基因组规模代谢的概念和构建

https://doi.org/10.27148/d.cnki.ghagu.2018.000017

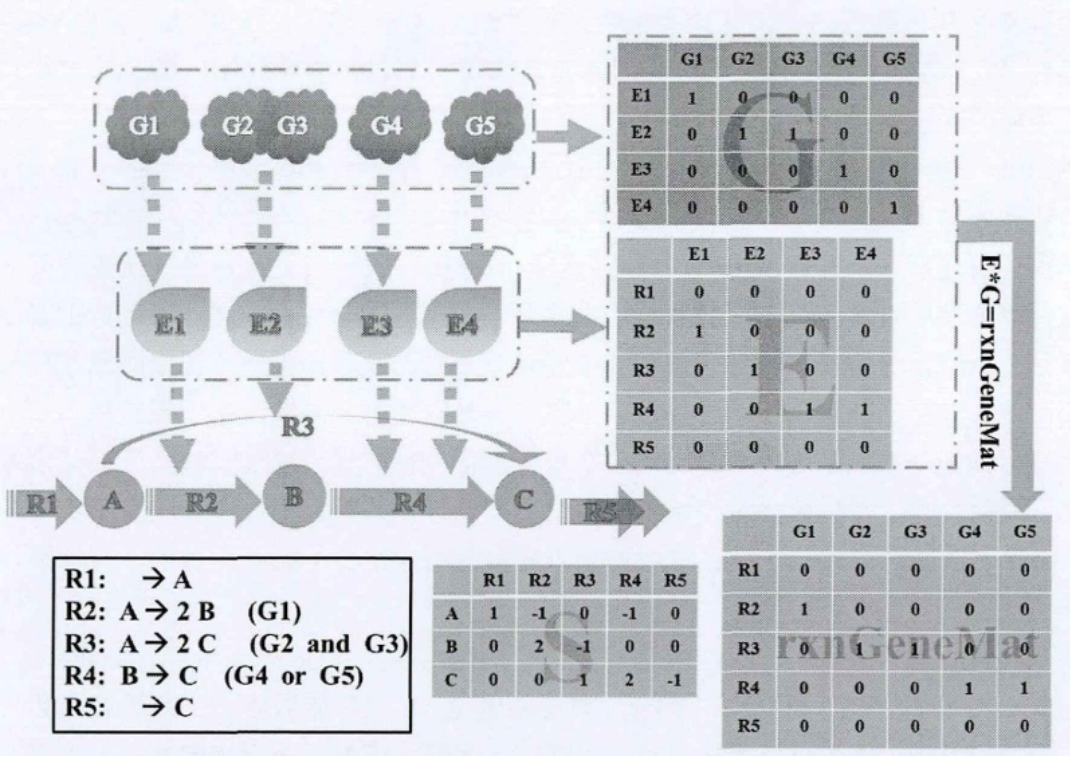
1. 系统生物学

> 1️⃣生物信息学：生物组学信息爆炸式增长，由此需要快速处理和挖掘，由此和计算机结合产生生物信息学
>
> 2️⃣系统生物学：基于高通量生物组学信息，借助计算机技术和数学工具
>
> 3️⃣基因组规模代谢网络模型：以菌株为例
>
> 1. 指该菌株内部所有代谢反应共同形成的系统模型
> 2. 包括代谢物+催化反应的酶+基因

2. 基因组规模代谢网络

2.1. 模型概念

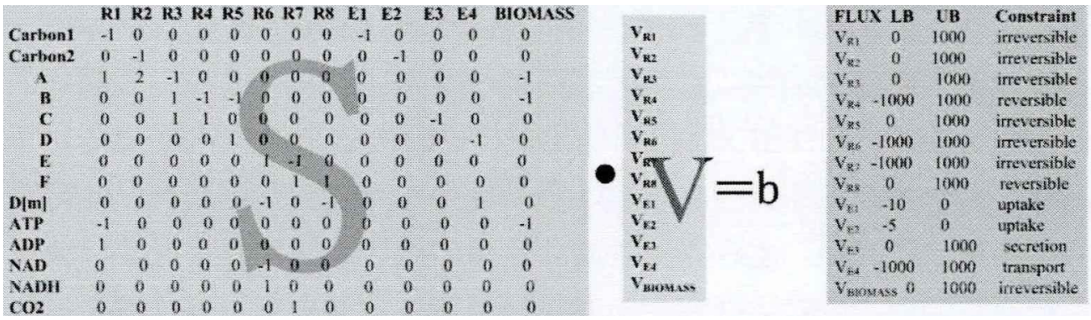
> 1️⃣核心：通过矩阵来反映GPR(基因-蛋白-反应)关系
>
> 2️⃣建模实例：注意，$\rightarrow A$的符号表示了底物的摄取



