V20 Flux Balance Analysis + algorithms on top

- Metabolic networks are also scale-free

- Flux balance analysis (FBA)

FBA-based algorithms:

- MOMA

- OptKnock

- NetworkReducer

- High Flux Backbone

Topology of metabolic networks

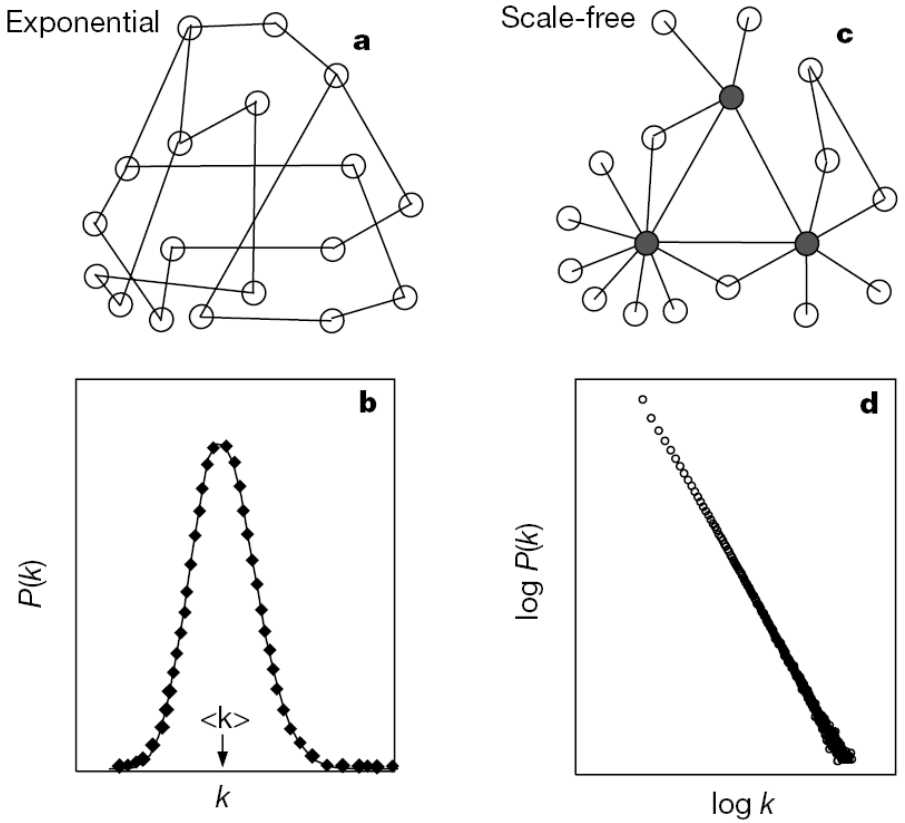
Review of 2 contrasting network topologies.

a, Representative structure of networks generated by the Erdös–Rényi model.

b, For a **random network,** $P(k)$ peaks strongly at $k = \langle k \rangle$ and decays exponentially for large k.

c, In the **scale-free network,** most nodes have only a few links, but a few nodes, called hubs (dark), have many links.

d, $P(k)$ for a scale-free network has no well-defined peak, and for large k it decays as a power-law, $P(k) \approx k^{-\gamma}$, appearing as a straight line with slope - on a log–log plot.



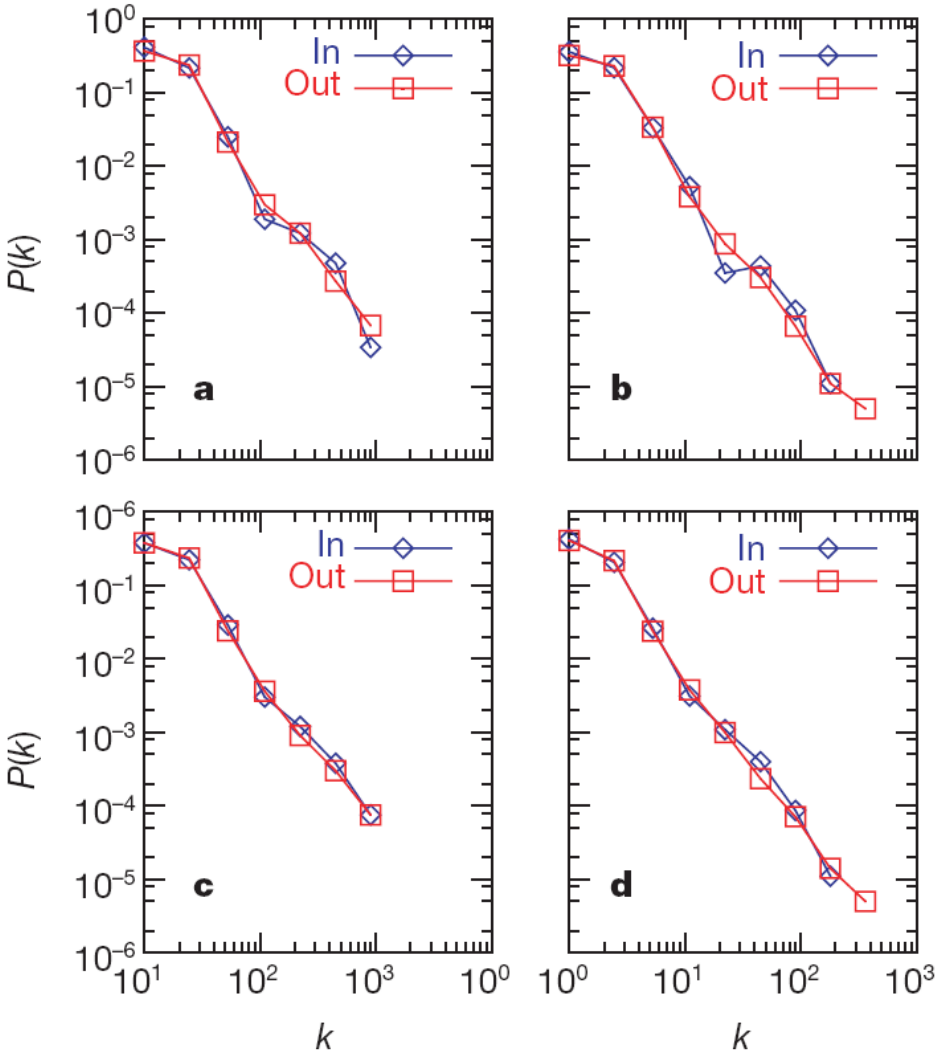
Jeong et al. Nature 407, 651 (2000)

Connectivity distributions *P(k)* for substrates

a, *Archaeoglobus fulgidus* (archae);
b, *E. coli* (bacterium);
c, *Caenorhabditis elegans* (eukaryote)
d, The connectivity distribution averaged over 43 organisms.

x-axis: metabolites participating in *k* reactions
y-axis (*P(k)*): number/frequency of such metabolites

log–log plot, counts separately the incoming (In) and outgoing links (Out) for each substrate.
k_{in} (k_{out}) corresponds to the number of reactions in which a substrate participates as a product (educt).



