Today:

Announcements:
- Presentation times
- Presentation grade and tips
- Problem set 5 will be posted Dec. 3rd
- Mathematica 4.2

References for FBA

Review:

Application to E. Coli:

J. Edwards, R. Ibarra, B. Palsson, “In silico predictions of E. coli metabolic capabilities are

Mathematical Methods:
C. Schilling, B. Palsson, “The underlying pathway structure of biochemical reaction networks”.


C. Schilling et al, “Metabolic pathway analysis: Basic concepts and scientific applications in the

These three are available on the course website.
METABOLIC NETWORKS

Why study biological networks?
Genomics and proteomics tell us the ‘parts’ of the cell, but they give limited information about their interactions and co-dependencies. Ideally, we’d like to understand the interactions well enough that we could predict the system’s behavior computationally – “in silico biology”.

Applications:
- Phenotype prediction
- Bioengineering
- Synthetic media design

“In Silico Biology”
Uses computational representations of a cell or organism to model the organism’s behavior. One way to do this is using a kinetic model (e.g., the red blood cell model discussed in class). Another useful method uses the stoichiometry of different cellular components and mass conservation; this method is called Flux Balance Analysis.

Definitions:

 Flux: Rate of a reaction

 Mass Balance: Just refers to conservation of matter; the amount of a material in the cell is the sum of how much is put in and synthesized, minus what is used or taken out.

How it works:
Four basic types of metabolic flux rates affect the concentration of any given metabolite pool:

\[ v_{syn} = \text{synthesis rate} \]
\[ v_{deg} = \text{degradation rate} \]
\[ v_{trans} = \text{transport rate (import and export)} \]
\[ v_{use} = \text{consumption rate (use rate required for growth and maintenance)} \]

The net flux rate for a metabolite \( X_i \) is:

\[ \frac{dX_i}{dt} = v_{syn} - v_{deg} + (v_{trans} - v_{use}) \]

\[ \frac{dX_i}{dt} = v_{syn} - v_{deg} - b_i \text{ where } b_i = v_{trans} - v_{use} \]
Relationships between net flux, individual fluxes (reaction rates), and metabolite concentrations:

- The reaction rates are generally non-linear functions of \([X]\).

  **example:** the Michaelis-Menten rate equation (hyperbolic)

  \[
  V = \frac{V_{max}X}{K_m + X}
  \]

- Therefore the dynamic mass balance equations are linear functions of the reaction rates, BUT are non-linear functions of the actual metabolite concentrations.

Consequently

⇒ they are difficult to treat mathematically
⇒ generally exact kinetic expressions for *in vivo* enzyme reactions are unavailable

**Steady state analysis**

The steady state assumption

Steady state analysis helps us get around this problem of non-linearity. The steady state assumption says that the rate of flux, \(\frac{dX}{dt}\), is 0; this assumption basically removes \(X\) and \(t\) as variables.

Solving the steady state mass balance equation

When \(S\) is available from genomic analysis and \(b\) can be determined experimentally, we can solve for \(v\).
How To Analyze a Network using FBA:

1. **Label Fluxes**
   - External fluxes: $b_1, b_2, b_3, \ldots$
   - Internal fluxes: $v_1, v_2, v_3, \ldots$

   Note: 1 flux per enzymatic reaction. Examples:

   \[
   \begin{align*}
   &G &\overset{\rightarrow}{\longrightarrow} &H \\
   &K &\overset{\rightarrow}{\longrightarrow} &L &\overset{\rightarrow}{\longrightarrow} &Q &\overset{\rightarrow}{\longrightarrow} &P
   \end{align*}
   \]

2. **Derive Equations**
   - One equation per biological component
   - Make sure you include all arrows/fluxes
   - Keep your signs consistent
     - E.g., arrows towards a component positive, arrows away negative

   Here we have three components, so we have three equations.

   \[
   \begin{align*}
   A: & \quad b_1 - v_1 - v_2 = 0 \\
   B: & \quad v_1 - b_2 = 0 \\
   C: & \quad v_2 - b_3 = 0
   \end{align*}
   \]

   Notice that we have five unknowns ($v_1, v_2, b_1, b_2, b_3$) and three equations. This is an example of an **underdetermined system**, which means we can’t find an exact solution to it. Instead we get a range of allowable solutions, where any set of parameters that falls within that range is a solution to the set of equations. This range is called the **null space**. We can find the null space of a set of equations using linear algebra, but to do this we need to convert our equations to matrix form.

