolic networks on interactive flux maps. Bioinformatics 19(2): 261-269.

Section 1.1 -Getting started : the METATOOL input file

First, copy the EFMA_Workshop folder from the workshop CD onto your computer's desktop. The folder contains all needed files.

Elementary flux mode analysis will be performed using the 'doubletool.exe' program and the model input file 'Ec200.txt'. The model represents the chemical reactions of the *E. coli* central metabolism along with reactions that account for metabolite transport into and out of the cell. The control volume for the model is a single cell. The model consists of 44 declared reactions and 46 declared metabolites. The contents of the example input file, 'Ec200.txt,' are shown below in small font, followed by a graphical representation of the reaction network. The example network is taken from Carlson and Srienc, 2004a.

```
-ENZREV
R2r R5r R6r R7r R8r R11r R12r R13r R14r R15r R23r R26r R27r R28r R29r R53r R54r R97r

-ENZIRREV
R1 R3 R4 R9 RR9 R10 R20 R21 R22 R24 R25 R40 R41 R42 R55 R70 R80 R81 R82 R83 R90 R91 R94 R95 R93 R96

-METINT
ATP ADP GLU_6_P FRU_6_P FRU_BIS_P DHAP GA_3P NAD NADH
RIBULOSE_5_P XYL_5_P RIBOSE_5_P SED_7_P ERYTH_4_P PYR PEP CITRATE
OXALO MALATE CoASH ACETYL_CoA FADH FAD AKG ISOCIT ACETATE SUCC
FUMARATE PG LACTATE SUCC_CoA NH3 ETOH FORMATE CO2

-METEXT
ATP_base GLU_ext ETOH_ext ACETATE_ext CO2_ext LACTATE_ext SUCC_ext
NH3_ext FORMATE_ext BIOMASS Oxy_ext PHB

-CAT

R1 :    GLU_ext  + PEP  = GLU_6_P + PYR  .
R2r :   GLU_6_P = FRU_6_P .
R3 :    FRU_6_P + ATP  = FRU_BIS_P + ADP  .
R4 :    FRU_BIS_P = FRU_6_P .
R5r :   FRU_BIS_P = DHAP + GA_3P .
R6r :   GA_3P = DHAP .
R7r :   GA_3P + ADP + NAD  = PG + ATP + NADH .
R8r :   PG = PEP .
R9 :    PEP + ADP = PYR + ATP .
RR9 :   PYR + 2 ATP = PEP + 2 ADP .

R10 :   GLU_6_P + 2 NAD = RIBULOSE_5_P + 2 NADH + CO2 .
R11r :  RIBULOSE_5_P = XYL_5_P .
R12r :  RIBULOSE_5_P = RIBOSE_5_P .
R13r :  RIBOSE_5_P + XYL_5_P = SED_7_P + GA_3P .
R14r :  GA_3P + SED_7_P = ERYTH_4_P + FRU_6_P .
R15r :  ERYTH_4_P + XYL_5_P = GA_3P + FRU_6_P .

R20 :   PYR + CoASH = ACETYL_CoA + FORMATE .
R21 :   PYR  + NAD + CoASH = ACETYL_CoA + CO2 + NADH .
R22 :   OXALO  + ACETYL_CoA  = CITRATE  + CoASH  .
R23r :  CITRATE  = ISOCIT  .
R24 :   ISOCIT  + NAD  = AKG  + NADH  + CO2 .
R25 :   AKG  + NAD + CoASH = NADH + SUCC_CoA  + CO2 .
R26r :  SUCC_CoA + ADP = SUCC + ATP + CoASH .
```

R27r : SUCC + FAD = FUMARATE + FADH .
R28r : FUMARATE = MALATE .
R29r : MALATE + NAD = OXALO + NADH .

R40 : PEP + CO2 = OXALO .
R41 : MALATE + NAD = PYR + NADH + CO2 .
R42 : OXALO + ATP = PEP + ADP + CO2 .

R53r : PYR + NADH = LACTATE + NAD .
R54r : ACETYL_CoA + 2 NADH = ETOH + 2 NAD + CoASH .
R55 : ACETYL_CoA + ADP = ACETATE + CoASH + ATP .

R70 : 4 GLU_6_P + 46 RIBOSE_5_P + 31 ERYTH_4_P + 156 PEP + 237 PYR + 72 ACETYL_CoA + 86 AKG + 139 OXALO + 2921 ATP + 856 NADH + 731 NH3 = BIOMASS + 72 CoASH + 2921 ADP + 856 NAD + 32 CO2 .

R80 : NADH + 2 ADP + Oxy_ext = NAD + 2 ATP .
R81 : FADH + ADP + Oxy_ext = FAD + ATP .
R82 : ATP = ADP + ATP_base .
R83 : NADH + FAD = NAD + FADH .

R90 : ETOH = ETOH_ext .
R91 : ACETATE = ACETATE_ext .
R93 : NH3_ext = NH3 .
R94 : LACTATE = LACTATE_ext .
R95 : SUCC = SUCC_ext .
R96 : FORMATE = FORMATE_ext .
R97r : CO2 = CO2_ext .

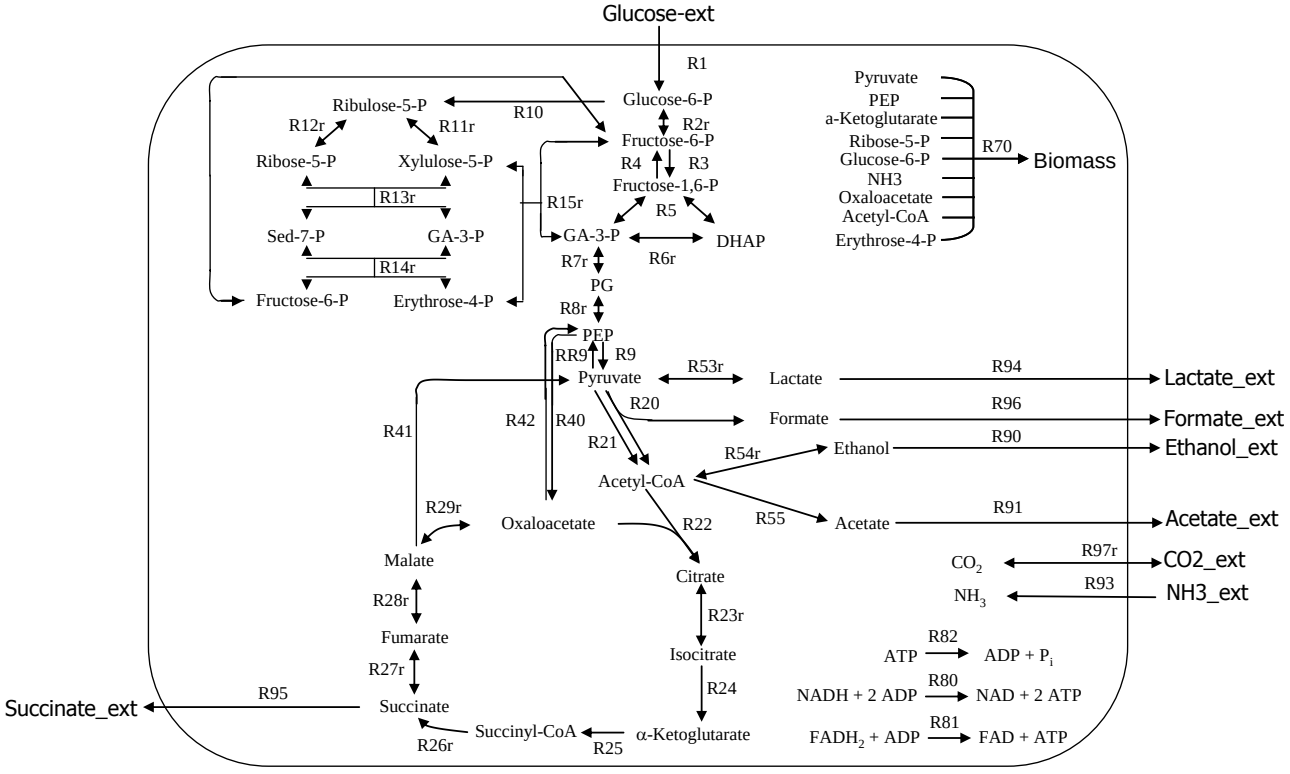

Figure 1. Graphical representation of the *E. coli* central metabolism model Ec200.txt. A modifiable PowerPoint version of this figure can be found in EFMA_model_diagrams.ppt.

The input file requires a number of declarations in a specified order. The first two sections are used to declare reactions. Reversible reactions (declared under the

heading '–ENZREV') may act in either direction (left-to-right or right-to-left), while irreversible reactions (declared under '–ENZIRREV') must act left-to-right. Next, the metabolites in the reactions must be declared as either an internal metabolite (under the heading '-METINT') or as an external metabolite (under the heading '–METEXT'). The elementary flux mode software enforces a steady-state assumption on all internal metabolites. This means each elementary flux mode must produce and consume the same number of each internal metabolite to prevent accumulation. The external metabolites, free of this constraint, serve as sources and sinks. The chemical reactions and metabolite transport steps are defined in the section '-CAT'. All reactions and metabolites must be declared in the appropriate section. The reactions can represent a number of actual processes, for instance the action of a single enzyme (R2r), the action of a series of enzymes (R70), or a physical process like the transport of a metabolite across the cellular membrane (R97r). NOTE: It is often easier to modify existing input files than to recreate the whole file from scratch. This also helps ensure proper file syntax.

For this model, glucose is the sole energy source (electron donor) while glucose and CO_2 are potential carbon sources. O_2 is a potential electron acceptor and a range of fermentation products including acetate, formate, ethanol, and succinate are also considered as potential redox sinks. The model considers biomass synthesis (R70) and maintenance energy production (R82). Both 'BIOMASS and 'ATP_base' are considered 'external metabolites. The biomass reaction (R70) accounts for the drain of central metabolism intermediates that are required for biomass synthesis. This treatment of biomass synthesis reduces the computational burden. Macromolecular composition changes with growth rate so the contribution of these central metabolism intermediates changes as well. The reaction in the example model 'Ec200.txt' represents a 200 minute doubling time; the Excel file 'EFMA_Workbook_Aids' contains a worksheet named 'Biomass_worksheet' which provides a template for generating biomass (or protein) terms for any desired macromolecular composition. The biomass data on this worksheet is from Carlson, 2007.

Important model considerations include:

1) What is the energy (electron donor) source?
   In this example, it is glucose. Depending on the organism being analyzed, many energy sources can be considered. Some possibilities are sunlight, H_2, and acetic acid.
2) What is the electron acceptor?
   For this model, oxygen is a possible electron acceptor, as are a number of fermentation products like ethanol and lactic acid.
3) Is a reaction reversible or irreversible?
   Reaction R94, which accounts for lactic acid secretion, is listed as irreversible in the example model. If reaction R94 is changed to reversible, lactic acid can serve as either a product (reaction runs left to

right) or an energy source (reaction runs right to left). Considering multiple substrates greatly increases the number of possible modes.