Jeong et al. Nature 407, 651 (2000)

Properties of metabolic networks

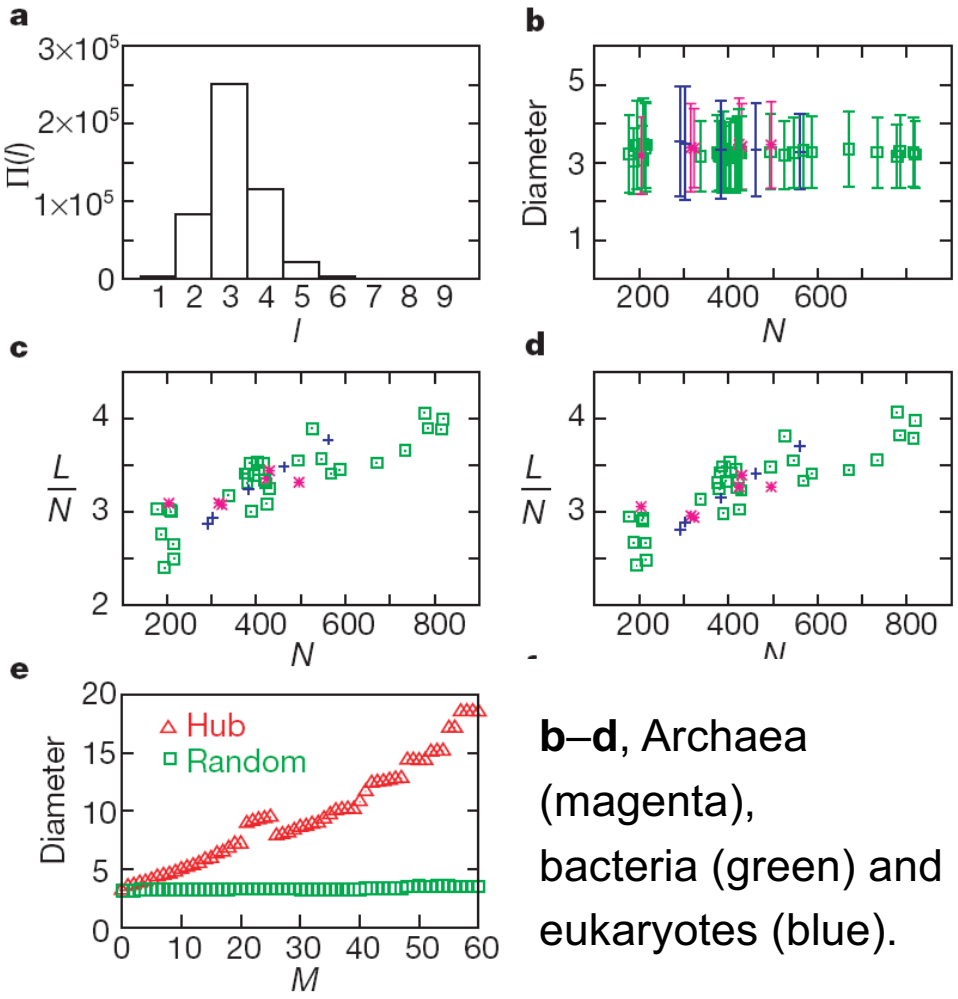
a, Histogram of biochemical pathway lengths, *l*, in *E. coli*.

b, Average path length (diameter) for each of 43 organisms.
N : # metabolites in each organism

c, **d**, Average number of incoming links (**c**) or outgoing links (**d**) per node.

e, Effect of substrate removal on metabolic network diameter of *E. coli*.

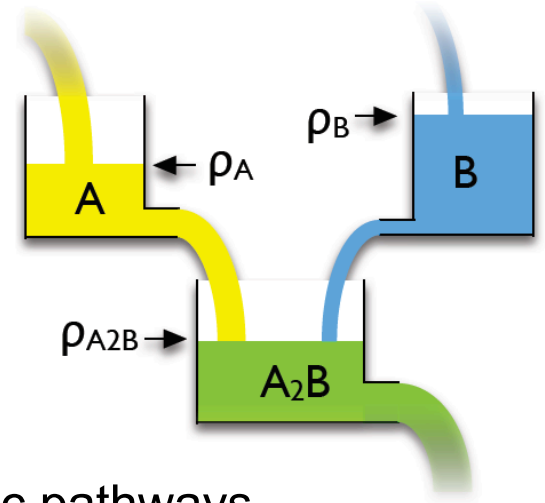
In the top curve (red) the most connected substrates are removed first. In the bottom curve (green) nodes are removed randomly.



b–**d**, Archaea (magenta), bacteria (green) and eukaryotes (blue).

Jeong et al. Nature 407, 651 (2000)

Flux balancing



Any chemical reaction requires **mass conservation**.

Therefore one may analyze metabolic systems
by requiring mass conservation.

Only required: knowledge about stoichiometry of metabolic pathways.

For each metabolite X_i :

$$dX_i/dt = V_{synthesized} - V_{used} + V_{transported\_in} - V_{transported\_out}$$

Steady state: concentrations are constant
=> flux in = flux out

$$\frac{dA_2B(t)}{dt} = G_{A_2B} - L_{A_2B} = 0$$

Flux balancing

Under **steady-state conditions**, the mass balance constraints in a metabolic network can be represented mathematically by the matrix equation:

$$\mathbf{S} \cdot v = 0$$

where

- the matrix **S** is the **stoichiometric matrix** and
- the vector *v* represents all **fluxes** in the metabolic network, including the internal fluxes and transport fluxes.

12.5 Flux Balance Analysis (FBA)

Since the number of metabolites is generally smaller than the number of reactions ($m < n$) the flux-balance equation is typically **underdetermined**.

-> There are generally multiple feasible
flux distributions that satisfy the mass balance constraints.

$$\mathbf{S} \cdot \mathbf{v} = \mathbf{0}$$

The set of solutions is confined to the **nullspace** of matrix **S**.

Null space: space of feasible solutions

Consider

$$\begin{pmatrix} 0 & 2 & 1 \\ 3 & -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

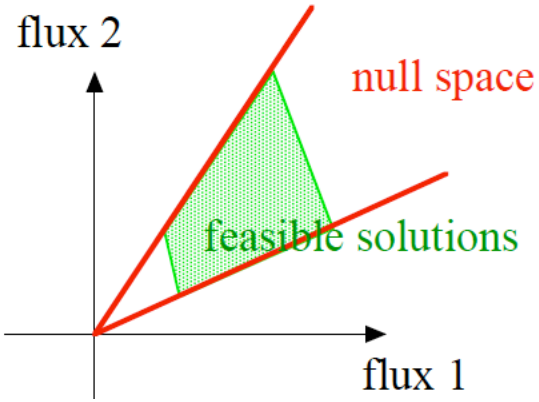
Corresponds to

$$\begin{aligned} 2x_2 + x_3 &= 0 \\ 3x_1 - x_2 + x_3 &= 0 \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} 2x_2 &= -x_3 \\ 2x_1 &= -x_3 \end{aligned}$$

=> only one free parameter: x_3

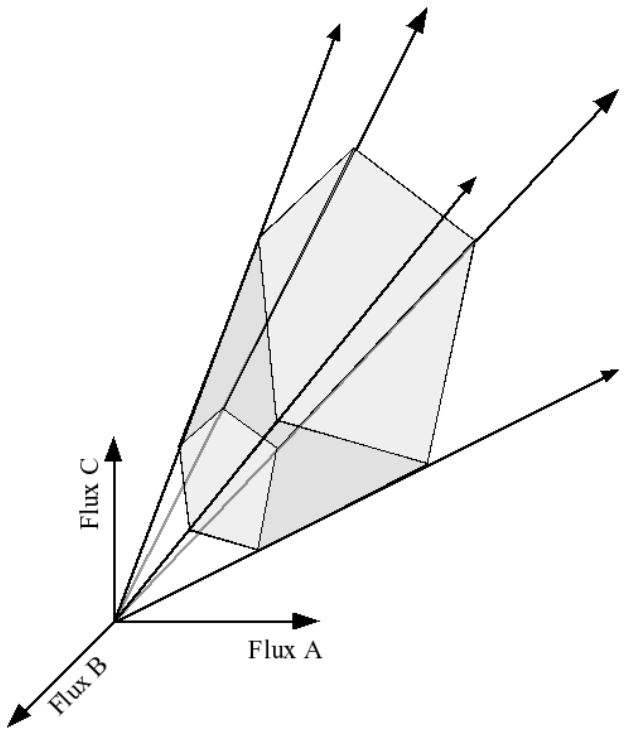
null space: $\vec{x} = \begin{pmatrix} -a \\ -a \\ 2a \end{pmatrix}$

Add inequalities for external fluxes
(here, e.g.: $x_3 \geq 0$)
=> **feasible** solutions for $a \geq 0$



Generally: null space is a cone,
constraints select part of it

Feasible solution set for a metabolic reaction network



Edwards & Palsson PNAS 97, 5528 (2000)

The steady-state operation of the metabolic network is restricted to the region within a **pointed cone**, defined as the feasible set.