1. 基因-酶矩阵G：如图中的G_{4*5}，例如E_2行对应$G_2 = G_3 = 1$表示酶2的合成需要2和3基因
2. 酶-反应矩阵E：如图中的G_{5*4}，例如R_2行对应$E_1 = 1$表示反应2需要酶1的催化
3. 基因-反应矩阵$E * G$：就是把酶-反应矩阵*基因-酶矩阵，可以理解为基因催化了某个反应
4. 化学计量矩阵S
    - 一行一个代谢物，一列一个反应，交叉的数代表反应的计量系数

- 计量数中，负数代表是反应物，正数代表是生成物，0代表不参加反应

3️⃣针对化学计量矩阵的限制向量v



	R1	R2	R3	R4	R5	R6	R7	R8	E1	E2	E3	E4	BIOMASS
Carbon1	-1	0	0	0	0	0	0	0	-1	0	0	0	0
Carbon2	0	-1	0	0	0	0	0	0	0	-1	0	0	0
A	1	2	-1	0	0	0	0	0	0	0	0	0	-1
B	0	0	1	-1	-1	0	0	0	0	0	0	0	-1
C	0	0	1	1	0	0	0	0	0	0	-1	0	0
D	0	0	0	0	1	0	0	0	0	0	0	-1	0
E	0	0	0	0	0	1	-1	0	0	0	0	0	0
F	0	0	0	0	0	0	1	1	0	0	0	0	0
D[m]	0	0	0	0	0	-1	0	-1	0	0	0	1	0
ATP	-1	0	0	0	0	0	0	0	0	0	0	0	-1
ADP	1	0	0	0	0	0	0	0	0	0	0	0	0
NAD	0	0	0	0	0	-1	0	0	0	0	0	0	0
NADH	0	0	0	0	0	1	0	0	0	0	0	0	0
CO2	0	0	0	0	0	0	1	0	0	0	0	0	0

$\bullet V = b$

FLUX	LB	UB	Constraint
V_{R1}	0	1000	irreversible
V_{R2}	0	1000	irreversible
V_{R3}	0	1000	irreversible
V_{R4}	-1000	1000	reversible
V_{R5}	0	1000	irreversible
V_{R6}	-1000	1000	irreversible
V_{R7}	-1000	1000	irreversible
V_{R8}	0	1000	reversible
V_{E1}	-10	0	uptake
V_{E2}	-5	0	uptake
V_{E3}	0	1000	secretion
V_{E4}	-1000	1000	transport
$V_{BIOMASS}$	0	1000	irreversible

2.2. 模型构建

就是将细胞内全基因组规模下所有基因-蛋白-反应的关系转为数学模型

2.2.1. GSMM构建的数据库和工具

1️⃣GSMM(akaGEM)构建有关数据库

资源库	描述
MetaCyc/BioCyc	代谢路径/代谢途径数据库
NCBI	基因组数据库
KEGG	代谢途径数据库
Uniport	蛋白质数据库
BRENDA	酶活力学信息数据库
ENZYME	酶数据库
REACTOME	代谢途径数据库
MetRxn	代谢物和反应的标准化平台
BiGG	GEM数据库
BioModels	生物模型数据库
antiSMASH 3.0	次级代谢物生物信息门户

2️⃣GSMM构建有关自动化工具

自动构建自动建模构建	描述
RAVEN	自动构建自动 GEM 构建工具集
Model SEED	自动构建自动 GEM 构图构建平台
MetaNetX.org	基于基因组/重组自动化 GEM 自动构建和分析平台
CoReCo	基于基因局序列和多种物种的基础数据生成 GEM 构建
merlin	基于基因组信息 GEM 自动构建平台