Section 1.2 -Running the METATOOL program

The input file needs to be in the same folder as the METATOOL executable program. We are using METATOOL version 4.9.2. The executable file is named 'doubletool.exe'. Double click on 'doubletool.exe'. A DOS window will open and ask for an input filename, type 'Ec200.txt' and press 'return'. When prompted for an output filename, type 'Ec200out.txt' and press return. The program will start running. Depending on the computer configuration, it should not take more than a few seconds to finish the analysis and to create the output file. When it's done, press 'return' to close the DOS window. The output file will be in the same folder as the 'doubletool.exe' program and input file.

Section 1.3 -METATOOL output files

The model 'Ec200.txt' results in 1,661 total modes with 832 modes synthesizing biomass.

The METATOOL output file contains a number of useful features. The software developers have described many of these features on their websites:

http://pinguin.biologie.uni-jena.de/bioinformatik/networks/metatool/metatool5.0/metatool5.0.html

http://pinguin.biologie.uni-jena.de/bioinformatik/networks/metatool//metatool.html

Briefly a few items of note:

METATOOL OUTPUT (double) Version 4.9.2 C:\My Docs 9100\Presentations\INL_2007\doubletool.exe

INPUT FILE: Ec200.txt

INTERNAL METABOLITES: 35
EXTERNAL METABOLITES: 11
REACTIONS: 44

12 int NAD
12 int NADH
11 int ATP ….

The first portion of the file summarizes characteristics of the input model and is useful for double checking declarations like the number and each type of metabolite and the number and type of reactions. The next section states whether each metabolite is internal or external and how many reactions include that metabolite.

STOICHIOMETRIC MATRIX

```
 matrix dimension r35 x c44
   0   -1    0    1   -2    0    0    0    0    0    0    0    0   -1    1 -2921    2    1   -1    0    0    0    0    0    0    0    0    0    0    0    1
0    0    0    0    0    0    0    1    0    0    0    0    0    0
   0    1    0   -1    2    0    0    0    0    0    0 …
```

The input model is converted into a stoichiometric matrix, with each row corresponding to an internal metabolite and each column corresponding to a reaction. The stoichiometry matrix contains all of the information from the reactions declaration in a mathematical format.

SUBSETS OF REACTIONS

…

REDUCED SYSTEM with 23 branch point metabolites in 32 reactions (columns)

. . .

Before solving the system, the stoichiometric matrix is simplified by clustering reactions that always occur together. This reduces the computational burden.

CONVEX BASIS

```
 matrix dimension r392 x c44
 0.2  0.2    0    0    0    0    0  0.2  0.2  0.2  0.2  0.2    0    0    0    0    1    0  2.4    0    0    0    0    0  0.2    0  0.2  0.2 -0.2  0.4  0.4    0    0
0    0    0  0.2  0.2    0    0    0    0    0  0.4
…
```

Elementary flux mode analysis is based on a branch of mathematics known as convex analysis (see references in Section 1.0). Basically, the input model is used to define a multidimensional convex cone which contains all possible model solutions. The convex basis is a mathematical description of that cone. All convex basis vectors correspond to elementary flux modes but not all elementary modes are part of the convex basis. The convex basis data is presented in three different formats. We won't be doing any work with this data.

Following the convex basis data are three sections containing elementary flux mode data. We will be focusing on the elementary flux mode data. The data is represented in three different formats.

ELEMENTARY MODES

```
 matrix dimension r1661 x c44
0.166667    0    0    0    0  0.5    0    0    0    0    0    0    0    0    0    0    0    1 1.16667    1    0    0    0 0.166667    0    0 -
0.333333    0    0 0.166667 0.166667 0.333333 0.166667 0.166667 0.166667 0.166667    0    0    0    0    0 0.166667    0  0.5
0.166667    0    0    0    0  0.5    0
```

. . .

The first representation of the elementary flux modes output data is the most useful. In this case all 1,661 modes are listed as row vectors in a matrix. Each column corresponds to one of the 44 reactions in the input file. If a mode utilizes a particular reaction, it has a non-zero coefficient in that column.

enzymes
in () indicates # of enzymes used by the elementary mode
1: (16) (0.166667 R1) (0.5 R10) R81 (1.16667 R82) R83 (0.166667 R94) (-0.333333 R2r) (0.166667 R7r) (0.166667 R8r) (0.333333 R11r) (0.166667 R12r) (0.166667 R13r) (0.166667 R14r) (0.166667 R15r) (0.166667 R53r) (0.5 R97r) irreversible

2: (15) (0.166667 R1) (0.5 R10) R80 (2.16667 R82) (0.166667 R94) (-0.333333 R2r) (0.166667 R7r) (0.166667 R8r) (0.333333 R11r) (0.166667 R12r) (0.166667 R13r) (0.166667 R14r) (0.166667 R15r) (0.166667 R53r) (0.5 R97r) irreversible
3: (17) (0.166667 R1) (0.5 R10) (1.16667 R40) …

The second representation of the data lists only those reactions utilized by a mode. The first number is the mode number, *e.g*. '1,' '2,' and '3' in the example data shown. The second number, in parenthesis, is the number of reactions utilized in that mode. The next string of numbers lists the relative flux through each utilized reaction. This is the same data as in the first output representation, but the second data format is less suitable for exporting to other programs.

overall reaction

1: GLU_ext + 6 Oxy_ext = 7 ATP_base + 3 CO2_ext + LACTATE_ext
2: GLU_ext + 6 Oxy_ext = 13 ATP_base + 3 CO2_ext + LACTATE_ext
3: GLU_ext + 6 Oxy_ext = 3 CO2_ext + LACTATE_ext

The third representation of the output is a listing of just the external metabolites involved in the mode. It represents the overall transformation of the mode. This data corresponds with the data in formats one and two, however the stoichiometry has been normalized so the smallest coefficient is equal to one.

Section 1.4.1 -Transferring METATOOL output data to Excel 2007

When run with Metatool (doubletool.exe), the model 'Ec200.txt' results in 1,661 total modes with 832 modes synthesizing biomass.

Copy the matrix format output data (first format) and paste it into a new text window (under the 'Start' menu in lower left corner of computer, select 'Accessories,' and then pick 'Notepad' – notepad is a text editor). Save the file as 'EcMatrix.txt' and close it.

Start MS Excel (all instructions are based on Excel 2007, similar functions are available with Excel 2003).

In the 'Get External Data' section of the 'Data' tab select 'from text', and browse to the 'EFMA_Workshop' folder from desktop and open EcMatrix.txt
The 'Text Import Wizard' window will appear.
Select 'Delimited' as the text file type and click 'Next'.
Select 'Space' as the text delimiter.
Click 'Finish', there is no need to use the 3rd Import Wizard page, and indicate the desired insertion point.

The Metatool output file should now be open inside an Excel worksheet and only one number should reside in each cell. The numbers in each row are the coefficients for each reaction in a single elementary flux mode.

The import step will not always properly align the data. The next step is to run an Excel macro to left align the data. This is important. We need each entry in a column to refer to the same model reaction.

On the 'View' tab, to the far right click macros, and type 'DelEmptyMoveLeft' into the top box and push the create button. The Visual Basic Window will appear with a new module already open.
Copy and paste the following macro (found in the Excel 'EFMA_Workbook_Aids' file on the 'Workbook_formulas' worksheet listed under Section 1.4.1) into the module window. Each line should appear only once.

```
Sub DelEmptyMoveLeft()
    Cells.Select
    Selection.SpecialCells(xlCellTypeBlanks).Select
    Selection.Delete Shift:=xlToLeft
End Sub
```

This macro will align the data along the left margin by deleting the empty cells and shifting the data to the left. Close the visual basic window.

To run the macro, return to the open Excel worksheet titled 'EcMatrix'.
Highlight all data by clicking the gray cell in the upper-left most part of the worksheet.

Press 'Alt' and the 'F8' key at the same time.
A 'Macros' window will appear.
Select 'DelEmptyMoveLeft' and click run.
The Macro will delete the empty cells and shift the data to the left.

Section 1.4.2 -Inserting column (reaction) identifiers

Each column represents a reaction from the input file model.

Insert a blank row on top of the 'EcMatrix' data set (left click the gray cell with the number 1, right click, select 'Insert').
Open the Excel workbook titled 'EFMA_Workbook_Aids'. The 'Workbook_formulas' worksheet has a row pre-labeled with the reaction names of each column.
Copy and paste this row into the empty row on 'EcMatrix'.

The columns correspond with the order of the declared reactions on the input file starting with the irreversible reactions followed by the reversible (note: the input file uses the opposite convention and lists reversible reactions first then the irreversible). Important transport reactions have the associated metabolite listed next to the reaction. For instance, R1 is the glucose uptake reaction and the abbreviation (gluc) is included in the label.

Section 1.4.3 -Insert row (mode) numbers

Insert a blank column to the left of the data (left click the gray cell with the letter 'A' in it, right click, select 'Insert').

Enter '1', '2', and '3' into the first three cells in the new column. Select these cells (by click-and-drag) and populate the rest of the column by double-clicking on the lower-right corner of the selected region.

Save the data as an Excel file (otherwise the worksheet will be saved as a text (txt) file and the formatting will be lost.):

Select 'Save As' under the 'Microsoft Office' pull-down menu, and click on 'Excel Workbook'. The default filename is fine.

Section 1.5 -Sorting exercise

Recall that each row corresponds to one elementary mode and each column represents the flux through a model reaction. Highlight all data by clicking the gray square in the upper left corner of the worksheet. Under the 'Data' tab select 'Sort'. A 'Sort' window will appear. First, check the box next to 'My data has headers' found at the top right of the window – this option identifies the first row as labels, excluding it from the sort. Use the 'Sort by' pull-down menu to select the column you want to sort the data by – to sort for biomass production select 'R70' – this reaction accounts for the synthesis of biomass. Next choose 'Largest to Smallest' from the 'Order' pull-down menu and click 'OK'. The matrix will then be sorted for modes which produce biomass, listing the mode with the largest coefficient for reaction R70 first. The Mode number in column 'A' will sort with the data and permits cross referencing with the original Metatool output.

EXERCISE:

How many modes produce biomass?

How many modes co-produce biomass and ethanol?

Section 1.6 -Yield and inverse yield parameters for mode data

It is often useful to gauge the relative fitness of modes based on production of biomass, ethanol, succinate, *etc*. The ratio of fluxes through output and uptake reactions provides an appropriate metric for this sort of comparison, termed a yield.