The **feasible set** contains all flux vectors that satisfy the physicochemical constraints.

Thus, the feasible set defines the capabilities of the metabolic network.
All feasible metabolic flux distributions lie within the feasible set.

The **extreme pathways** (see V19) are the corner rays of this cone.

The **origin** (all fluxes = 0) is typically a valid flux distribution.

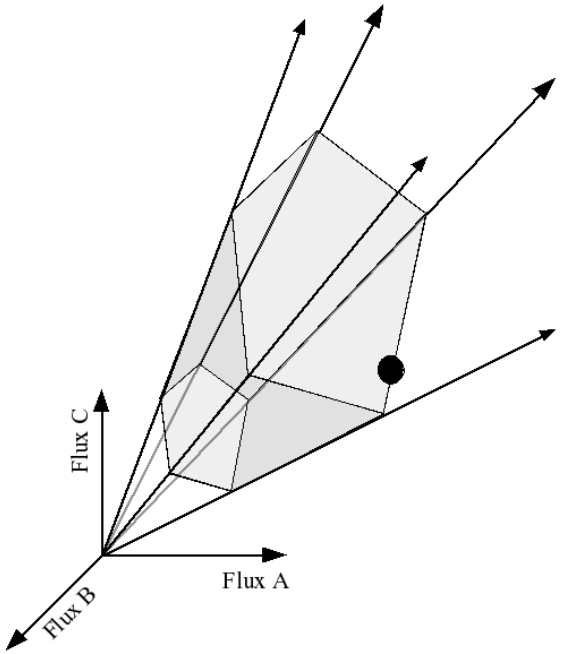
True biological flux

To find the „true“ biological flux in cells (→ e.g. Wittmann / UdS) one needs additional (often experimental) information that impose constraints

$$\alpha_i \leq v_i \leq \beta_i$$

on the magnitude of each individual metabolic flux.

The intersection of the nullspace and the region defined by those linear inequalities defines a region in flux space = the **feasible set of fluxes**.



In the limiting case, where all constraints on the metabolic network are known, such as the enzyme kinetics and gene regulation, the feasible set may be reduced to a single point. This single point must lie within the feasible set.

E.coli in silico

Best studied cellular system: *E. coli*.

In 2000, Edwards & Palsson constructed an *in silico* representation of *E.coli* metabolism.

There were 2 good reasons for this:

(1) genome of *E.coli* MG1655 was already completely sequenced,

(2) Because of long history of *E.coli* research, biochemical literature, genomic information, metabolic databases EcoCyc, KEGG contained biochemical or genetic evidence for every metabolic reaction included in the *in silico* representation. In most cases, there existed both.

Edwards & Palsson

PNAS 97, 5528 (2000)

Genes included in *in silico* model of *E.coli*

Table 1. The genes included in the *E. coli* metabolic genotype (21)