2.2.2. 草图的构建

1️⃣自上而下：自全基因组出发→注释基因→找出基因表达的酶及其酶促反应，这一过程忽略了RNA的剪切，由此常用于原核生物

2️⃣自下而上：直接从数据库中找出目标菌株的蛋白

2.2.3. 草图的修剪

1️⃣填补代谢途径的空隙：使得中间产物产销平衡，有时甚至需要添加一些未证实的反应

2️⃣移除影响较小的代谢途径

3️⃣添加拟反应和拟代谢物

1. 拟反应(Pseudo Reactions)：模拟不易量化和不明确的反应，不对应特定酶促反应，而是填补代谢途径
2. 以下反应用拟反应构建
    - 生物量合成反应：模拟细胞生长和生物质积累的过程
    - 细胞维持能量消耗反应：模拟细胞为维持基本生命活动所消耗的能量，如ATP代谢

2.3. GSMM构建现状

2.3.1. 模型构建

1️⃣原核生物最完善的GSMM：大肠杆菌GSMM

2️⃣真核生物最完善的GSMM：酿酒酵母GSMM

2.3.2. 模型分析工具

Tool	URL
COBRA toolbox	opencobra.github.io/cobratoolbox
RAVEN toolbox	github.com/SysBioChalmers/RAVEN
CellNetAnalyzer	mpi-magdeburg.mpg.de/projects/cna/cna.html
FBA-SimVis	immersive-analytics.infotech.monash.edu/fbasimvis
OptFlux	wwwoptflux.org
COBRAjl	opencobra.github.io/COBRA.jl
Sybil	rdrr.io/cran/sybil
COBRApy	opencobra.github.io/cobrapy
CBMPy	cbmpy.sourceforge.net
SurreyFBA	sysbio.sbs.surrey.ac.uk/sfba
FASIMU	bioinformatics.org/fasimu
FAME	f-a-m-e.org

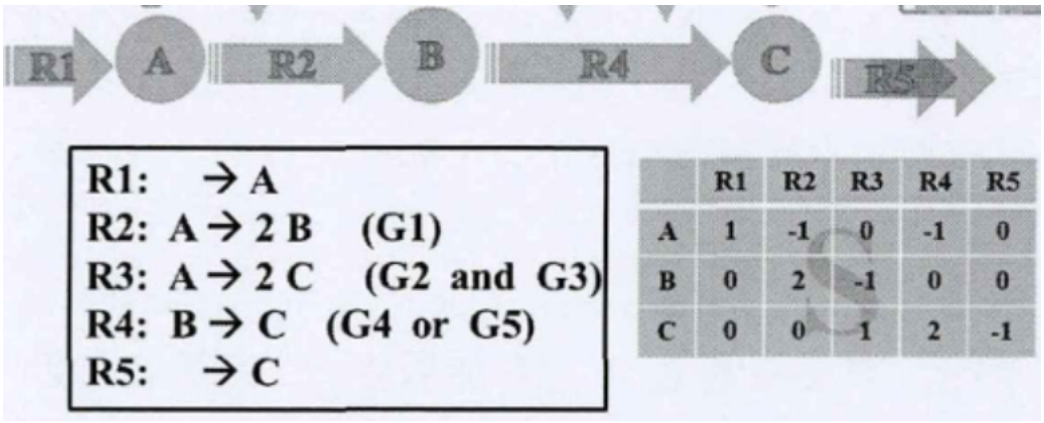
Tool	URL
Pathway Tools	bioinformatics.ai.sri.com/ptools
KBase	kbase.us

其实论文中有一处错误，COBRA工具箱不是Matlab写的，源代码是Go语言

2.4. GSMM仿真和算法

2.4.1. GSMM仿真过程

1️⃣对代谢网路的描述：化学计量矩阵S_{m*n}，有m个代谢物和n个反应



	R1	R2	R3	R4	R5
A	1	-1	0	-1	0
B	0	2	-1	0	0
C	0	0	1	2	-1

2️⃣对细胞生理状态的描述：速率向量V

1. 是$n*1$的列向量，包含了所有j个反应的速率，V_j表示第j个反应的速率
2. $\frac{dx_i}{dt} = S_{ij} \cdot v_j$表示，第$i$个物质的消耗速率，等于第$i$个物质第$j$个反应*第$j$个反应的速率