For instance: reaction R70 corresponds to biomass production and reaction R1 corresponds to glucose uptake. The ratio R70 / R1 gives the 'biomass yield', or in other words, how much biomass is produced for an amount of consumed glucose. We can choose to express this yield with a variety of units. Here, we will look at the carbon moles (Cmol) of biomass produced per Cmol of glucose consumed. The yield will then represent the fraction of carbon atoms in glucose that end up in the biomass. The carbon atoms that are not incorporated into biomass are released in by-products.

Yields can be calculated for any number of product / substrate relationships. The Excel file 'EFMA_Workbook_Aids' contains formulas for calculating biomass yields on glucose (Cmol X/Cmol glucose), biomass yields on oxygen (Cmol X/mol O_2) as well as a number of other yields. The formulas start in cell 'AU50' of the 'Workbook_formulas' worksheet.

Copy and paste these yield formulas into your 'EcMatrix' worksheet. To work properly, the formulas must be pasted into cell 'AU1'.

Double check the formulas by clicking on cell 'AU2'. The formula should be the ratio of R70 / R1 with the corresponding conversion factors.

Conversion of mode output data into Cmol yields requires that we multiply the R70 cell by 2,652. The 'Ec200.txt' model's 200 minute doubling time biomass term represents 2,652 atoms of carbon (Carlson and Srienc, 2004a). For each molecule of glucose, there are 6 carbon atoms, so the R1 cells (representing uptake of a single glucose molecule) need to be multiplied by 6 to convert from units of moles glucose to Cmol glucose. The formulas contain an Excel logical statement, for instance: =IF(NOT(Q51=0),Q51*2652/(B51*6),0). This states that if cell Q51 is not equal to 0, then Excel should calculate the ratio of Q51*2652 (biomass coefficient times number of carbon atoms in biomass term) divided by B51*6 (the glucose coefficient times 6 Cmol per mole glucose). If Q51 does equal zero, Excel will put a '0' in the cell. The logical statement cleans up the results, because otherwise some cells would try to divide by zero, returning errors.

Highlight the bottom row of formulas, grab the lower right corner, and pull down to fill the cells corresponding to remaining modes. You now have multiple yield data for all 1,661 modes.

NOTE: The variable 'Oxy' in the 'Ec200.txt' model represents 0.5 O_2 molecules. The provided yield formulas account for this conversion, so the output has units of Cmol X/mol O_2.

Section 1.7 -Plotting useful data

The yield data can be plotted to determine useful mode relationships. As an example, we will look at a method for determining the optimal production of biomass as a function of oxygen availability. The method is analogous to a financial analysis where we will minimize the substrate costs (both glucose and O_2) to produce 1 Cmol of biomass. This technique is equivalent to popular linear programming (LP) methods. However in addition to finding optimal solutions, we also develop a mathematical basis for describing these results as a simple linear equation and also find all sub-optimal solutions.

In the previous exercise, we calculated the biomass yield on glucose and on O_2. We also calculated the inverse yields. The inverse yields have units of Cmol

glucose / Cmol X or mol O_2 / Cmol X. These inverse yields are convenient because they can be interpreted as the substrate cost to produce 1 Cmol of biomass. Every inverse yield represents the same 1 Cmol basis, so they can be compared directly. Biological organisms often operate under conditions of limited nutrients so producing biomass or some other product efficiently would likely be competitive from an evolutionary standpoint.

Plot the inverse biomass yields on glucose and oxygen by selecting both columns and selecting 'Scatter with only Markers' from the 'Scatter' menu in the 'Charts' section of the 'Insert' tab.

In the 'Labels' section of the 'Chart Tools Layout' tab, choose 'Above Chart' from the 'Chart Title' pull-down menu. An appropriate title is Optimal Biomass Synthesis. Label the x axis Cmol glucose / Cmol X, and the y axis mol O2 / Cmol X, using the nearby 'Axis Titles' pull-down.

I usually select the legend and press delete to simplify the plot.

Right-click an empty part of the chart and select 'Move chart'. Choose 'new sheet' and name it 'Biomass_GlcVO2'.

The plot contains as X and Y coordinates the glucose and oxygen costs, respectively, to synthesize 1 Cmol of biomass. Every point represents the same amount of biomass, so cost comparisons can be made directly. We are interested in low cost strategies for making biomass so we will focus on the points near the origin. Right-click on the X-axis and choose 'Format Axis'. Under 'Axis Options', set the minimum and maximum to 'Fixed' values of 1 and 3, respectively. Similarly, set the Y-axis and range from 0 to 0.25.

EXERCISE: Identify the biomass synthesizing mode with the smallest glucose cost per Cmol of biomass. **(answer: approx. 1.2187, 0.22923)**
Identify the anaerobic biomass synthesizing mode with the smallest glucose cost per Cmol of biomass. **(answer: approx. 2.9887, 0)**

Section 1.8 -Minimization envelope

We have found the least expensive strategies for synthesizing biomass under oxygen sufficiency and anaerobic conditions – what about conditions between these two extremes? Using the 'Line' tool from the 'Shapes' button in the 'Illustrations' section of the 'Insert' tab, draw a line between the least expensive modes for oxygen sufficiency and anaerobic conditions. For conditions between these two extremes, points closer to the origin than this line represent less expensive biomass synthesis than linear combinations of just these two modes. Additional line segments can be used to identify two intermediary modes which minimize the combined oxygen and glucose costs for culturing conditions between oxygen sufficient and anaerobic.

EXERCISE: Identify the two additional modes lying on the minimization envelope.
answer: The approximate coordinates of the two additional modes are from left to right: (1.4500, 0.17415) (1.75424, 0.12344)
Note that the mode with the approximate coordinates (1.6756, 0.1365) lies exactly on the line segment between the two points on the minimization envelope.

Section 1.9 -Enzyme usage analysis

Using the inverse yield coordinates, locate the minimization envelope modes on the 'EcMatrix' data worksheet and copy those four modes into a new worksheet. Also paste the reaction order reference row above the mode data (to create a new worksheet, click the new worksheet button at the bottom of the workbook).

Paste or sort the modes so that the oxygen sufficient mode (MS1) is first and the anaerobic mode (MS4) is on the bottom. This can be accomplished by sorting the modes so the inverse oxygen yield (oxygen cost) decreases as you move down the column. Scroll through the reactions listed in the columns. Which reactions change as a function of oxygen stress (recall that smaller inverse oxygen yields correspond to limited oxygen availability)?

Examples:

R20-R21, R25 thru R29, R55, R80-R81, R90, R91

-Notice how optimal modes initially use R21 (the pyruvate dehydrogenase complex, PDHc) to convert pyruvate into acetyl-CoA and NADH, but with decreasing oxygen availability shift to R20 (the pyruvate formate-lyase enzyme, Pfl) which converts pyruvate into acetyl-CoA and formate. The use of Pfl results in the production and secretion of formate (R96).

-The oxygen sufficient case uses R25-R29: cyclic, oxidative flux through the TCA cycle. During oxygen limitation or anaerobicity, the modes no longer use these enzymes; instead they adopt a branched flux through the TCA cycle, necessary to support biomass synthesis.

-With the onset of oxygen limitation, the cells begin secreting acetate (R91). As this stress intensifies, acetate and formate will be secreted (R91 and R96). At the most severe levels, the by-products are acetate, formate, and ethanol (R91, R96, and R90).

The large number of alternative modes near the minimization envelope highlights the robustness of the network. There are many other pathways which use slightly different enzyme combinations that are also very efficient.

Section 1.10 -Generating three-dimensional representations of optimal fluxes

The modes give the stoichiometric relationship between enzyme fluxes in a pathway. These can be converted into biologically relevant rates. To accomplish this, it is useful to know that *E. coli* is approximately 50% carbon on a dry mass basis (references in Carlson and Srienc, 2004b). For any specific growth rate, the specific rate of carbon incorporation into biomass can be calculated as in the following example:

$$\mu = \left(\frac{0.1}{\text{hr}}\right) = \left(\frac{0.1\,\text{g cdw}}{\text{hr}\times\text{g cdw}}\right)\left(\frac{0.5\,\text{g carbon}}{1\ \text{g cdw}}\right)\left(\frac{1\,\text{Cmol}}{12\,\text{g carbon}}\right) = \frac{0.00417\,\text{Cmol X}}{\text{hr}\times\text{g cdw}}$$

The inverse biomass yields on glucose or oxygen can then be used to convert this rate into either a specific glucose consumption rate or a specific oxygen consumption rate, respectively.

For the oxygen sufficient case (MS1), the inverse biomass/glucose yield is:

$$\frac{1}{Y_{x/glc}} = \frac{1.219\,\text{Cmol glucose}}{\text{Cmol X}}$$

For a specific growth rate of 0.1 hr^{-1}, the specific glucose consumption rate for biomass synthesis is:

$$\left(\frac{0.00417\,\text{Cmol X}}{\text{hr}\times\text{g cdw}}\right)\left(\frac{1.219\,\text{Cmol glucose}}{\text{Cmol X}}\right) = \frac{0.00508\,\text{Cmol glucose}}{\text{hr}\times\text{g cdw}}$$

The yield was used to convert specific growth rate to specific glucose consumption rate. The same calculation can be done with the inverse oxygen yield. These rates are for biomass synthesis only and do not yet include ATP required for maintenance processes.

EXERCISE: Open 'EFMA_Workbook_Aids', select the 'Workbook_formulas' worksheet and copy and paste the templates for the 3-D plot into the 'EcMatrix' worksheet listing the four most efficient biomass modes (Section 1.9). The template is 75% complete; fill in the missing data. The converted specific growth rate, in units of $\frac{\text{Cmol X}}{gCDW\times hr}$, is given in the template. Use the inverse MS2 biomass yields for glucose and oxygen to fill in the missing data. After the missing data is calculated, plot the iso-metabolic state lines with a scatter plot. The X and Y axis data will be the specific glucose uptake rate and the specific oxygen uptake rate respectively. Next, plot the iso-growth rate lines on the same plot. The plot will contain 3 dimensions of data: the specific glucose uptake rate, the specific oxygen uptake rate, and the specific growth rate.

NOTE: The exercise assumes the same inverse yields for biomass produced at all growth rates. This is a reasonable assumption although the yields do vary slightly (see Carlson and Srienc, 2004b). The analysis considers only biomass synthesis. It does not consider maintenance energy requirements, which will be covered in Section 1.11.

EXERCISE: We know which 4 modes permit the most efficient conversion of glucose into biomass under varying conditions of oxygen availability. Which enzymes are not needed in these pathways? Sort the data and identify knockout targets that, if removed, would force the cells to adopt efficient strategies. This technique has been used successfully to generate efficient-growth *E. coli* strains (Carlson and Srienc, 2004a, Trinh *et al.,* 2006).

(answer: R10, R41, R83 R94, R95)

Section 1.11 -Integrating experimental substrate uptake data with models to generate detailed flux maps

We have examined a graphical two-parameter optimization technique for biomass synthesis under varying levels of oxygen limitation. The same technique can be applied to pathways that produce ATP. The ATP can be used for maintenance energy requirements like maintaining ion gradients, repairing macromolecules, and other cellular processes.