Central metabolism (EMP, PPP, TCA cycle, electron transport)	*aceA, aceB, aceE, aceF, ackA, acnA, acnB, acs, adhE, agp, appB, appC, atpA, atpB, atpC, atpD, atpE, atpF, atpG, atpH, atpI, cydA, cydB, cydC, cydD, cyoA, cyoB, cyoC, cyoD, dld, eno, fba, fbp, fdhF, fdnG, fdnH, fdnI, fdoG, fdoH, fdoI, frdA, frdB, frdC, frdD, fumA, fumB, fumC, galM, gapA, gapC_1, gapC_2, glcB, glgA, glgC, glgP, glk, glpA, glpB, glpC, glpD, gltA, gnd, gpmA, gpmB, hyaA, hyaB, hyaC, hybA, hybC, hycB, hycE, hycF, hycG, icdA, lctD, ldhA, lpdA, malP, mdh, ndh, nuoA, nuoB, nuoE, nuoF, nuoG, nuoH, nuoI, nuoJ, nuoK, nuoL, nuoM, nuoN, pckA, pfkA, pfkB, pflA, pflB, pflC, pflD, pgi, pgk, pntA, pntB, ppc, ppsA, pta, purT, pykA, pykF, rpe, rpiA, rpiB, sdhA, sdhB, sdhC, sdhD, sfcA, suoA, sucB, sucC, sucD, talB, tktA, tktB, tpiA, trxB, zwf, pgl* (30), *maeB* (30)
Alternative carbon source	*adhC, adhE, agaY, agaZ, aldA, aldB, aldH, araA, araB, araD, bglX, cpsG, deoB, fruK, fucA, fucI, fucK, fucO, galE, galK, galT, galU, gatD, gatY, glk, glpK, gntK, gntV, gpsA, lacZ, manA, melA, mtlD, nagA, nagB, nanA, pfkB, pgi, pgm, rbsK, rhaA, rhaB, rhaD, srlD, treC, xylA, xylB*
Amino acid metabolism	*adi, aldH, alr, ansA, ansB, argA, argB, argC, argD, argE, argF, argG, argH, argI, aroA, aroB, aroC, aroD, aroE, aroF, aroG, aroH, aroK, aroL, asd, asnA, asnB, aspA, aspC, avtA, cadA, carA, carB, cysC, cysD, cysE, cysH, cysI, cysJ, cysK, cysM, cysN, dadA, dadX, dapA, dapB, dapD, dapE, dapF, dsdA, gabD, gabT, gadA, gadB, gdhA, glk, glnA, gltB, gltD, glyA, goaG, hisA, hisB, hisC, hisD, hisF, hisG, hisH, hisI, ilvA, ilvB, ilvC, ilvD, ilvE, ilvG_1, ilvG_2, ilvH, ilvI, ilvM, ilvN, kbl, ldcC, leuA, leuB, leuC, leuD, lysA, lysC, metA, metB, metC, metE, metH, metK, metL, pheA, proA, proB, proC, prsA, putA, sdaA, sdaB, serA, serB, serC, speA, speB, speC, speD, speE, speF, tdcB, tdh, thrA, thrB, thrC, tnaA, trpA, trpB, trpC, trpD, trpE, tyrA, tyrA, tyrB, ygjG, ygjH, alaB* (42), *dapC* (43), *pat* (44), *prr* (44), *sad* (45), *methylthioadenosine nucleosidase* (46), *5-methylthioribose kinase* (46), *5-methylthioribose-1-phosphate isomerase* (46), *adenosyl homocysteinase* (47), *L-cysteine desulfhydrase* (44), *glutaminase A* (44), *glutaminase B* (44)
Purine & pyrimidine metabolism	*add, adk, amn, apt, cdd, cmk, codA, dcd, deoA, deoD, dgt, dut, gmk, gpt, gsk, guaA, guaB, guaC, hpt, mutT, ndk, nrdA, nrdB, nrdD, nrdE, nrdF, purA, purB, purC, purD, purE, purF, purH, purK, purL, purM, purN, purT, pyrB, pyrC, pyrD, pyrE, pyrF, pyrG, pyrH, pyrI, tdk, thyA, tmk, udk, udp, upp, ushA, xapA, yicP, CMP glycosylase* (48)
Vitamin & cofactor metabolism	*acpS, bioA, bioB, bioD, bioF, coaA, cyoE, cysG, entA, entB, entC, entD, entE, entF, epd, folA, folC, folD, folE, folK, folP, gcvH, gcvP, gcvT, gltX, glyA, gor, gshA, gshB, hemA, hemB, hemC, hemD, hemE, hemF, hemH, hemK, hemL, hemM, hemX, hemY, ilvC, lig, lpdA, menA, menB, menC, menD, menE, menF, menG, metF, mutT, nadA, nadB, nadC, nadE, ntpA, pabA, pabB, pabC, panB, panC, panD, pdxA, pdxB, pdxH, pdxJ, pdxK, pncB, purU, ribA, ribB, ribD, ribE, ribH, serC, thiC, thiE, thiF, thiG, thiH, thrC, ubiA, ubiB, ubiC, ubiG, ubiH, ubiX, yaaC, ygiG, nadD* (49), *nadF* (49), *nadG* (49), *panE* (50), *pncA* (49), *pncC* (49), *thiB* (51), *thiD* (51), *thiK* (51), *thiL* (51), *thiM* (51), *thiN* (51), *ubiE* (52), *ubiF* (52), *arabinose-5-phosphate isomerase* (22), *phosphopantothenate-cysteine ligase* (50), *phosphopantothenate-cysteine decarboxylase* (50), *phospho-pantetheine adenylyltransferase* (50), *dephosphoCoA kinase* (50), *NMN glycohydrolase* (49)
Lipid metabolism	*accA, accB, accD, atoB, cdh, cdsA, cls, dgkA, fabD, fabH, fadB, gpsA, ispA, ispB, pgpB, pgsA, psd, pssA, pgpA* (53)
Cell wall metabolism	*dcuA, ddlB, galF, galU, glmS, glmU, htrB, kdsA, kdsB, kdtA, lpxA, lpxB, lpxC, lpxD, mraY, msbB, murA, murB, murC, murD, murE, murF, murG, murI, rfaC, rfaD, rfaF, rfaG, rfaI, rfaJ, rfaL, ushA, glmM* (54), *lpoA* (55), *rfaE* (55), *tetraacyldisaccharide 4' kinase* (55), *3-deoxy-D-manno-octulosonic-acid 8-phosphate phosphatase* (55)
Transport processes	*araE, araF, araG, araH, argT, aroP, artI, artJ, artM, artP, artQ, brnQ, cadB, chaA, chaB, chaC, cmtA, cmtB, codB, crr, cycA, cysA, cysP, cysT, cysU, cysW, cysZ, dctA, dcuA, dcuB, dppA, dppB, dppC, dppD, dppF, fadL, focA, fruA, fruB, fucP, gabP, galP, gatA, gatB, gatC, glnH, glnP, glnQ, glpF, glpT, gltI, gltK, gltL, gltP, glt5, gntT, gpt, hisJ, hisM, hisP, hisQ, hpt, kdpA, kdpB, kdpC, kgtP, lacY, lamB, livF, livG, livH, livJ, livK, livM, lktP, lysP, malE, malF, malG, malK, malX, manX, manY, manZ, melB, mglA, mglB, mglC, mtlA, mtr, nagE, nanT, nhaA, nhaB, nupC, nupG, oppA, oppB, oppC, oppD, oppF, panF, pheP, pitA, pitB, pnuC, potA, potB, potC, potD, potE, potF, potG, potH, potI, proP, proV, proW, proX, pstA, pstB, pstC, pst5, ptsA, ptsG, ptsI, ptsN, ptsP, purB, putP, rbsA, rbsB, rbsC, rbsD, rhaT, sapA, sapB, sapD, sbp, sdaC, srlA_1, srlA_2, srlB, tdcC, tnaB, treA, treB, trkA, trkG, trkH, tsx, tyrP, ugpA, ugpB, ugpC, ugpE, uraA, xapB, xylE, xylF, xylG, xylH, fruF* (56), *gntS* (57), *metD* (43), *pnuE* (49), *scr* (56)

Edwards & Palsson

PNAS 97, 5528 (2000)

E.coli in silico – Flux balance analysis

Define $\alpha_i = 0$ for **irreversible** internal fluxes,

$\alpha_i = -\infty$ for **reversible** internal fluxes (use biochemical literature)

Transport fluxes for PO_4^{2-}, NH_3, CO_2, SO_4^{2-}, K^+, Na^+ were unrestrained.

For other metabolites, $0 < v_i < v_i^{max}$ except for those that are able to leave the metabolic network (i.e. acetate, ethanol, lactate, succinate, formate, pyruvate etc.)

Find particular metabolic flux distribution in feasible set by **linear programming**.

LP finds a solution that **minimizes** a particular metabolic **objective** –*Z* (subject to the imposed constraints) where e.g.

$$Z = \sum c_i \cdot v_i = \langle \mathbf{c} \cdot \mathbf{v} \rangle$$

In FBA, c_i are the (known) coefficients of the optimization goal.

Edwards & Palsson, PNAS 97, 5528 (2000)

E.coli in silico – Flux balance analysis

In the case of biomass maximization, vector ***c*** is an all-zero vector except for a one (1.0) in the position corresponding to the biomass reaction:

$$Z = \sum c_i \cdot v_i = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \end{pmatrix} \begin{pmatrix} v_{bio} \\ v_1 \\ v_2 \\ \dots \\ v_N \end{pmatrix}$$

What is the **biomass reaction**?

(Montezano *et al*.) used the mixture on the right that reflects the actual composition of cells of *Mycobacterium tuberculosis*.

$$\begin{aligned} & 0.214\ \text{PROTEIN} \\ + \ & 0.036\ \text{RNA} \\ + \ & 0.022\ \text{DNA} \\ + \ & 0.050\ \text{SMALLMOLECULES} \\ + \ & 0.006\ \text{PE} \\ + \ & 0.016\ \text{TAGbio} \\ + \ & 0.040\ \text{PIMs} \\ + \ & 0.186\ \text{LAM} \\ + \ & 0.208\ \text{MAPC} \\ + \ & 0.035\ \text{P} - \text{L} - \text{GLX} \\ + \ & 0.007\ \text{CL} \\ + \ & 0.054\ \text{LM} \\ + \ & 47\ \text{ATP} \\ = \ & 1.0\ \text{BIOMASS} + 47\ \text{ADP} + 47\ \text{PI} \end{aligned}$$

Montezano et al (2015) PLoS ONE 10(7): e0134014.

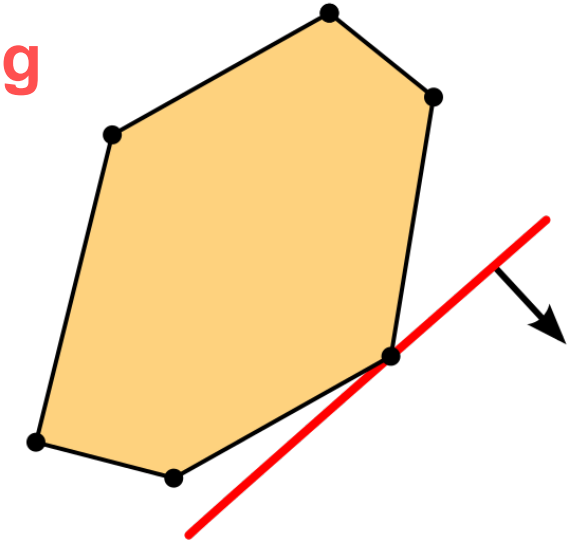
Linear programming

Linear programming is a technique for the **optimization** of a **linear objective function**, subject to linear equality and inequality **constraints**.