3. **Convert the system of equations to a matrix equation of the form: $S \cdot V = 0$**
   - $S$ is a matrix that contains the numerical coefficients in the equations
   - $V$ is a vector whose components are the fluxes
   - Each row of $S$ corresponds to one equation, so our matrix has as many rows as our system has metabolites
   - Each column of $S$ corresponds to a flux, so our matrix has as many columns as our system has fluxes
   - The equation $S \cdot V = 0$ means that the dot product of the matrix $S$ with the vector $V$ equals 0
   - To enter a matrix with Mathematica, separate entries with “,”, and rows with {}. For example, a 2x2 matrix of all 1’s would be: $m=\{\{1,1\},\{1,1\}\}$
   - Matrices can be printed to output with more standard matrix-formatting using the command `MatrixForm[m]`, where $m$ is the matrix to be printed
The matrix equation for our system is:

\[
\begin{pmatrix}
-1 & -1 & 1 & 0 & 0 \\
1 & 0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0 & -1
\end{pmatrix}
\begin{pmatrix}
v[1] \\
v[2] \\
b1 \\
b2 \\
b3
\end{pmatrix}
= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}
\]

Note that you can perform the dot product to confirm that your matrix is correct.

A: \((-1)(v1) + (-1)(v2) + (1)(b1) + (0)(b2) + (0)(b3) = -v1 - v2 + b1\)

4. Solve for the “null space” of S

- There are linear algebra techniques that will allow you to do this by hand, but it’s much easier to just do it with Mathematica which has a built-in function for it:

\[
\text{MatrixForm[NullSpace[m]]}
\]

Here ‘m’ is the matrix whose null space we want, “NullSpace” is the Mathematica command to do it, and nesting it within “MatrixForm” turns the result into matrix format.

For our example, the solution is:

\[
\begin{pmatrix}
0 & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 & 0
\end{pmatrix}
\]

Each row is a linearly- and genetically-independent pathway in our system. Because they are linearly independent, they form a basis, meaning any pathway in our system can be constructed from linear combinations (additions and subtractions) of these vectors. Other basis vectors are i, j, and k which are often taken as basis vectors for the Cartesian coordinate system.

These solutions can be converted to reactions by taking the dot product of the row with the flux vector. For example, the first basis vector is equivalent to the following:

\[
= (0)(v1) + (1)(v2) + (1)(b1) + (0)(b2) + (1)(b3)
= v2 + b1 + b3
\]

This is equivalent to the following pathway:

\[
\begin{array}{c}
\text{b1} \\
\text{A} \\
\text{v2} \\
\text{C} \\
\text{b3}
\end{array}
\]
5. Construct a Biochemically Feasible Basis (if necessary)

A solution is unfeasible if it contains internal fluxes that are negative. The biochemical interpretation of this is that an enzyme is catalyzing the reverse reaction, which is not feasible. We can construct a feasible set by performing linear operations on the original set.

With our example, all of the internal fluxes are positive for all of the basis vectors, so this step isn’t necessary. A sample nullspace for which this isn’t the case is:

\[
\begin{pmatrix}
0 & 1 & -1 & 0 & -1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & -1
\end{pmatrix}
\]

We can construct a new basis as follows:

\[
P1 = s1 + s2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}
\]
\[
P2 = s2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}
\]
\[
P3 = s2 + s3 = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}
\]

This set corresponds to the range of “allowable” reactions in our system.

Note: You can access individual rows of a matrix in Mathematica. For example, to assign the fourth row of matrix \( m \) to the variable \( w \), the syntax is: \( w = \left[ \begin{bmatrix} m \end{bmatrix} \right] \)

6. Prediction of System Behavior using Linear Programming

Now that we have a biochemically feasible set of basis vectors, we can use them to predict the behavior of the system under a variety of different conditions/constraints. We do this using linear programming, which is a type of algorithm that optimizes a system of equations for a given cost function and under a set of constraints.