EXERCISE: Plot the ATP producing modes in the Excel worksheet 'EcMatrix' to identify the most efficient ATP production under varying levels of oxygen limitation. Hints: The most efficient ATP modes do not make biomass and, like the biomass case, there are four modes that most efficiently span all conditions between oxygen sufficiency and anaerobic conditions.

The four most efficient modes for synthesizing ATP under varying levels of oxygen stress are given in 'EFMA_Workbook_2'. The associated glucose and oxygen yields are also presented, along with a 3 dimensional representation of the optimal specific glucose uptake rates, specific oxygen uptake rates, and specific ATP production rates. The data processing mirrored the biomass analysis, except the maintenance energy reaction (R82) was utilized instead of the biomass reaction (R70).

Cells have both anabolic (biomass synthesis) and catabolic (ATP production) fluxes. We are interested in partitioning the cellular fluxes into these two categories, but first we need to quantify the substrate flux into the cell. The following data was obtained from a variety of *E. coli* strains using a literature search:

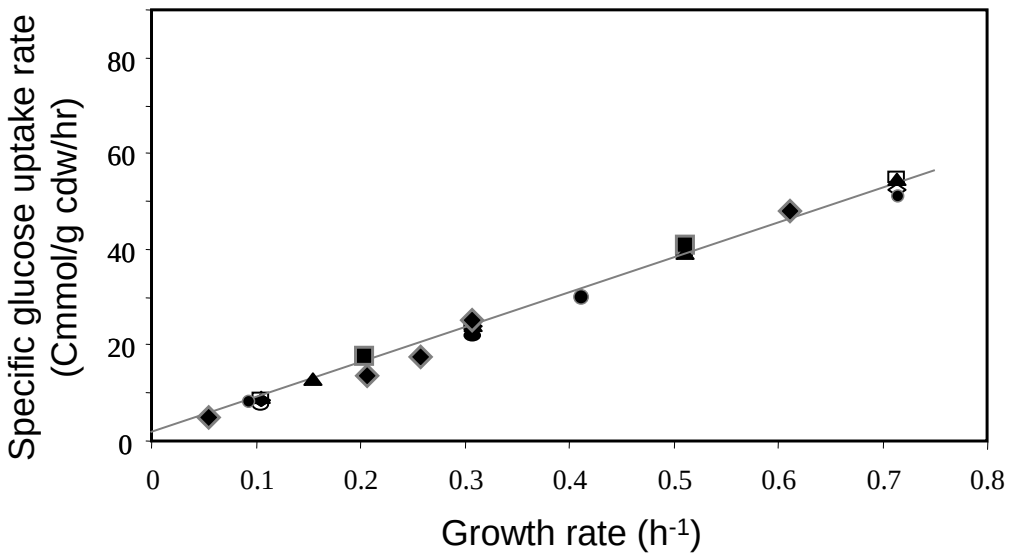

Figure 2. Specific glucose uptake rate as a function of *E. coli* specific growth rate. The experimental data is for oxygen sufficient growth conditions. References for data can be found in Carlson and Srienc, 2004b. The line equation is: $q_{glc} = 72.84*u + 1.02$, with q_{glc} having units of $\frac{Cmmol}{gCDW \times hr}$ and specific growth rate u having units of hr^{-1}.

The data in Figure 2 show a linear relationship between glucose uptake rates and growth rate. This data fixes the total specific glucose uptake flux. We have calculated the minimum specific glucose uptake flux for biomass synthesis as a function of growth rate in Section 1.10. This flux is less than the actual flux of glucose into the cell because, in addition to synthesizing biomass, cells produce ATP for other cellular processes.

'EFMA_Workbook_2', worksheet 'Maintenance_Energy' has templates for calculating the ATP requirements as a function of growth rate. The glucose uptake data in these graphs all assume oxygen sufficient conditions (MS1).

EXERCISE: Calculate the maintenance energy ATP requirements as a function of growth rate, assuming optimal energy production and oxygen sufficient conditions (MS1). Plot this data in 'EFMA_Workbook_2', worksheet 'Maintenance_Energy' and determine the equation relating maintenance energy requirements to dilution rate (D). The exercise requires that the data in the red boxes be completed.

answer: qATP = 95.355D + 4.4197, mmol ATP/ g cdw/ hr

We would like to extend these results to other culturing conditions that are identical except for the availability of oxygen(same temperature, same pH, same salt concentrations, *etc*). If we assume that the maintenance energy

requirements are only a function of growth rate under this constraint, we can extend the results to the other energy producing modes MS2-MS4. Using this assumption, we can now predict a continuous operating solution space comprised of the most efficient cellular metabolisms for any growth rate and any level of oxygen availability.

Section 1.12.1 -Constructing continuous operating solution space

For each growth rate, we know the rate of biomass production, so we can calculate the rates for each reaction in the biomass modes. These rates include measurable fluxes like glucose consumption rates and by-product secretion rates. Likewise, for every growth rate, we can calculate the rates of glucose consumption and by-product secretion associated with energy production. Add these two components together to approximate the overall growth metabolism.

As an example, the anaerobic ATP mode (MS4) is summarized below.

1 glucose = 3 ATP + 1 ethanol + 1 acetate + 2 formate

With a growth rate of 0.3 hr^{-1}, *E. coli* requires 33 $\frac{mmol\ ATP}{gCDW \times hr}$ (see 'Workbook_2EFMA', 'Maintenance_energy' worksheet). To produce this ATP using the specified mode, the cell will consume 11 $\frac{mmol\ glucose}{gCDW \times hr}$ and produce 11 $\frac{mmol\ acetate}{gCDW \times hr}$, 11 $\frac{mmol\ ethanol}{gCDW \times hr}$ and 22 $\frac{mmol\ formate}{gCDW \times hr}$. This represents only maintenance energy. There would also be the biomass synthesis contribution. Efficient anaerobic biomass production would be similarly processed and added to these results to represent overall metabolism.

EXERCISE: The worksheet 'Overall_metabolism' (found in 'EFMA_Workbook_2.xls') calculates the overall cellular metabolism by adding the contributions from biomass synthesis and maintenance energy production. The worksheet lists the MS1-MS4 biomass and ATP producing pathways along with associated yields and includes a 3-dimensional plot for optimal growth. The worksheet has all data except the data for the MS2 biomass section and the MS3 ATP section. Fill in the missing data.

Note: This exercise simplifies biomass production by using the results from one biomass composition to calculate the fluxes for a range of growth rates. While the mode properties like glucose and oxygen yields are very similar at all growth rates, there are differences in biomass fluxes based on the required amounts of macromolecules (like proteins or mRNA).

Section 1.12.2 -Scaling factor calculations for predicting enzyme flux

Fluxes for optimal growth through all enzymes in the model can now be predicted for any growth rate and any level of oxygen availability. 'Workbook_2EFMA', worksheet 'Overall_metabolism' lists the specific uptake or specific secretion rates of external metabolites ($q_{compound}$). The fluxes through all internal reactions can also be predicted using simple scaling factors with the appropriate units. The scaling factors are calculated on the worksheet 'Overall_metabolism' (S_X or S_{ATP}) by dividing either the biomass or the ATP glucose consumption rate (in $\frac{mmol}{gCDW \times hr}$, NOT $\frac{Cmmol}{gCDW \times hr}$) by the elementary mode coefficient for glucose (reaction R1 in the model). This scaling factor converts mode coefficients into biologically relevant fluxes with the units of $\frac{mmol}{gCDW \times hr}$.

An example case is considered on the worksheet 'Scaling_factor'. Consider the flux around the central metabolite acetyl-CoA (see slide 1 in EFMA_model_diagrams.ppt). The model 'Ec200.txt' has two reactions that produce acetyl-CoA: R20 (pyruvate formate-lyase) and R21 (pyruvate dehydrogenase complex). The model has four reactions that consume acetyl-CoA: R22 (citrate synthase), R54 (acetaldehyde + alcohol dehydrogenase), R55 (phosphotransacetylase + acetate kinase) and R70 (biomass synthesis requirements for acetyl-CoA). The exercise will examine predicted fluxes through these reactions for a growth rate of μ= 0.2 hr^{-1} as a function of oxygen availability.

EXERCISE: The scaling factors and the coefficients for the biomass and ATP producing modes are listed for MS1-MS4. The biologically relevant rates are also calculated for all of the reactions except R22. Calculate the missing data for the biomass and ATP modes separately and then sum the rates to get the overall flux through reaction R22. The rates have units of $\frac{mmol}{gCDW \times hr}$, not $\frac{Cmmol}{gCDW \times hr}$. This type of analysis can be done for any metabolite or enzyme in the model.

Section 1.12.3 -The lever rule and the continuous solution space

Sections 1.10 thru 1.12.2 calculated fluxes at defined points corresponding to a given growth rate and a given metabolic state (specified by oxygen availability). The flux values for points in between these defined nodes can be explicitly defined using the classic lever rule.

Example calculation: acetate secretion rates are explicitly listed for a culture growing at MS2 with a μ = 0.2 hr^{-1} and μ = 0.3 hr^{-1}. If a culture is growing at μ = 0.27 hr^{-1} and conditions consistent with MS2, what is the specific acetate secretion rate?

From 'EFMA_Workbook_2', worksheet 'Overall_metabolism', we have:

$q_{acetate,MS2}(\mu=0.2\ hr^{-1}) = 10.2\ \frac{Cmmol\ Acetate}{gCDW \times hr}$

$q_{acetate,MS2}(\mu=0.3\ hr^{-1}) = 14.6\ \frac{Cmmol\ Acetate}{gCDW \times hr}$

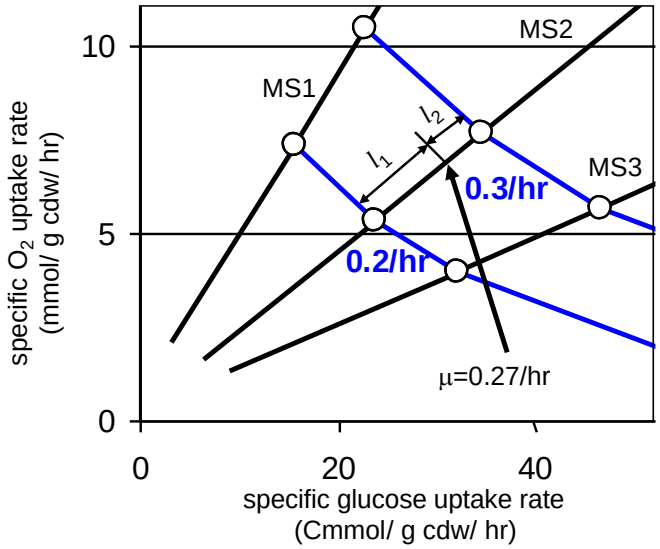

Figure 3. Illustration of distances for lever rule calculation.

$$q_{ac,MS2}(\mu=0.27hr^{-1}) = q_{ac,MS2}(\mu=0.2hr^{-1})\left(1-\frac{l_1}{l_1+l_2}\right) + q_{ac,MS2}(\mu=0.3hr^{-1})\left(1-\frac{l_2}{l_1+l_2}\right)$$

$$q_{ac,MS2}(\mu=0.27hr^{-1}) = q_{ac,MS2}(\mu=0.2hr^{-1})(0.3) + q_{ac,MS2}(\mu=0.3hr^{-1})(0.7)$$

$$q_{acetate,MS2}(\mu=0.27hr^{-1}) = 13.28\ \frac{Cmmol\ acetate}{gCDW \times hr}$$

The same lever rule technique is used to calculate flux states at environmental conditions not defined by an exact metabolic state. For these cases, the lever rule is applied to the two closest metabolic states. See Carlson and Srienc, 2004b for more examples.