Its **feasible region** is a convex polytope, which is a set defined as the intersection of finitely many half spaces, each of which is defined by a linear inequality.

Its **objective function** is a real-valued linear function defined on this polyhedron.

A linear programming algorithm finds a point in the polyhedron where this function has the smallest (or largest) value if such a point exists.



A pictorial representation of a simple linear program with 2 variables (x and y-axes) and 6 inequalities (borders).
The set of feasible solutions is depicted in yellow and forms a polygon, a 2-dimensional polytope.
The linear **cost function** is represented by the red line and the arrow:
The arrow indicates the direction in which we are optimizing.

www.wikipedia.org

Linear programming

Linear programs are problems that can be expressed in canonical form as

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^{\mathrm{T}}\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \leq \mathbf{b} \\ \text{and} \quad & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

where **x** represents the vector of variables (to be determined),
c and **b** are vectors of (known) coefficients,
A is a (known) matrix of coefficients, and $(.)^{\mathrm{T}}$ is the matrix (vector) transpose.

The expression to be maximized or minimized is called the **objective function** ($\mathbf{c}^{\mathrm{T}}\mathbf{x}$ in this case).

The inequalities $A\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq \mathbf{0}$ are the **constraints** which specify a convex polytope over which the objective function is to be optimized.

www.wikipedia.org

Integer linear programming

Linear programming problems can be solved efficiently in polynomial time, e.g. by **Karmarkar's** algorithm.

If all unknown variables are required to be integers, then the problem is called an integer programming (IP) or integer linear programming (ILP) problem.

In contrast to linear programming problems,
integer programming problems are in many practical situations NP-hard.

The **branch and bound algorithm** is one type of algorithm to solve ILP problems.

www.wikipedia.org

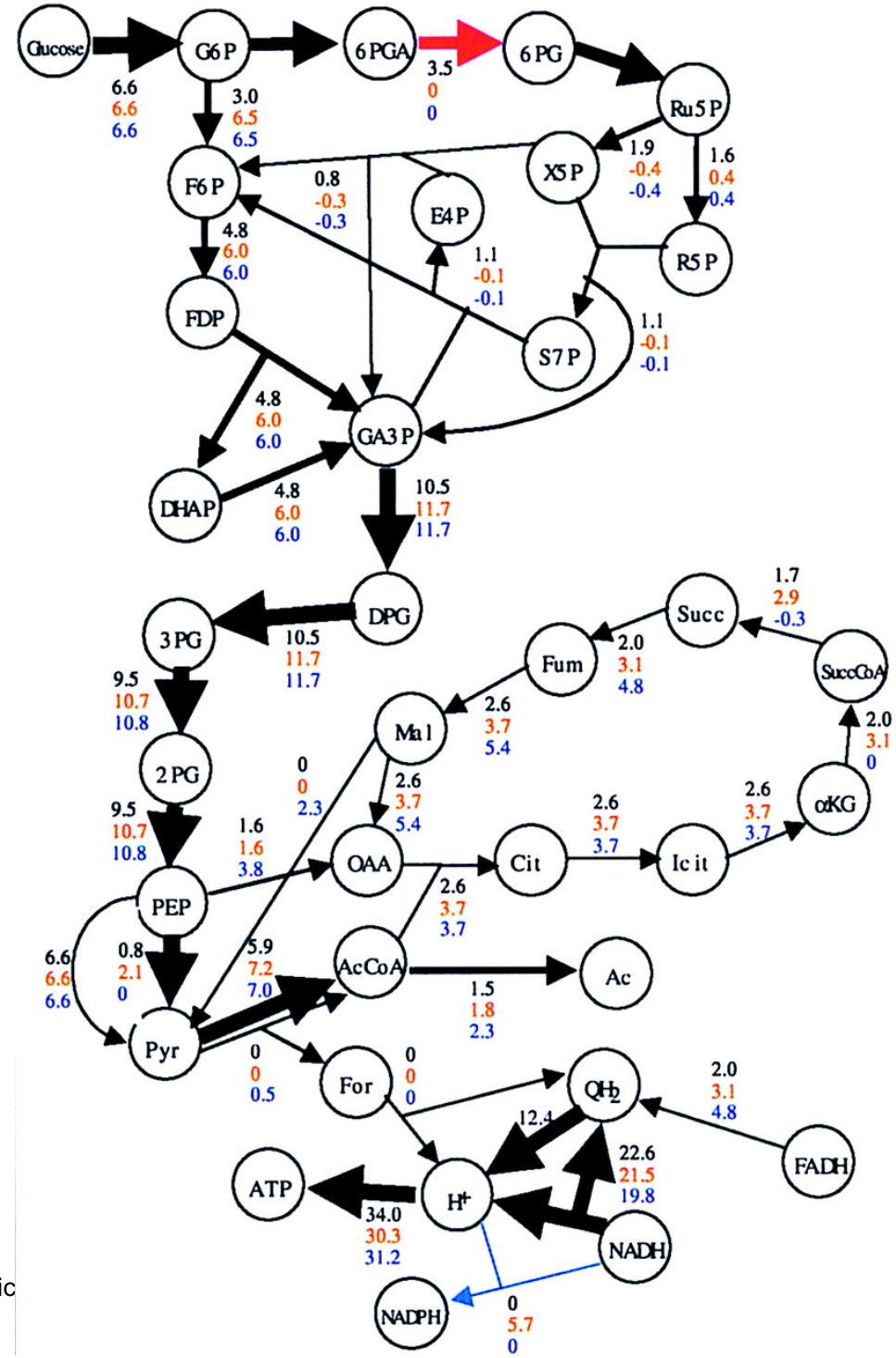
Rerouting of metabolic fluxes

(Black) Flux distribution for the wild-type.

(**Red**) *zwf*- mutant. Biomass yield is 99% of wild-type result.

(**Blue**) *zwf*- *pnt*- double mutant. Biomass yield is 92% of wildtype result.

Note how *E.coli in silico* circumvents removal of one critical reaction (**red arrow**) by increasing the flux through the alternative G6P → P6P reaction.



Edwards & Palsson PNAS 97, 5528 (2000)

E.coli in silico

Examine **changes** in the **metabolic capabilities** caused by hypothetical **gene deletions.**

To simulate a gene deletion, the flux through the corresponding enzymatic reaction is restricted to zero.