3️⃣明确反应中的约束

1. 反应的方向是否可逆
2. 上下限约束
   - 上限约束UB_j：最大反应速率
   - 下限约束LB_j：最大逆反应速率，对于不可逆反应这个值=0，对于可逆反应这个值通常是上限的负值
   - 通常这两个约束都强行设定为1000mmol/g DCW/h(每克干细胞重量每小时转换或产生的毫摩尔数)
3. 关于交换反应的限制
   - 交换反应是什么：描述代谢物进入或者离开细胞外周微环境的扩散作用
   - 特性：不需要酶参与，反应速率直接取决于代谢物的有无，所以干脆可以强行设定上下限

2.4.1.* 流量平衡分析(FBA)：GSMM代谢流量仿真

1 拟稳态的线性微分约束

1. 初代GSMM中有且仅有一个约束，就是要求细胞处于拟稳态
2. 拟稳态：胞内代谢物浓度保持不变，每种代谢物的生成速率与消耗速率平衡
3. 数学描述：假设细胞处于拟稳态，即$\frac{dx_i}{dt} = S_{ij} \cdot v_j = 0$其中$i \in m, j \in n$

$$\begin{bmatrix} \frac{dx_1}{dt} = S_{11} \cdot v_1 & \frac{dx_1}{dt} = S_{12} \cdot v_2 & \cdots & \frac{dx_1}{dt} = S_{1n} \cdot v_n \\ \frac{dx_2}{dt} = S_{21} \cdot v_1 & \frac{dx_2}{dt} = S_{22} \cdot v_2 & \cdots & \frac{dx_2}{dt} = S_{2n} \cdot v_n \\ \vdots & \vdots & \ddots & \vdots \\ \frac{dx_m}{dt} = S_{m1} \cdot v_1 & \frac{dx_m}{dt} = S_{m2} \cdot v_2 & \cdots & \frac{dx_m}{dt} = S_{mn} \cdot v_n \end{bmatrix} = 0$$

采纳更宽松的表达：$\begin{bmatrix} \sum_{j=1}^{n} S_{1j} \cdot v_j \\ \sum_{j=1}^{n} S_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^{n} S_{mj} \cdot v_j \end{bmatrix} = 0$，也就是物质在所有反应中产销平衡

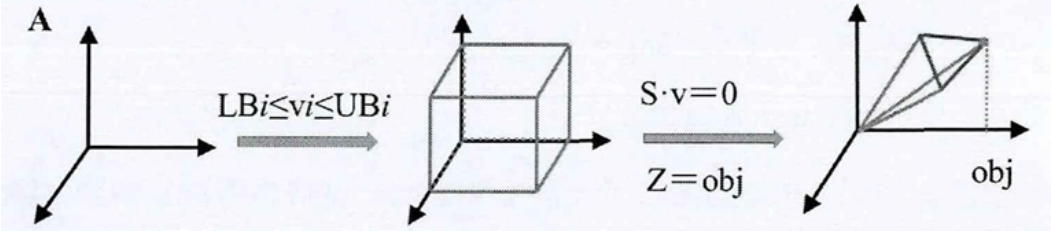
2 约束条件的解空间

1. 在以上多元线性微分方程的限制下，解空间是严格限制的
2. 解空间中每一点都代表一个解，可以表示为$V = \begin{bmatrix} v'_1 \\ v'_2 \\ \vdots \\ v'_n \end{bmatrix}$
3. 解的含义：其实就是一个速率向量，对应一个细胞的一个生理状态