Section 1.13.1 -Pathway investment analysis

Microorganisms often reside in environments with scarce resources. Different metabolic pathways require different amounts of resources to synthesize the associated enzymes. We will analyze a method of calculating pathway synthesis requirements and integrating this data with earlier metabolic yield data to describe each elementary mode using multiple fitness metrics.

This section defines two different costs associated with microbial growth. The first is the 'operating cost'. This is the inverse yield we examined previously, and it represents the amount of substrate required to produce either 1 mol ATP or 1

Cmol biomass. The second type of cost is the 'investment cost'. It represents the investment of resources like carbon, nitrogen, and sulfur required to construct the enzymes in an elementary mode. This analysis is analogous to classical factory design theory where designers consider both operating costs (raw materials, labor, and energy) and investment costs (computers, machinery, assembly lines, *etc*). The two extreme factory design cases are 'low investment cost-high operating cost' (inexpensive, low-tech, low-efficiency machinery, high labor and energy costs) or 'high investment cost-low operating cost' (expensive, efficient, automated machinery, low labor and energy costs). Most factories fall somewhere between these two extremes depending on factors like the regional cost of materials, labor, energy, taxes, *etc*.

Section 1.13.2 -Determining enzyme elemental requirements

EXERCISE: The EFMA_Workshop folder contains a text file listing the amino acid sequence for the reporter protein green fluorescence protein (gfp).

Open 'protein_worksheet' in 'Workbook_Aids.xlsx'.

Select 'From Text' in the 'Get External Data' section of the 'Data' tab. Navigate to the workshop folder and select 'gfp.txt'.

Click 'Finish' and choose the top cell within the bordered region.

Elemental composition for all amino acid strings within the bordered region will now be calculated. To add extra cells to the calculation for very long amino acid sequences, edit cell B1.

The protein counting template returns the elemental requirements for each of the amino acids necessary to construct the complete protein as well as the total number of amino acids found in the protein.

Section 1.13.3 -Determining pathway investment costs

The investment costs for the reactions in the sample model 'Ec200.txt' are presented in Excel file 'EFMA_Workbook_3', worksheet 'Rxn_Gene_Invest'. For each model reaction, the associated enzymes and subunit compositions are listed. This information, along with the amino acid sequence of each subunit, was used to calculate the total investment requirement of carbon, nitrogen, sulfur, and amino acids as well as the length of the DNA coding sequence for each reaction. The DNA coding sequence length was calculated by multiplying the protein amino acid count by three (1 amino acid = 1 codon = 3 nucleotides).

The five investment costs can be calculated for each elementary flux mode using linear algebra. First, some assumptions have to be made regarding the relationship between metabolite flux and enzyme concentration. As a starting

point, we will assume the flux through elementary flux modes behaves according to saturation type kinetics - analogous to Monod (cell growth) and Michaelis-Menten (enzyme) kinetics. Saturation kinetics can be simplified to a first order approximation for low substrate concentrations (relative to a half-saturation constant) and to a zeroth order approximation when the substrate concentration is very high (relative to the same constant). The presented example assumes first order kinetics, where the substrate concentration is so low that the rate of substrate hitting the cell is limiting flux. Under this assumption, the enzyme concentrations are not limiting and only a minimal concentration of each enzyme is required. See Carlson, 2007 for a more in-depth discussion. To approximate this case, we will assume that the molar ratio of any two enzymes in an elementary mode is equal to one.

In order to translate these assumptions into practice, we must first convert the METATOOL output data into a binary matrix.

- Copy and paste the processed 'EcMatrix' biomass mode data into the worksheet 'Binary_Matrix' found in 'EFMA_Workbook_3'. Be sure to include the yield and inverse yield columns with this data.
- Highlight the cells containing the logical formula on the right side of the sheet and fill downward until all modes have been converted into a binary format. The template formula uses a logical 'IF' statement (=IF(NOT(B9=0),1,0)) to convert the mode coefficients into either a one or a zero. A one corresponds to the reaction being utilized (with either forward or reverse flux) and a zero corresponds with the reaction not being utilized.
- Copy and paste special (values) the yield and inverse yield columns to the right of the binary matrix (paste as values, otherwise the formulas will recalculate the yields based on the binary transformation).

Now that we have the matrix converted to a binary matrix, we will determine the first order investment costs for each mode using linear algebra. This is accomplished by multiplying the bit-masked mode matrix by the investment column vectors.

- Copy the binary mode data along with the yield data and paste it into the 'Matrix_Multiply' worksheet found in 'EFMA_Workbook_3'.
- Drag the cells containing the formula down the length of the data.

The template calculates the carbon, nitrogen, sulfur, amino acid, and DNA nucleotide investment costs for each of the elementary modes. We now have an extensive set of coordinates that represent the relationship between first order investment costs and operating costs for each elementary flux mode.

EXERCISE: Plot the biomass glucose operating cost versus the amino acid investment cost in a scatter plot (copy and paste the data as values into

worksheet ‘Biomass_data’ to facilitate sorting and plotting). The amino acid investment cost can be related to the energy state of the cell and phosphorus availability. Adjust the scale of the x-axis (glucose operating cost) to values between 1 and 3.5 and adjust the y-axis (amino acid investment cost) to 20,000 to 110,000. A minimization envelope can be established and the corresponding pathways can be identified as previously demonstrated. As before, this envelope represents a continuum of metabolic strategies minimizing the combined costs represented by each axis.

Note: The results from the ‘Ec200.txt’ *E. coli* model are slightly different than some of the presented data taken from Carlson, 2007. While the methodologies are the same, the models are quite different.

Special topics:

-Knock out mutants, recombinant pathways, and Euclidean distances

Elementary flux mode analysis is very useful for analyzing how recombinant pathways can be integrated into a host's existing metabolic structure or how a cell might respond if a gene is knocked out (KO).

EXERCISE: We will modify the 'Ec200.txt' model to examine the production of a biopolymer known as polyhydroxybutyric acid (PHB). *E. coli* does not natively produce PHB. The addition of the three gene PHB operon, however, permits *E. coli* to make significant amount of the bioplastic (up to 80-90% of the cell's dry weight in some cases).

Add the following reaction (period included) to the 'Ec200.txt' METATOOL input file –CAT section:

R100 : 2 ACETYL_CoA + NADH = 2 CoASH + NAD + PHB .

Declare the reaction 'R100' in the –ENZIRREV reaction section and the new metabolite 'PHB' in the –METEXT external metabolite section.

To make the analysis simpler, remove the biomass synthesis reaction from the model (R70). This corresponds to analyzing PHB production under non-growth conditions. You will then have to remove the R70 reaction from the –ENZIRREV declaration and remove the BIOMASS metabolite from the –METEXT declaration.

HINT: By adding or removing reactions, the reaction associated with each column will change in the output file and require a change in our templates. To minimize the changes, delete the R70 declaration in '–ENZIRREV' and declare R100 in the same position (R100 will be flanked by R55 and R80). Now the only change necessary to the templates is to substitute the 4 carbon PHB monomer for biomass. All other columns will remain unchanged and in-place.

Save the text file as 'EcPHB.txt'.

Run elementary mode analysis on the new input file 'EcPHB.txt'. When processing the data, remember that the yield formulas are set-up for biomass, so the formulas will need to modified (1 mole PHB = 4 Cmol PHB).

1) How many PHB producing modes do you find? **(answer: 222)**
2) How many anaerobic PHB modes do you find? **(answer: 44)**

3) How many anaerobic PHB modes do you find for an *E. coli* strain with a knocked-out *zwf* gene (oxidative pentose phosphate pathway, R10)? **(answer: 36)**

NOTE: The results here are different than Carlson *et al.,* 2005 because the current case considers only glucose as a carbon and energy source. Carlson *et al*., 2005 considers additional possible substrates.

-Euclidean distance

It is often useful to predict likely cell behavior prior to running an experiment. A method known as minimization of metabolic adjustment (MOMA) has been used to predict the behavior of KO mutants and recombinant hosts expressing foreign pathways (Segre *et al*., 2002, PNAS 99: 15112-15117; Carlson *et al*., 2005). The method analyzes the Euclidean distance between two metabolic flux patterns. Calculation of Euclidean distance is an extension of determining the distance between two points in a two-dimensional Cartesian system.

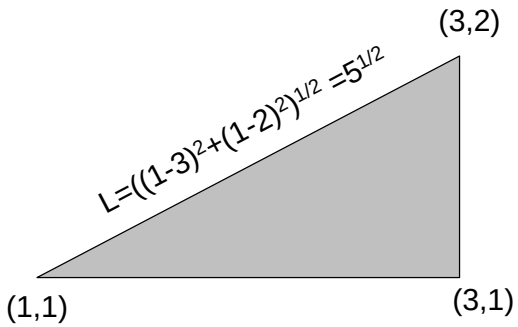

Figure 4. Illustration of 2-dimensional distance calculation.

The MOMA method assumes that flux patterns with smaller Euclidean distances represent smaller metabolic adjustments (perturbations), and that engineered cells will minimize perturbations from the native metabolic state. Flux patterns that are close use similar enzyme sets with similar magnitude fluxes.

Under anaerobic, non-growth conditions, we will assume *E. coli* produces maintenance energy using the most efficient ATP producing elementary flux mode (M_4^{ATP}). The Euclidean distance between the mode for the ATP synthesis and the modes for PHB production can be calculated. If we limit our analysis to just individual modes and not linear combinations of modes, the PHB modes with the shortest Euclidean distance from the energy generating modes would represent the smallest metabolic adjustment of native fluxes. If the MOMA theory is appropriate, we would expect the cells to produce PHB under anaerobic conditions using these pathways or linear combinations of these pathways.