Compare optimal value of mutant (Z_{mutant}) to the „wild-type“ objective Z

$$\frac{Z_{mutant}}{Z}$$

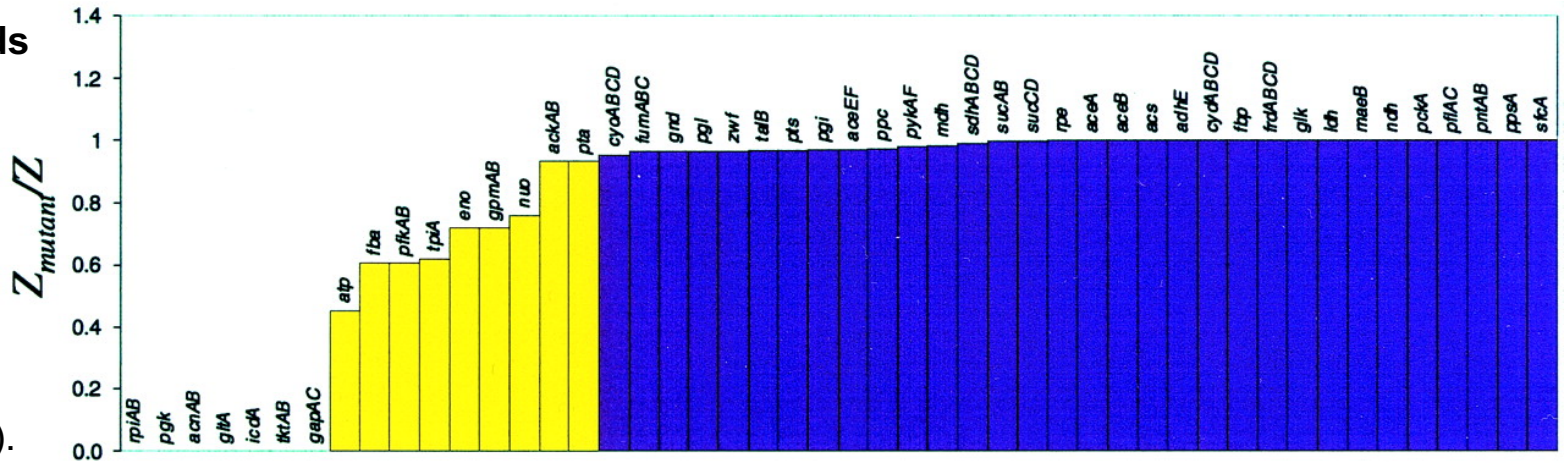
to determine the systemic effect of the gene deletion.

Edwards & Palsson

PNAS 97, 5528 (2000)

Gene deletions in central intermediary metabolism

Maximal biomass yields on glucose for all possible single gene deletions in the central metabolic pathways (gycolysis, pentose phosphate pathway (PPP), TCA, respiration).



The results were generated in a simulated aerobic environment with glucose as the carbon source. The transport fluxes were constrained as follows:
glucose = 10 mmol/g-dry weight (DW) per h; oxygen = 15 mmol/g-DW per h.

The maximal yields were calculated by FBA with the objective of maximizing growth.

Yellow bars: gene deletions that reduced the maximal biomass yield of Z_{mutant} to less than 95% of the *in silico* wild type Z_{wt}.

Edwards & Palsson PNAS 97, 5528 (2000)

Interpretation of gene deletion results

The essential gene products were involved in

- the 3-carbon stage of glycolysis,
- 3 reactions of the TCA cycle, and
- several points within the pentose phosphate pathway (PPP).

The remainder of the central metabolic genes could be removed while *E.coli in silico* maintained the potential to support cellular growth.

Edwards & Palsson PNAS 97, 5528 (2000)

E.coli in silico – validation

+ and – means growth or no growth.

± means that suppressor mutations have been observed that allow the mutant strain to grow.

4 virtual growth media:

glc: glucose, gl: glycerol, succ: succinate, ac: acetate.

In 68 of 79 cases, the prediction was consistent with exp. predictions.

Red and yellow circles: predicted mutants that eliminate or reduce growth.

Edwards & Palsson

PNAS 97, 5528 (2000)

Table 2. Comparison of the predicted mutant growth characteristics from the gene deletion study to published experimental results with single mutants

Gene	glc	gl	succ	ac
aceA	+/+		+/+	−/−
aceB				−/−
*aceEF**	−/+			
ackA				+/+
acn	−/−			−/−
acs				+/+
cyd	+/+			
cyo	+/+			
eno†	−/+	−/+	−/−	−/−
fbaI	−/+			
fbp	+/+	−/−	−/−	−/−
frd	+/+		+/+	+/+
gap	−/−	−/−	−/−	−/−
glk	+/+			
gltA	−/−			−/−
gnd	+/+			
idh	−/−			−/−
mdh††	+/+	+/+	+/+	
ndh	+/+	+/+		
nuo	+/+	+/+		
pfk†	−/+			
pgi‡	+/+	+/−	+/−	
pgk	−/−	−/−	−/−	−/−
pgl	+/+			
pntAB	+/+	+/+	+/+	
ppc§	±/+	−/+	+/+	
pta				+/+
pts	+/+			
pyk	+/+			
rpi	−/−	−/−	−/−	−/−
sdhABCD	+/+		−/−	−/−
sucAB	+/+		−/+	−/+
tktAB	−/−			
*tpi***	−/+	−/−	−/−	−/−
unc	+/+		±/+	−/−
zwf	+/+	+/+	+/+	

Summary - FBA

FBA analysis constructs the **optimal network utilization** simply using the stoichiometry of metabolic reactions and capacity constraints.

For *E.coli,* the *in silico* results are mostly **consistent** with experimental data.

FBA shows that the *E.coli* metabolic network contains relatively **few critical gene products** in central metabolism.
However, the ability to adjust to different environments (growth conditions) may be diminished by gene deletions.

FBA identifies „**the best**“ the cell can do, not how the cell actually behaves under a given set of conditions. Here, survival was equated with growth.

FBA does not directly consider **regulation** or regulatory constraints on the metabolic network. These can be treated separately.

Edwards & Palsson PNAS 97, 5528 (2000)

12.5.1 Gene knock-outs: MOMA algorithm

As just shown, FBA can also predict phenotypes associated with genetic manipulations.

To realize the effects of a **gene knockout** in FBA calculations,

one simply sets the entries of the stoichiometric matrix related to the respective protein to zero and then obtains an optimal flux by LP.

This approach assumes that the mutant bacteria also adopt an optimal metabolic state,

although these artificially generated strains have not been exposed to the typical **evolutionary pressure** that formed the metabolic profile of the wild-type.

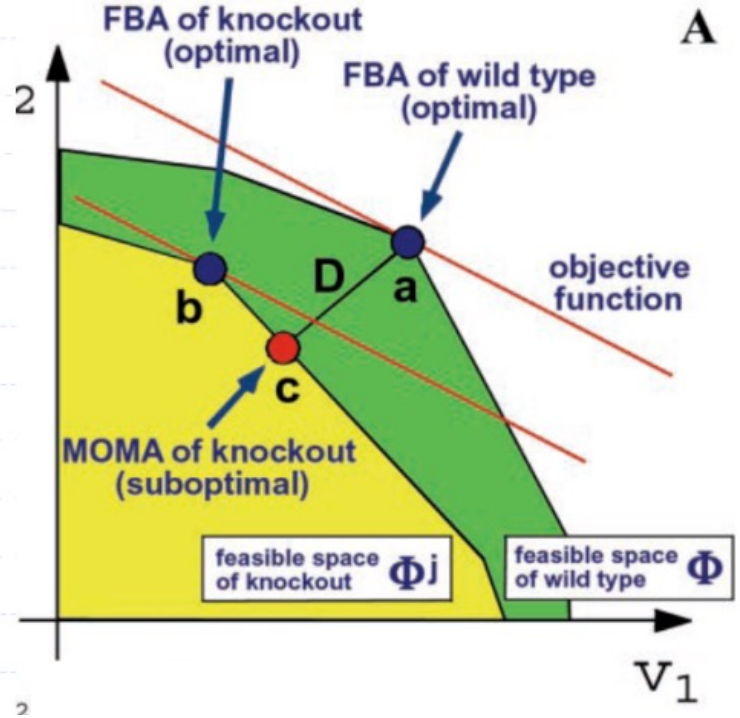
Segre D, Vitkup D, Church GM (2002)
PNAS 99, 15112-15117.