The cost function is the equation or component that you would like to optimize. For example, if we wanted to minimize the export of \( C \), the costs function is \( b3 \). The vector form of this, which is required by Mathematica, is written as follows: \( \{0, 0, 0, 0, 1\} \)

Note that Mathematica’s linear programming function only allows you to minimize a cost function, it won’t let you maximize something. To maximize a given parameter, you minimize the equivalent negative value. For example, to maximize the export of \( C \), the cost function is written as: \( \{0, 0, 0, 0, -1\} \)

The constraints for this type of question are:

- The components of the stoichiometric matrix, which specifies how the different fluxes relate to each other
- The range of values each flux can take
The corresponding Mathematica function for this is called **LinearProgramming**. This will only run in the way we need it to on version 4.2 (remember, upgrade if you’re running 4.1…). There are many different ways to use it, the most suitable for this is:

```
LinearProgramming[{0, 0, 0, 0, 1}, s,
    {{0, 0}, {0, 0}, {0, 0}},
    {{0, Infinity}, {0, Infinity}, {0, 100}, {5, 10}, {0, 10}}]
```

Here we are telling Mathematica the following:
- Minimize the export flux $k_3$
- The fluxes must satisfy matrix $s$, and each row of matrix $s$ is equal to zero
- Internal fluxes $v_1$ and $v_2$ can take any positive value
- Export fluxes $b_1$, $b_2$, and $b_3$ can be between 0 and 100, 5 and 10, and 0 and 10, respectively

This gives the following result:

$$\{5, 0, 5, 5, 0\}$$

This means that $k_3$ reaches its lowest value given the above constraints when the fluxes are: $v_1 = 5$, $v_2 = 0$, $v_3 = 5$, $v_4 = 5$, $v_5 = 0$.

---

**Figure of multi-dimensional null space:**

A schematic representation of the null space. This is a figure representing the null space in a hypothetical 3-flux system. All the feasible solutions to the FBA equation will be contained within this domain.

The null space of $S$ represents the metabolic capabilities of the network.
Most biologically meaningful objective function, at least for prokaryotic metabolic networks such as *E. coli*, is the biomass growth function.

\[ f = -\nu_{\text{growth}} \]

\[ f = \sum d_M \cdot M \rightarrow \text{biomass} \]

\( d_M \) represents stoichiometric ratios in which the metabolites \( M \) are used for growth.

These metabolic demands for growth are based on composition analysis of cell mass which for *E. coli* are:

<table>
<thead>
<tr>
<th>Cofactor</th>
<th>( d_M ) (mmol)</th>
<th>Precursor</th>
<th>( d_M ) (mmol)</th>
<th>Precursor</th>
<th>( d_M ) (mmol)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATP</td>
<td>41.2570</td>
<td>G6P</td>
<td>0.2050</td>
<td>3PG</td>
<td>1.4960</td>
</tr>
<tr>
<td>NADH</td>
<td>-3.5470</td>
<td>F6P</td>
<td>0.0709</td>
<td>PEP</td>
<td>0.5191</td>
</tr>
<tr>
<td>NADPH</td>
<td>18.2250</td>
<td>R5P</td>
<td>0.8977</td>
<td>Pyr</td>
<td>2.8328</td>
</tr>
<tr>
<td>E4P</td>
<td>0.3610</td>
<td>AcCoA</td>
<td>3.7478</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T3P</td>
<td>0.1290</td>
<td>OA</td>
<td>1.7867</td>
<td>oKG</td>
<td>1.0789</td>
</tr>
</tbody>
</table>

**Example *E. coli*:**

<table>
<thead>
<tr>
<th>Gene</th>
<th>Glucose</th>
<th>Glycerol</th>
<th>Succinate</th>
<th>Acetate</th>
<th>Glucose</th>
<th>Glycerol</th>
<th>Succinate</th>
<th>Acetate</th>
</tr>
</thead>
<tbody>
<tr>
<td>aceEF</td>
<td>-/+</td>
<td>-/-</td>
<td>-/-</td>
<td>-/-</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>aceA</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>aceB</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>ackA</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>acs</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>acn</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>cyd</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>cyo</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>eno</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>fba</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>fbp</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>gap</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>glk</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>glyA</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>gnd</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>idh</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>ndh</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>nrd</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>nruo</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>gpi</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>pgk</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>pgk</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>sda</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>ssp</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>ppc</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>pta</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>ptaacs</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>trkA</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
<tr>
<td>trkB</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
<td>-/+</td>
</tr>
</tbody>
</table>