The equation below is used to calculate the Euclidian distance between two different elementary modes.

$$D_j\left(M_j^{PHB}, M_4^{ATP}\right) = \sqrt{\sum_{i=1}^{R}(m_{i,j}^{PHB} - m_{i,4}^{ATP})^2} \quad ; \text{j=1,n}^{\text{PHB}}$$

D is the calculated Euclidean distance. There is a Euclidean distance (D_j) calculated for each PHB producing mode (M_j^{PHB}) measured from the ATP producing mode (M_4^{ATP}). R is the number of reactions in the model (for this case we have 44 reactions), m_{ij} is a normalized flux coefficient for reaction i in mode j, and n^{PHB} is the number of PHB producing modes.

EXERCISE: Calculate the Euclidean distances between the most efficient anaerobic ATP producing mode and the anaerobic PHB producing modes. Before the Euclidean distance can be calculated, the anaerobic PHB producing modes must be normalized with respect to glucose flux and converted into an absolute value format. The glucose flux for each mode will be normalized to 1. The template for this conversion is found in the 'EFMA_Workbook_Aids' 'Absolute_value_matrix' worksheet.

Copy and paste this template into a new worksheet in your 'EcPHB' workbook.

Next, paste the anaerobic PHB modes into the template, then perform the matrix transformation by pulling down the formula on the right side of the workbook.

Next, copy and paste the Euclidean distance template from 'EFMA_Workbook_Aids','Workbook_formulas' into another new worksheet on your 'EcPHB' workbook. The formula for calculating the Euclidean distance will have to be adjusted so that it does the calculations based on the proper range of cells.

NOTE: The formula is quite long and editing it in Excel is tedious and difficult. A shortcut technique is to copy the formula and paste it into MS Word. Control-H opens the 'Find and Replace' window, allowing the cell range to be quickly modified.

As an example the Euclidean distance formula from cell BE85 in 'EFMA_Workbook_Aids' is:

```
=((B85-B$85)^2+(C85-C$85)^2+(D85-D$85)^2+
(E85-E$85)^2+(F85-F$85)^2+(G85-G$85)^2+(H85-H$85)^2+
(I85-I$85)^2+(J85-J$85)^2+(K85-K$85)^2+(L85-L$85)^2+
(M85-M$85)^2+(N85-N$85)^2+(O85-O$85)^2+(P85-P$85)^2+
(Q85-Q$85)^2+(R85-R$85)^2+(S85-S$85)^2+(T85-T$85)^2+
(U85-U$85)^2+(V85-V$85)^2+(W85-W$85)^2+(X85-X$85)^2+
(Y85-Y$85)^2+(Z85-Z$85)^2+(AA85-AA$85)^2+(AB85-AB$85)^2+
(AC85-AC$85)^2+(AD85-AD$85)^2+(AE85-AE$85)^2+(AF85-AF$85)^2+
(AG85-AG$85)^2+(AH85-AH$85)^2+(AI85-AI$85)^2+
(AJ85-AJ$85)^2+(AK85-AK$85)^2+(AL85-AL$85)^2+(AM85-AM$85)^2+
(AN85-AN$85)^2+(AO85-AO$85)^2+(AP85-AP$85)^2+(AQ85-AQ$85)^2+
(AR85-AR$85)^2+(AS85-AS$85) )^0.5
```

Paste this formula into MS Word.
Select Edit/Replace and replace the row indicator 85 with the appropriate row indicator from the new worksheet (*e.g*. row 6).
With the Euclidean distance template properly configured, copy and paste special (values) the normalized absolute value matrix into the Euclidean distance template.
Pull down the formula to calculate the Euclidean distance for each anaerobic PHB producing mode.
Sort the results for the elementary modes that have the shortest and largest Euclidean distances.

Shortest distance: 2.7689; Longest distance: 6.3605

-Compartmentalizing metabolites and reactions: applications for eukaryotic cells and interacting cell populations

Elementary mode analysis can be used to study metabolite fluxes between cellular compartments like the cytosol and mitochondria or between different organisms. The following example is a model of the eukaryotic yeast *Saccharomyces cerevisiae*, and contains extracellular, cytosolic and mitochondrial compartments. This same type of modeling can be used to explore interactions between different cells or between different organisms as shown below.

The input file, a modification of the original model found in Carlson *et al.,* 2002, is listed below. The file, which includes biomass synthesis, is named 'SC01.txt'. It has 59 reactions and 48 metabolites. The model only considers glucose as a possible energy source while CO_2, ethanol, glycerol, acetate, succinate and biomass are possible by-products. The biomass composition is for a *S. cerevisiae* culture growing at $\mu = 0.1\ hr^{-1}$. The biomass term in this model is constructed in a manner analogous to the *E. coli* model biomass term. 'EFMA_Workbook_Aids' contains a worksheet titled 'Biomass_Worksheet' that can be used to calculate these model contributions. An illustration of the central reactions is shown below in Figure 5. A second *S. cerevisiae* model input file, with the biomass reaction removed, is named 'SCnoX.txt'.

-ENZREV
G2r G5r G6r G7r PPP2r PPP3r PPP4r PPP5r PPP6r IM10r IM11r IM14r M3r M6r M7r M8r M11r T6r MT2r

-ENZIRREV
G1 G3 G4 G8 PPP1 IM1 IM2 IM3 IM4 IM5 IM6 IM7 IM8 IM9 IM12 IM13 IM15 IM16 M1 M2 M4 M5 M9 M10 M12 M13 M14
BIOM E1 T1 T2 T3 T4 T5 T7 T8 MT1 MT3

-METINT
GLU_cyt ATP_cyt GLU_6_P FRU_6_P FRU_BIS_P DHAP GA_3P NADH_cyt NADPH
RIBULOSE_5_P XYL_5_P RIBOSE_5_P SED_7_P ERYTH_4_P PYR_cyt
CITRATE OXALO ACETYL_CoA_cyt GLYCEROL_P PEP AKG ISOCIT PYR_mit MALATE
ACETYL_CoA_mit ATP_mit NADH_mit ETOH ACEADH ACETATE GLYCEROL_cyt
SUCC GLYOX_cyt FUMARATE NADPH_mit OXY CO2 NH3

-METEXT
ATP_base ACETATE_ext CO2_ext SUCC_ext
ETOH_ext GLYCEROL_ext GLU_ext BIOMASS NH3_ext OXY_ext

-CAT

G1 : GLU_cyt + ATP_cyt = GLU_6_P .
G2r : GLU_6_P = FRU_6_P .
G3 : FRU_6_P + ATP_cyt = FRU_BIS_P .
G4 : FRU_BIS_P = FRU_6_P .
G5r : FRU_BIS_P = DHAP + GA_3P .
G6r : GA_3P = DHAP .
G7r : GA_3P = PEP + ATP_cyt + NADH_cyt .
G8 : PEP = PYR_cyt + ATP_cyt .

PPP1 : GLU_6_P = RIBULOSE_5_P + 2 NADPH + CO2 .
PPP2r : RIBULOSE_5_P = XYL_5_P .
PPP3r : RIBULOSE_5_P = RIBOSE_5_P .
PPP4r : RIBOSE_5_P + XYL_5_P = SED_7_P + GA_3P .
PPP5r : GA_3P + SED_7_P = ERYTH_4_P + FRU_6_P .
PPP6r : ERYTH_4_P + XYL_5_P = GA_3P + FRU_6_P .

IM1 : PYR_cyt = ACEADH + CO2 .
IM2 : ACEADH + NADH_cyt = ETOH .
IM3 : ACEADH = ACETATE + NADPH .
IM4 : ACETATE + 2 ATP_cyt = ACETYL_CoA_cyt .
IM5 : ACETYL_CoA_cyt = ACETATE .
IM6 : GLYOX_cyt + ACETYL_CoA_cyt = MALATE .
IM7 : ISOCIT = GLYOX_cyt + SUCC .
IM8 : OXALO + ACETYL_CoA_cyt = CITRATE .
IM9 : ISOCIT = AKG + NADPH + CO2 .
IM10r : FUMARATE + NADH_cyt = SUCC .
IM11r : OXALO + NADH_cyt = MALATE .
IM12 : OXALO + ATP_cyt = PEP + CO2 .
IM13 : PYR_cyt + ATP_cyt + CO2 = OXALO .
IM14r : DHAP + NADH_cyt = GLYCEROL_P .
IM15 : GLYCEROL_P = GLYCEROL_cyt .
IM16 : GLYCEROL_cyt + ATP_cyt = GLYCEROL_P .

M1 : PYR_mit = ACETYL_CoA_mit + NADH_mit + CO2 .
M2 : OXALO + ACETYL_CoA_mit = CITRATE .
M3r : CITRATE = ISOCIT .
M4 : ISOCIT = AKG + NADH_mit + CO2 .
M5 : AKG = NADH_mit + ATP_mit + SUCC + CO2 .
M6r : SUCC = FUMARATE + NADH_mit .
M7r : FUMARATE = MALATE .
M8r : MALATE = OXALO + NADH_mit .
M9 : MALATE = PYR_mit + NADPH_mit + CO2 .
M10 : ETOH = ACEADH + NADH_mit .
M11r : ACEADH = ACETATE + NADH_mit .
M12 : ACETATE + 2 ATP_mit = ACETYL_CoA_mit .
M13 : NADH_mit + OXY = ATP_mit .
M14 : ACETYL_CoA_mit = ACETATE .

BIOM : 6 GLU_6_P + RIBULOSE_5_P + ERYTH_4_P + 4 PEP + 8 PYR_cyt + 5 ACETYL_CoA_cyt + 3 AKG + 5 OXALO + 102 ATP_cyt + 36 NADPH + 21 NH3 + CO2 = BIOMASS + 6 NADH_cyt .

E1 : ATP_cyt = ATP_base .

T1 : GLU_ext = GLU_cyt .
T2 : ETOH = ETOH_ext .
T3 : ACETATE = ACETATE_ext .
T4 : SUCC = SUCC_ext .
T5 : GLYCEROL_cyt = GLYCEROL_ext .
T6r : CO2_ext = CO2 .
T7 : OXY_ext = OXY .
T8 : NH3_ext = NH3 .

MT1 : PYR_cyt = PYR_mit .

MT2r : ACETYL_CoA_mit = ACETYL_CoA_cyt .
MT3 : ATP_mit = ATP_cyt .