12.5.1 Gene knock-outs: MOMA algorithm

To characterize the flux states of mutants, Church and collagues formulated the method **MOMA** = „**minimization of metabolic adjustment**“.

MOMA applies the same stoichiometric constraints as FBA but does not assume that gene knock-out mutants will show optimal growth flux.

Idea behind MOMA: in the beginning, a mutant will likely possess a suboptimal flux distribution that lies in between the wild-type optimum (**a**) and the mutant optimum (**b**).



MOMA approximates this intermediate suboptimal state by assuming that the flux values in the mutant will initially take on values that match those of the wild-type optimum as closely as possible.

Segre D, Vitkup D, Church GM (2002) *PNAS* 99, 15112-15117.

12.5.1 Gene knock-outs: MOMA algorithm

To predict a metabolic phenotype,

MOMA determines a flux vector **v** in the flux space Φ of a mutant

with smallest Euclidian distance

from a given flux vector **w** for the wild-type organism.

This means that: $$D(w,x) = \sqrt{\sum_{i=1}^{N}(w_i - v_i)^2} = \sqrt{\sum_{i=1}^{N} w_i^2 - 2w_i v_i + x_i^2}$$

should be minimized.

Minimizing *D* is equivalent to minimizing the square of *D.*

Constant terms (the wild-type flux w_i^2) can be left out from the objective function.

Segre D, Vitkup D, Church GM (2002)
PNAS 99, 15112-15117.

12.5.1 Gene knock-outs: MOMA algorithm

With **Q** as the $n \times n$ unit matrix and **L** set to **–w,** this criterion is equivalent to a quadratic programming problem where the aim is to minimize:

$$f(x) = \mathbf{L} \cdot \mathbf{v} + \frac{1}{2}\mathbf{v}^T \mathbf{Q}\mathbf{v}$$

under a set of linear constraints.

The vector **L** of length N and the $N \times N$ matrix **Q** define the linear and quadratic part of the objective function, respectively, and $\mathbf{v}^T$ represents the transpose of **v**.

Flux predictions made by MOMA were

reported to show good correlation to experimental findings.

Segre D, Vitkup D, Church GM (2002)
PNAS 99, 15112-15117.

12.5.1 OptKnock algorithm

In **genetic strain optimization**, the aim can also be to **maximize the yield** of a particular **chemical compound**.

This can also be formulated as a linear programming problem, just like in FBA.

There exist several bi-level strain design approaches that employ **mixed-integer programming** (MIP) to find the mutations required to obtain the **largest synthesis yields of a chemical**.

Such bi-level MIP methods involve an **"outer" problem** and an **"inner" problem**.

In the outer problem, an engineering objective function (selection of **optimal mutant strains**) is optimized.

In the inner problem, a cellular objective function is optimized such as **maximizing** the **total flux** via FBA and linear programming.

As one representative of this class of algorithms, we will discuss the **OptKnock** algorithm

Burgard AP, Pharkya P, Maranas CD (2003) *Biotechnology and Bioengineering* 84, 647-57.

12.5.1 OptKnock algorithm

The aim of OptKnock is to over-produce desired chemicals, e.g. in *E. coli*.

Given a fixed amount of glucose uptake, the cellular objective can be to **maximize the yield of biomass**.

The effects of gene deletions are modeled by incorporating binary variables y_j into the FBA framework that describe whether reaction *j* is active or not :

$$y_j = \begin{cases} 1 \text{ if reaction flux } v_j \text{ is active} \\ 0 \text{ if reaction flux } v_j \text{ is not active}, \forall j \in M \end{cases}$$

The constraint:

$$v_j^{min} \cdot y_j \leq v_j \leq v_j^{max} \cdot y_j, \forall j \in M$$

guarantees that reaction flux v_j is set to zero only in cases where variable y_j is zero.

When y_j is equal to 1, v_j can adopt values between v_j^{min} and v_j^{max}.

The authors determined v_j^{min} and v_j^{max} by minimizing and subsequently maximizing every reaction flux subject to the constraints from the primal problem.

Burgard AP, Pharkya P, Maranas CD (2003) *Biotechnology and Bioengineering* 84, 647-57.

12.5.1 OptKnock algorithm

If biomass formation is the cellular objective, the best gene/reaction knockouts may be modeled mathematically as the following bilevel mixed-integer optimization task:

$$maximize \;\; v_{chemical} \quad (OptKnock - outer\ problem)$$

$$whereby\ y_j\ is\ subject\ to\ y_j \in \{0,1\}\ \forall j \in M, \sum_{j \in M}(1 - y_j) \leq K\ and$$

$$[maximize \;\; v_{biomass} \quad (Primal - inner\ problem)$$

$$whereby\ v_j\ is\ subject\ to \sum_{j=1}^{M} S_{ij} v_j = 0$$

$$v_{pts} + v_{glk} = v_{glc-uptake}$$

$$v_{atp} \geq v_{atp-main}$$

$$v_{biomass} \geq v_{biomass}^{target}$$

$$v_j^{min} \cdot y_j \leq v_j \leq v_j^{max} \cdot y_j, \forall j \in M]$$

K : maximal number of gene knockouts allowed.

The vector v holds both internal and transport reactions.

v_j : flux of reaction *j*

v_{glc_uptake} implements the glucose uptake scenario.

v_{pts} : uptake of glucose through phosphotransferase system , v_{glk} : synthesis of glucose by glucokinase.

v_{atp_main} : lower flux threshold keeping ATP level constant in non-growth-conditions

$v_{biomass}^{target}$: minimum level of biomass production.

12.5.1 OptKnock algorithm

Solving this two-stage optimization problem in a reasonable time can be challenging due to

- the high dimensionality of the flux space (the system implemented by the authors contained over 700 reactions) and
- the two nested optimization problems.

To overcome this, the authors turned the linear programming problem into an optimization problem.

Palsson and co-workers applied OptKnock to genome-scale metabolic models of *E. coli* wild-type and mutants followed by adaptive evolution of the engineered strains.

They managed to design bacterial production strains that produced **more lactate** than wild-type *E. coli* (Fong *et al.* 2005).

Burgard AP, Pharkya P, Maranas CD (2003) *Biotechnology and Bioengineering* 84, 647-57.

Compress genome-scale models: Network Reducer

Detailed genome-scale metabolic models contain thousands of metabolites and reactions. Their interpretation and application of the EP method is difficult.

Thus, one wishes to reduce genome scale models to **„core“** models of **lower complexity** but having the **same key elements** and/or key functional features.

One such method is the network reduction algorithm **NetworkReducer**.

It can simplify an input large-scale metabolic network to a smaller subnetwork whereby desired properties of the larger network are kept (Erdrich *et al.* 2015).

As in FBA, one consider vectors **v** of net reaction rates that fulfil $S \cdot v = 0$.

The fluxes **v** satisfying this equation form the null space of S. Its dimensionality may also be termed the number of **degrees of freedom** (dof) and is given by

$$dof = n - rank(S)$$

where *n* is the number of reactions in the system.

Specifications of Network Reducer

A key property of the algorithm is how it treats desired (protected) functions and phenotypes.

(a) PM : set of „**protected metabolites**“ that **must be kept** in the reduced network.

(b) PR : set of „**protected reactions**“ that must be kept in the reduced network.

(c) Protected **functions** (e.g. production of a chemical) and **phenotypes** are characterized by appropriate inequalities.

(d) The reduced network may not have fewer **degrees of freedom** (dof) than a predefined minimum number: $dof \geq dof_{min}$.

(e) A specified **minimal number of reactions** must be kept ($n \geq n_{min}$).

Network Reducer

Each protected functionality (there are *s* of them in total) is formulated by a respective set of linear equalities/inequalities,
$D_k v \leq d_k, k = 1 \ldots s$.