3 目标函数：目标反应物的反应速率，也就是要求拟稳态下目标反应物反应速率最大

4 所以FBA到底是什么：

1. 一言蔽之：从解空间中找一个解，使得目标函数的反应流量最大/最小(最优解)
2. 最优解又称最优代谢流量分布
3. 可视化



5 一个典型的FBA：最大化细胞的比生长速率

1. 目标函数Max: $Z_{obj} = V_{biomass}$即细胞生物量的产生速率达到最大

2. 约束条件subject to: $$\begin{cases} \begin{bmatrix} \sum_{j=1}^{n} S_{1j} \cdot v_j \\ \sum_{j=1}^{n} S_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^{n} S_{mj} \cdot v_j \end{bmatrix} = 0 \\ \begin{bmatrix} LB_1 \\ LB_2 \\ \vdots \\ LB_n \end{bmatrix} \leq \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \leq \begin{bmatrix} UB_1 \\ UB_2 \\ \vdots \\ UB_n \end{bmatrix} \end{cases}$$

2.4.1.** FBA变种

1️⃣流量变化分析(FVA):

1. 表现：当FBA的目标反应最优时，其他所有反应速率的区间变化，反应了代谢路径的稳定性
2. 目标函数$\mathrm{Max/Min}$: V_j
3. 约束条件subject to: $$\begin{cases} Z_{obj} = V_{biomass} \\ \begin{bmatrix} \sum_{j=1}^{n} S_{1j} \cdot v_j \\ \sum_{j=1}^{n} S_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^{n} S_{mj} \cdot v_j \end{bmatrix} = 0 \\ \begin{bmatrix} LB_1 \\ LB_2 \\ \vdots \\ LB_n \end{bmatrix} \leq \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \leq \begin{bmatrix} UB_1 \\ UB_2 \\ \vdots \\ UB_n \end{bmatrix} \end{cases}$$

2️⃣吝啬流量分析(pFBA)

1. 限制：在拟稳态基础上，加一个细胞内催化反应酶的用量最省，由此得到的代谢流量分布通常比FBA更准确
2. 表现：目标函数最优，其余反应速率不再有变化区间，而是截取所有反应速率最小和
3. 目标函数Max: $\sum_{j=1}^{n} v_j$

4. 约束条件subject to:

$$\begin{cases} Z_{obj} = V_{biomass} \\ \begin{bmatrix} \sum_{j=1}^{n} S_{1j} \cdot v_j \\ \sum_{j=1}^{n} S_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^{n} S_{mj} \cdot v_j \end{bmatrix} = 0 \\ \begin{bmatrix} LB_1 \\ LB_2 \\ \vdots \\ LB_n \end{bmatrix} \leq \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \leq \begin{bmatrix} UB_1 \\ UB_2 \\ \vdots \\ UB_n \end{bmatrix} \end{cases}$$

2.4.1.*** ACHR算法(Artificial Centering Hit-and-Run)

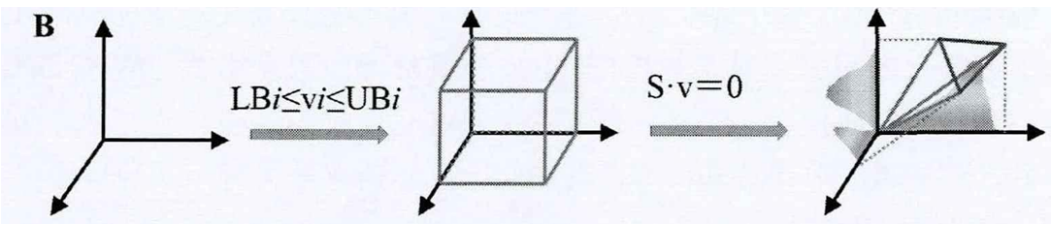
1 限制：假设处于拟稳态，同样得到约束条件

$$\text{subject to:} \begin{cases} \begin{bmatrix} \sum_{j=1}^{n} S_{1j} \cdot v_j \\ \sum_{j=1}^{n} S_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^{n} S_{mj} \cdot v_j \end{bmatrix} = 0 \\ \begin{bmatrix} LB_1 \\ LB_2 \\ \vdots \\ LB_n \end{bmatrix} \leq \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \leq \begin{bmatrix} UB_1 \\ UB_2 \\ \vdots \\ UB_n \end{bmatrix} \end{cases}$$

2 取样：

1. 不设定目标函数，而是设定从上述约束条件解空间中取样的点数
2. 每个点代表细胞的一个生理状态
3. 取样点数到一定多(2000个)，点就会收敛到解空间，从而得到解空间的大致轮廓