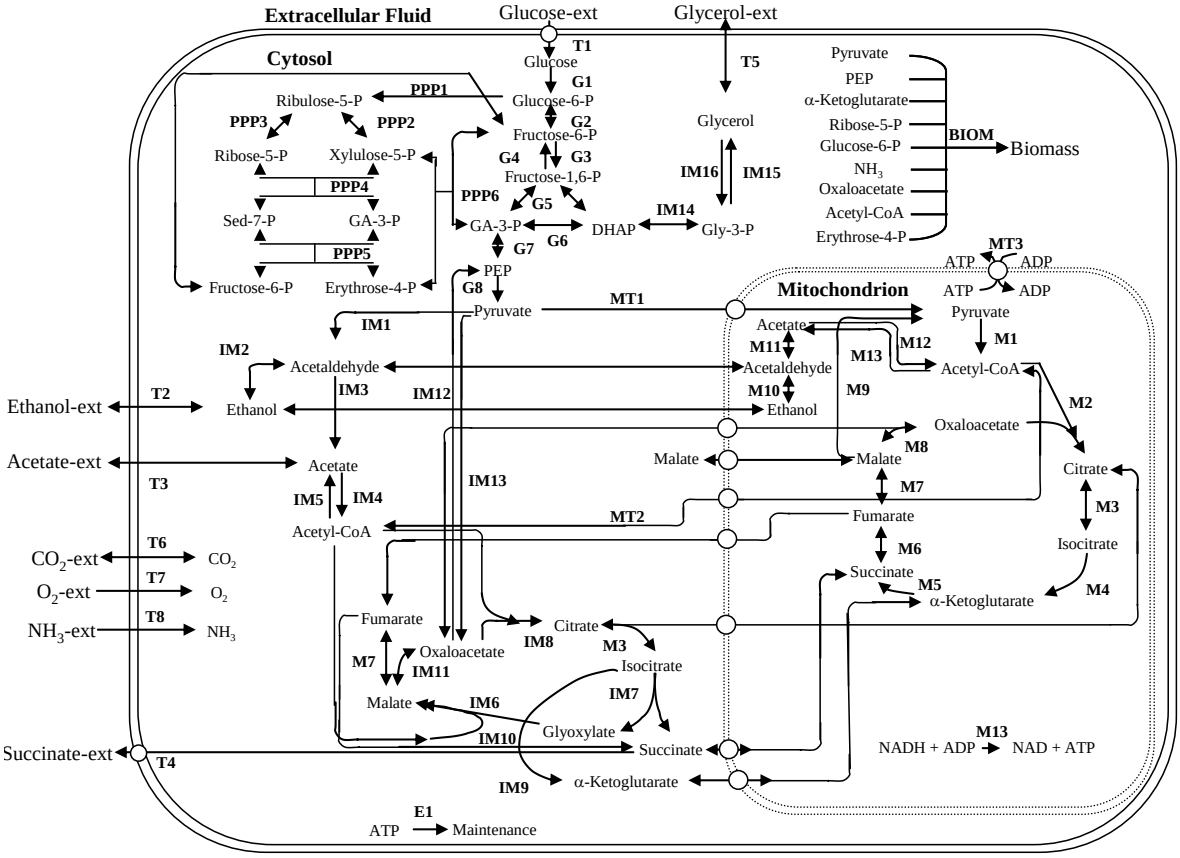

Figure 5. Graphical representation of many reactions from the compartmentalized *S. cerevisiae* model 'SC01.txt'. A PowerPoint version of this file can be found in 'EFMA_model_diagrams.ppt'.

Points of note on the model: Some metabolites occur in both the cytosol and the mitochondria. These metabolites are partitioned into different physical locations within the cell, and the different pools are distinguished in the model by the suffix '_cyt' or '_mit'. For instance, the mitochondrial transport reactions MT1, MT2r and MT3 contain both cytosolic and mitochondrial localized metabolites. Other metabolites, like CO_2, are not split into different metabolite pools even though they participate in reactions in both locations (see reaction mitochondrial reaction M1 and cytosolic reaction PPP1). This is due to the relatively free diffusion of CO_2 across cellular membranes. Knowing when it is appropriate to specify separate pools takes practice. If in doubt, always create partitioned versions of the metabolite. Be aware, however, that this will increase the computational burden.

The reactions labeled with an 'M' are mitochondrial reactions. The mitochondrial reactions are integrated with the cytosolic reactions so that each elementary

mode has balanced carbon, electron, and energy fluxes. Sorting the output for mitochondrial reactions will reveal which modes involve flux through the separate compartment. For instance, the TCA cycle is found in the mitochondria but is typically fed with pyruvate from the cytosolic glycolysis enzymes. ATP and reducing equivalents can be shuttled back and forth between the cytosol and mitochondria using shuttle metabolites like malate and oxaloacetate or using transporters which can exchange the high energy phosphate bonds from mitochondrial ATP to cytosolic ATP.

The model 'SC01.txt' takes approximately 5 minutes to run with the 'doubletool.exe' program and generates 96,850 modes. Old versions of MS Excel could not handle this many rows, so the data had to be broken into separate, smaller files or transferred to a program like MATLAB. The same model, with the biomass reaction removed, is named 'SCnoX.txt'. This model generates 10,787 modes, runs in approximately 1 minute, and the output can easily be processed with MS Excel.

Exploration of this model will depend on the available time and will be likely left up to the workshop participants.

-interacting microbes

An additional example is provided that looks at the interactions between a photoautotroph and a heterotroph. The photoautotroph harvests energy from sunlight and electrons from water, releasing O_2 as a by-product. The electrons can be used to fix CO_2 into organic carbon. Some of this organic carbon can be excreted, feeding the heterotrophic metabolism.

The metabolite pools are partitioned between the organisms using the same approach as the fungal model: the suffix 'auto' corresponds to metabolites found in the photoautotroph while the suffix 'heter' refers to metabolites found in the heterotroph.

-ENZREV
A1r A4r A8r A9r A10r A11r A13r H2r H6r H7r H8r H9r H11r H14r H16r H21r T3r

-ENZIRREV
A2 A3 A5 A6 A7 A12 A14 A15 A16 A17 A18 A19 A21 A22 A23 A24 A25 A26 A27 A28 A29 A30 A31 A32 H1 H3 H4 H5 H10 H12 H13 H15 H17 H18 H19 H20 H22 H23 H24 H25 H26 H27 H28 H29 H30 H31 H32 H33 T1 T2

-METINT
ATP_auto ac_CoA_auto ac_pool akg_auto CO2_auto ery4p_auto fru6p_auto ga3p_auto glc6p_auto glyc_auto glycpool_gen icit_auto mal_auto NADH_auto NH3_auto O2_auto O2gen_pool oaa_auto PEP_auto pyr_auto rbl5p_auto rbo5p_auto succ_auto xll5p_auto ac_CoA_heter akg_heter ATP_heter CO2_heter ery4p_heter fru6p_heter ga3p_heter glc6p_heter glyc_heter glyox_heter H2_heter icit_heter mal_heter NADH_heter NH3_heter

O2_heter oaa_heter PEP_heter pyr_heter rbl5p_heter rbo5p_heter succ_heter xll5p_heter

-METEXT
hv_gen CO2ex_gen NH3ex_gen ATPex_auto bm_auto O2ex_pool ATPex_heter bm_heter H2ex_gen acex_pool glycex_pool

-CAT

A1r : O2_auto = O2gen_pool .
A2 : fru6p_auto = glc6p_auto .
A3 : 2 ga3p_auto = fru6p_auto .
A4r : ga3p_auto = ATP_auto + NADH_auto + PEP_auto .
A5 : 2 ATP_auto + pyr_auto = PEP_auto .
A6 : pyr_auto = ac_CoA_auto + CO2_auto + NADH_auto .
A7 : glc6p_auto = CO2_auto + 2 NADH_auto + rbl5p_auto .
A8r : rbl5p_auto = xll5p_auto .
A9r : rbl5p_auto = rbo5p_auto .
A10r : rbo5p_auto + xll5p_auto = ery4p_auto + fru6p_auto .
A11r : ery4p_auto + xll5p_auto = fru6p_auto + ga3p_auto .
A12 : ac_CoA_auto + oaa_auto = icit_auto .
A13r : CO2_auto = CO2ex_gen .
A14 : NH3ex_gen = NH3_auto .
A15 : ATP_auto = ATPex_auto .
A16 : 1.233 ac_CoA_auto + 1.472 akg_auto + 50 ATP_auto + 0.531 ery4p_auto + 0.069 glc6p_auto + 14.653 NADH_auto + 12.513 NH3_auto + 2.379 oaa_auto + 2.67 PEP_auto + 4.057 pyr_auto + 0.787 rbo5p_auto = bm_auto .
A17 : 8 hv_gen = 3 ATP_auto + 2 NADH_auto + O2_auto .
A18 : 2 NADH_auto + O2_auto = 5 ATP_auto .
A19 : CO2_auto + PEP_auto = oaa_auto .
A21 : ATP_auto + rbl5p_auto + O2_auto = PEP_auto + glyc_auto .
A22 : glyc_auto = glycpool_gen .
A23 : PEP_auto = ATP_auto + pyr_auto .
A24 : ATP_auto + CO2_auto + rbl5p_auto = 2 PEP_auto .
A25 : ATP_auto + fru6p_auto = 2 ga3p_auto .
A26 : icit_auto = akg_auto + CO2_auto + NADH_auto .
A27 : mal_auto = CO2_auto + NADH_auto + pyr_auto .
A28 : akg_auto = ATP_auto + CO2_auto + NADH_auto + succ_auto .
A29 : ATP_auto + oaa_auto = CO2_auto + PEP_auto .
A30 : ac_CoA_auto = ATP_auto + ac_pool .
A31 : O2_auto + 2 succ_auto = 2.5 ATP_auto + 2 mal_auto .
A32 : 2 hv_gen = ATP_auto .

H1 : fru6p_heter = glc6p_heter .
H2r : ATP_heter + fru6p_heter = 2 ga3p_heter .
H3 : ATP_heter + NADH_heter + PEP_heter = ga3p_heter .
H4 : 2 ATP_heter + pyr_heter = PEP_heter .
H5 : glc6p_heter = CO2_heter + 2 NADH_heter + rbl5p_heter .
H6r : rbl5p_heter = xll5p_heter .
H7r : rbl5p_heter = rbo5p_heter .
H8r : rbo5p_heter + xll5p_heter = ery4p_heter + fru6p_heter .
H9r : ery4p_heter + xll5p_heter = fru6p_heter + ga3p_heter .
H10 : ac_CoA_heter + oaa_heter = icit_heter .
H11r : icit_heter = akg_heter + CO2_heter + NADH_heter .
H12 : akg_heter = ATP_heter + CO2_heter + NADH_heter + succ_heter .
H13 : succ_heter + 0.5 O2_heter = 1.25 ATP_heter + mal_heter .
H14r : mal_heter = NADH_heter + oaa_heter .

