

Stoichiometric matrix

The stoichiometric matrix corresponds to adjacency matrix (graph theory). All kinetic laws can be expressed as

$$\frac{dx}{dt} = Sv$$

We have: m reactions and n molecular species

$$x = (x_1 x_2 \dots x_n)^T \in R^n$$

$$v = (v_1 v_2 \dots v_m)^T \in R^m$$

typical example: $A + B \rightarrow C, C \rightarrow A + B$

$$\frac{d}{dt}[A] = k_2[C] - k_1[A][B]$$

$$\frac{d}{dt}[B] = k_2[C] - k_1[A][B]$$

$$\frac{d}{dt}[C] = k_1[A][B] - k_2[C]$$

Singular value decomposition

Let A be an $m \times n$ matrix. Then, there exist orthogonal matrices U and V and a matrix Σ such that $A = U\Sigma V^T$

Steady states

A steady state is a state x of the system for which

$$\frac{dx}{dt} = Sv = 0$$

Solving for $v = (V_1 V_2 \dots V_m)^T$ will give us the possible reaction rates at equilibrium

Matlab code

```
% Solving linear systems in Matlab
A = [1 2, 0 1]
Y = [1, 2]

% If we want to solve the equation Ax = y, we do...
x=A\y

S = [-1 1 -1, 1 -1 0, 0 0 1]
v = S\zeros(3,1) % uh oh...singular matrix
null(S)
```

We want to describe all possible solutions of $Sv = 0$ (equation *). It turns out that every solution v of * can be written as

$$v = c_1 v_1 + c_2 v_2 + \dots + c_k v_k$$

where $v_1 \dots v_k$ are some special solutions of *.

```
S = [-1 1 0 0 -1 1, 0 -1 1 0 0 0, 0 0 -1 1 -1 1]
A = S'
null(A)
```

Conservation relations

$$\frac{dx}{dt} = Sv$$

S is nxm

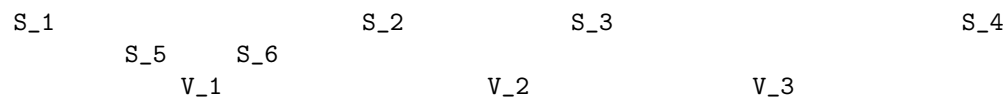
Consider a vector x such that $x^T S = 0$

$$y^T \frac{dx}{dt} = y^T (Sv) \rightarrow \frac{dy^T x}{dt} = y^T S v = 0 \rightarrow \frac{d}{dt} y^T x = 0$$

This implies that $y^T x$ is constant. $y^T x = [y_1 \dots y_n][x_1 \dots x_n]^T = y_1 x_1 + y_2 x_2 + \dots + y_n x_n = \text{constant}$

Example

glucose $\xrightarrow{\text{ATP} \rightarrow \text{ADP}}$ gluc_6P $\xleftrightarrow{\quad}$ fruc_6P $\xrightarrow{\text{ATP} \rightarrow \text{ADP}}$ fruc-1,6P



$$S = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & -1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

Matlab code starts here

```
-----
A = [-1 1 0 0 -1 1, 0 -1 1 0 0 0, 0 0 -1 1 -1 1]
null(A)
% if you scale, you get this vector
[ 2 1 1 0 0 1, 0 0 0 0 1 1, 1 1 1 1 0 0 ]

% first column y_1, second y_2, third y_3
% additional conservation relation
% -y_1 + 3*y_2 + 2*y_3 = [0 1 1 2 3 2]^T
```