3 取样后的统计



1. 法1：找出每个代谢反应出现频数最多的反应速率，组成一个速率向量
2. 法2：计算每个反应的平均反应速率，组成一个速率向量

2.4.2. GSMM仿真算法

2.4.2.1. 双边优化：满足外部目标产物最大&内部细胞生长最大

1️⃣ 外层优化目标：最大化目标产品的生产

2️⃣ 内层优化目标：最大化细胞生长

3️⃣ 同时满足限制：

1. 强行设定底物摄取速率
2. 拟稳态
3. 实验的特定限制条件
4. 外层优化敲除反应(敲除反应数量不超过预设)

2.4.2.2. 典型仿真算法

1️⃣ OptKnock

1. 基本思想：基于双边优化，通过对GSMM突变(基因敲除)最大化目标产物的生产，同时必须满足一定的细胞生长
2. 小缺陷：算法是通过尝试所有突变策略来挑选最优敲除反应靶点，当要敲除的靶点大于3个时，计算量就是灾难性的

2️⃣ IdealKnock：在算法层面优化了OptKnock的时间复杂度，使得敲除10个以内靶点都可接受

3️⃣ OptReg：

1. 和OptKnock识别敲除靶点一样的原理，识别过表达靶点和限制表达的靶点
2. 反应靶点从0和1的二元关开状态，变成了0到1的连续状态
3. 算法时间复杂度较OptKnock增加
4. 识别过/限表达的算法还有：FSEOF/FVSEOF，OptForce，k-OptForce，APGC

4️⃣ OptSwap

1. 辅因子：协助酶进行催化反应的非蛋白质组分
2. 催化相同反应的不同酶
    - 酶的辅因子不同，导致酶不同，但是却能够催化同一化学反应
    - 典型例子如酶+NADH，酶+NADPH
3. OptSwap算法：基于双边优化，特异性识别辅因子特异性反应靶点，而这些靶点即使改变也不会影响细胞生长并且能提高目标产量

5️⃣ OptStrain和SimOptStrain

1. OptStrain：实现GSMM添加外源基因从而提高产量
2. SimOptStrain：同时仿真了外源基因的添加和基因的敲除

6️⃣ ReacKnock：将OptKnock的线性规划换成混合整数线性规划，能更快识别敲除靶点和列举同等最优解

2.4.2.3. 其他仿真算法

1. 基于最小代谢调整量(MOMA)的算法
    1. MOMA：也就是改造后的GSMM和改造前的尽量要相似
    2. 例如BiMOMA，MOMAKnock
2. 基于优化算法的仿真算法
    1. 基于遗传算法的OptGene：能不增加时间复杂度情况下识别更多敲除靶点
    2. 基于进化算法的CiED：
    3. 基于蜂群算法的BAFBA，DBFBA
3. 考虑其他生物学过程和因素的算法
    1. OptORF：在原有基础上增加转录限制
    2. EMILiO/CosMos：胁迫各自反应的通量增加来模拟基因上调的效果
    3. Redirector：引入负值权重(什么的权，原文又没说清楚)

2.4.2.3. 反应靶点-基因靶点的关联矛盾

1. 矛盾点
    1. GSMM仿真实现的都是反应靶点的识别，而代谢/基因工程改造的是基因靶点
    2. 反应靶点可对应多个基因靶点的调控，而一个基因靶点可以调控多个反应靶点
2. 两种解决方案：一言以蔽之，对传统GSMM进行逻辑转换
    1. LTM：从基因水平上对模型进行转换，加入不存在的物质(拟代谢物)，使得所有反应至多被一个基因催化
    2. gModel：从酶水平上对模型进行转换，所有反应都只和一个酶有关

3. GSMM仿真的应用

1. 最主要的两个应用
    1. 基于GSMM仿真，理解工程菌株中基因扰动或环境扰动对菌株表型的影响
    2. 基于GSMM仿真预测，设计出目标菌株
2. 应用步骤
    1. 准备好和菌株匹配的GSMM
    2. 在特定条件下，仿真GSMM的代谢流量分布
    3. 分析代谢流量分布