H15 : icit_heter = glyox_heter + succ_heter .
H16r : ac_CoA_heter + glyox_heter = mal_heter .
H17 : mal_heter = CO2_heter + NADH_heter + pyr_heter .
H18 : ATP_heter + oaa_heter = CO2_heter + PEP_heter .
H19 : ATP_heter + 2 glyox_heter + NADH_heter = CO2_heter + PEP_heter .
H20 : ATP_heter = ATPex_heter .
H21r : CO2_heter = CO2ex_gen .
H22 : NH3ex_gen = NH3_heter .
H23 : 1.233 ac_CoA_heter + 1.472 akg_heter + 50 ATP_heter + 0.531 ery4p_heter + 0.069 glc6p_heter + 14.653 NADH_heter + 12.513 NH3_heter + 2.379 oaa_heter + 2.67 PEP_heter + 4.057 pyr_heter + 0.787 rbo5p_heter = bm_heter .
H24 : 2 NADH_heter + O2_heter = 5 ATP_heter .
H25 : glyc_heter = glyox_heter + NADH_heter .
H26 : CO2_heter + PEP_heter = oaa_heter .
H27 : pyr_heter = ac_CoA_heter + CO2_heter + NADH_heter .
H28 : ATP_heter + 4 NADH_heter = 4 H2_heter .
H29 : PEP_heter = ATP_heter + pyr_heter .
H30 : glycpool_gen = glyc_heter .
H31 : O2gen_pool = O2_heter .
H32 : 2 ATP_heter + ac_pool = ac_CoA_heter .
H33 : H2_heter = H2ex_gen .

T1 : ac_pool = acex_pool .
T2 : glycpool_gen = glycex_pool .
T3r : O2ex_pool = O2gen_pool .

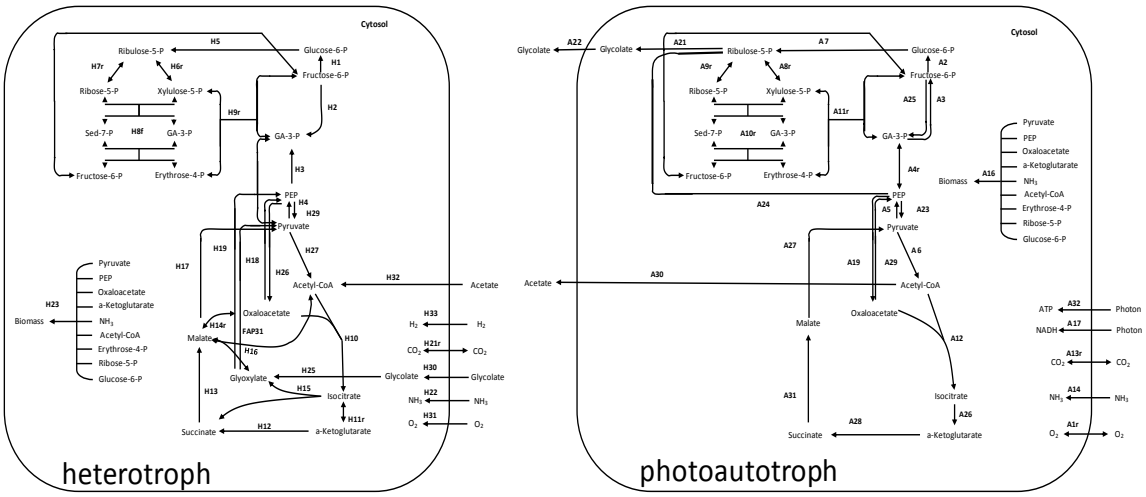

Figure 6. Graphical representation of the photoautotroph and heterotroph model microbial community found in file 'AHC.txt'. The model permits the exchange of acetate and glycolate from the photoautotroph to the heterotroph. PowerPoint versions of models can be found in 'EFMA_model_diagrams.ppt'.

The model community possesses 808 unique elementary modes which harness sunlight to produce photoautotroph or heterotroph biomass, cellular energy, or fixed carbon like acetate or glycolate.

Exploration of this model will depend on the available time and will be likely left up to the workshop participants. For an example of a three guild model, please see Taffs *et al.,* 2009 included in the Manuscript folder.

-Incorporation of ^{13}C fluxomic data and elementary modes

Please see the presentation files Carlson, 2009 for details. This analysis requires the use of mathematical programs such as MATLAB. The material will be presented without associated exercises.

-Large-scale elementary mode computation with efmtool

METATOOL is a convenient standalone program for elementary mode computation, but it can be outperformed in terms of speed and allowable input network size by the latest algorithms. The current state of the art elementary mode calculation software is known as efmtool, freely available at http://www.csb.ethz.ch/tools/efmtool. The relevant reference is: Terzer, M. and Stelling, J. (2008) Large-scale computation of elementary flux modes with bit pattern trees Bioinformatics, 24(19): 2229-2235.

Setting up efmtool

efmtool can be utilized from within MATLAB, but not everybody has a MATLAB license. Additionally, because MATLAB uses RAM and overhead, we get better results running efmtool from the command prompt. To do this we use a shortcut to command.exe that executes a batch file with the command to run efmtool. Both the batch file and the command prompt need onetime adjustments to accommodate the machine configuration.

To modify the command.exe shortcut, right click 'command prompt' and choose properties. In the 'start in' box change this to match the path where you have installed efmtool. This *.bat file is efmtool.bat on the CD. Right click it and choose edit to change it to suit your needs. An example batch file command to start efmtool is given below.

```
java -Xmx7g -XX:+UseParallelOldGC -XX:+UseAdaptiveSizePolicy -XX:+UseTLAB -
XX:+ResizeTLAB -XX:MaxPermSize=15m -cp lib\metabolic-efm-all.jar;lib\dom4j-1.6.1.jar;lib\junit-
3.8.1.jar ch.javasoft.metabolic.efm.main.CalculateFluxModes -kind stoichiometry -stoich
tmp\stoich.txt -rev tmp\revs.txt -meta tmp\mnames.txt -reac tmp\rnames.txt -arithmetic double -
zero 1e-010 -out matlab tmp\efms.mat -compression default -log console -level INFO -tmpdir
C:\Documents and Settings\James.folsom\My Documents\MATLAB\efmtool\tmp -maxthreads -1 -
normalize min -adjacency-method pattern-tree-minzero -rowordering MostZerosOrAbsLexMin
```

Java switches follow the 'java' command and are preceded by dashes. 'ch.javasoft.metabolic.efm.main.CalculateFluxModes' is actually the Java code that performs the calculations. efmtool options follow this command and are preceded by a dash. The switches we use are explained below. efmtool options,

including some that are not described below, are specified in the header for 'CreateFluxModeOpts.m'. That file is included on the CD as 'efmtool_opts.txt'.

Java switches

***-**Xmx7g*

This switch sets the maximum amount of RAM that efmtool will use. The above example sets the maximum to 7 gigabytes. 'm' is for megabytes. The switch must be whole numbers, and need to be changed to suit your machine. For a typical 32 bit machine with 4 GB of RAM, use –Xmx3g or –Xmx3072m.

-XX:+UseParallelOldGC

This switch enhances memory management on multi-core computers. Replace with *–XX:+UseSerialGC* on single core machines, or omit this option entirely.

XX:+UseAdaptiveSizePolicy

This setting optimizes the Java environment to better suit how efmtool works. This permits monitoring and resizing of Java memory generations, so you may also find this switch useful for other Java programs that are memory intensive.

-XX:+UseTLAB and -XX:+ResizeTLAB

These optimize the cooperation between different processors on a multi-core machine. Omit this switch on single core machines.

-XX:MaxPermSize=15m

This setting prevents Java from reserving more memory for overhead than necessary.

Useful Java technical details can be found at the following websites:
http://blogs.sun.com/daviddetlefs/entry/tlab_sizing_an_annoying_little
http://developers.sun.com/mobility/midp/articles/garbagecollection2/
http://java.sun.com/javase/technologies/hotspot/gc/gc_tuning_6.html#icms.available_options

efmtool switches

-kind stoichiometry -stoich tmp\stoich.txt -rev tmp\revs.txt -meta tmp\mnames.txt -reac tmp\rnames.txt

This set of switches tells EFMtool that the input is a stoichiometric matrix located in the "tmp" subdirectory of the directory you put EFMtool in. The names of the files are specified: 'stoich.txt' contains the stoichiometric matrix, 'revs.txt' contains the reversibility of the reactions as a binary vector, 'mnames.txt' contains the names of the metabolites, and 'rnames.txt' contains the names of the reactions. These are the default settings.

-out matlab tmp\efms.mat

This specifies that the output containing elementary modes will be a MATLAB workspace. If you don't use MATLAB, elementary modes can be provided in a

text file (in the same matrix format we used for processing METATOOL output) by replacing this switch with the following: *-out text-doubles tmp\efms.txt*.

-tmpdir C:\Documents and Settings\James.folsom\My Documents\MATLAB\efmtool\tmp
This specifies the path to the input files and the location of the output file. It can be adjusted to suit the user.

-memory sort-out-core
-memory out-core
-adjacency-method rankup-modpi-outcore
These three switches are options to use disk space for storage of intermediate results during the computation. This allows the analysis of larger models, but at a slower rate.
Either of the first two switches can be added at the end of the .bat file, and the third switch replaces the existing adjacency-method switch.

The path to Java usually needs to be specified to the operating system in order for efmtool to work. In Windows XP or 7, this is done by right-clicking 'my computer' and choosing 'properties'. (In Windows 7, you must now click 'Advanced System Settings' from the left pane.) Choose the 'Advanced Tab', and click 'Environment Variables' at the bottom of the window. Look for the 'System Variables' pane, and under it click 'new'. In the window that opens, type 'java' in the 'variable name' box and provide the path to java.exe. You can obtain this path using the search features of the OS. The correct path is usually similar to 'C:\Program Files\Java\jre6\bin'.

Running efmtool
When all the files have been customized to your needs, place the efmtool input files in the directory specified by the '-tmp' switch, and double click the command prompt shortcut.

The input files you've built for METATOOL can be quickly converted into input files for efmtool using a Python script provided on your CD. Download and install Python 2.6.5 (freely available @ http://www.python.org/download/). Copy the script 'metatool2efmtool.py' to the directory containing the metatool input file that needs to be converted. Double-click on the 'metatool2efmtool.py' icon and follow the directions. (Alternatively, navigate to the appropriate folder with the command prompt and then use the following syntax: python metatool2efmtool.py.)

Additional files and tools:

The folder 'FAEc07' contains the FluxAnalyzer (v 5.2) input files required to run the *E. coli* model found in Carlson, 2007. This model has approximately 3.5 million elementary flux modes. The 'EFMA_Workbook_aids',

‘Rxn_Gene_Investment_worksheet’ has the reaction designation, gene assignment, enzyme subunit information, and elemental investment requirement data to reproduce the investment cost analysis. The ‘EFMA_Workbook_Aids’, ‘Biomass_worksheet’ has the templates for the biomass term for seven different doubling times and can be modified to generate biomass terms for other macromolecular compositions. The PowerPoint file ‘EFMA_model_diagrams.ppt’ has a graphical representation of the included models.

Finally…

Feedback, suggestions, and questions are encouraged. Please feel free to contact Ross at:

Ross Carlson, PhD
Montana State University, Bozeman
406-994-3631
rossc@erc.montana.edu