The network reduction algorithm first checks the **feasibility** of the protected reactions in the input network.

Then, a loop tries to iteratively discard non-protected reactions unless this violates any of the desired conditions (a) - (e).

To decide on the order of this process, the algorithm computes for each removable (non-protected) reaction *i* the feasible flux ranges.

Let F_i^k denote the **flux range** of reaction *i* under the protected function *k, k* = 1…*s*.

From this, the union F_i of all flux ranges is formed:
$F_i = \bigcup_{k=1}^{s} F_i^k$

Network Reducer

Essential reactions possess an entirely positive or entirely negative flux range F_i^k for any of the desired functionalities *k.*

Such essential reactions are deleted from the list of removable reactions.

From the current set of removable reactions, the next candidate reaction to be discarded is the reaction with overall **smallest flux range** F_i.

It can be safely assumed that a considerable amount of flux variability remains in the network after deleting this reaction.

After discarding a reaction, one needs to test the **feasibility** of the protected functions (condition (c)), protected reactions and of protected metabolites.
If any of these conditions is not fulfilled, then the reaction that was just deleted is reinserted and labeled as non-removable.

Then one continues with the reaction having the second smallest overall range of fluxes F_i.

Network Reducer

After deleting a reaction, the flux ranges are recomputed in the next iteration.

The main loop of network pruning terminates when no additional reaction can be removed without violating any of conditions (a) - (e).

Finally, **unconnected metabolites** in the reduced network that do not participate in any of the remaining reactions are deleted from the network.

In a post-processing step, the network can be (optionally) **compressed** further without loosing degrees of freedom.

For example, reaction sets or enzyme sets belonging to a **linear chain** of reactions can be combined into a single reaction with collapsed stoichiometries.

Compression does not affect protected reactions and metabolites.

Application of NetworkReducer

Klamt and co-workers applied NetworkReducer to a genome-scale metabolic model of *E. coli* with 2384 reactions.

The algorithm pruned this model to a reduced model with 105 reactions.

This is close to a manually constructed core model of *E.coli* that contains 88 reactions.

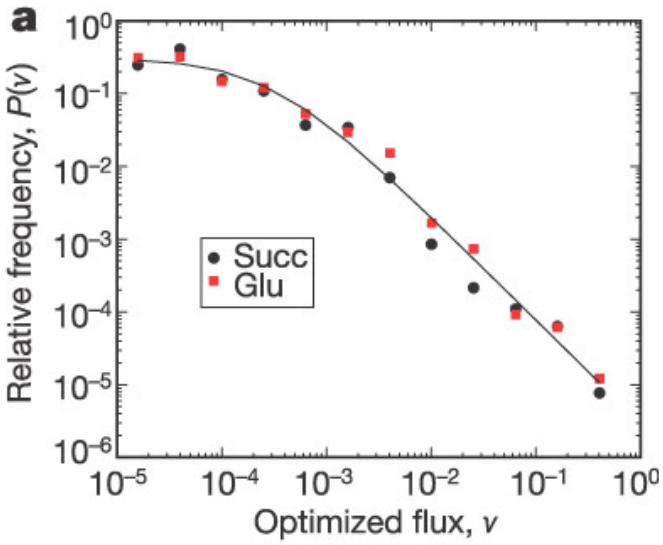
	***E. coli* genome-scale model**	***E. coli* pruned model**	***E. coli* pruned and compressed model**	***E. coli* core model of Orth et al.**
# reactions	2384	455	105	88
# internal metabolites	1669	438	85	69
# external metabolites	305	33	33	17
degrees of freedom	753	26	26	24
μ_{max} (aerobic)	0.9290 h^{-1}	0.9288 h^{-1}	0.9288 h^{-1}	0.8739 h^{-1}
μ_{max} (anaerobic)	0.2309 h^{-1}	0.2309 h^{-1}	0.2309 h^{-1}	0.2117 h^{-1}

Taken from Erdrich *et al.* (2015).

Overall flux organization of *E.coli* metabolic network

a, Flux distribution from FBA for optimized biomass production on succinate (black) and glutamate (red) substrates.

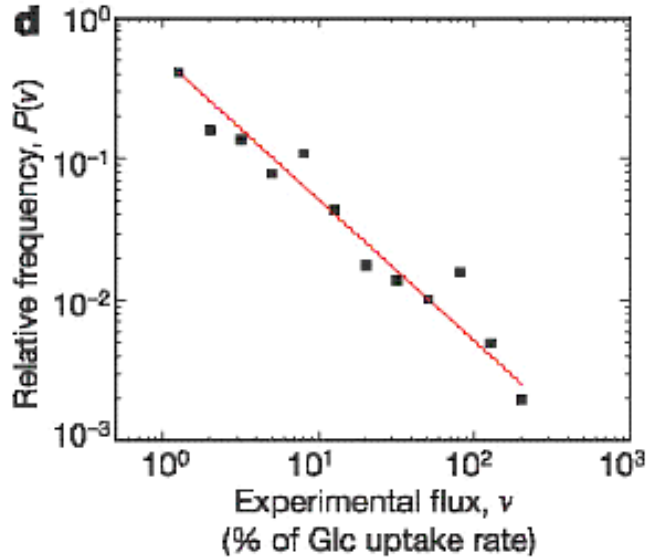
Solid line : power-law fit



d, Experimentally determined fluxes for reactions of the central metabolism of *E. coli.*

Clear **power-law** behaviour.
Best fit with $P(v) \propto v^{-\alpha}$ with $\alpha = 1$.

Both computed and experimental flux distribution show wide spectrum of fluxes.



Almaar et al., Nature 427, 839 (2004)

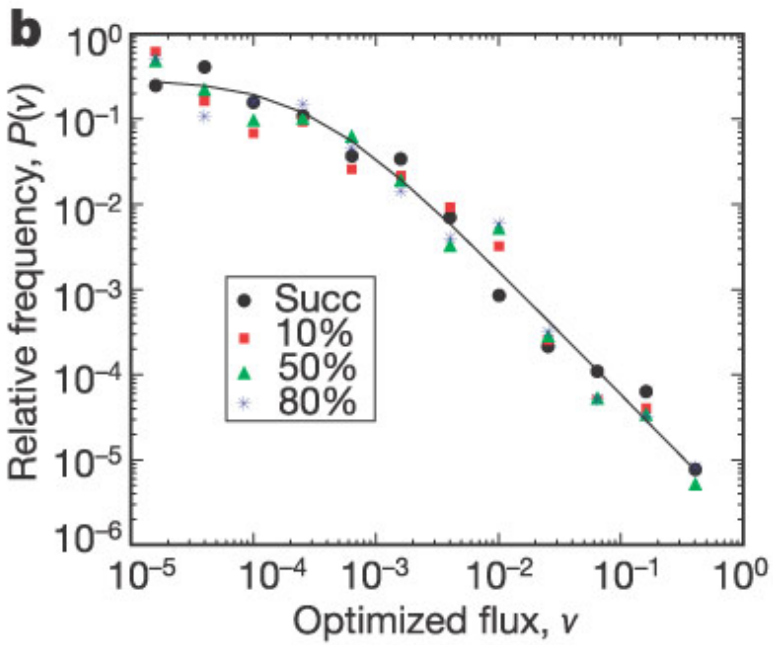
Response to different environmental conditions

Is the flux distribution independent of environmental conditions?

Black: Flux distribution for optimized biomass on pure succinate substrate.

Red / **green** / **blue** :
Flux distributions when an additional 10%, 50%, or 80% of randomly chosen subsets of the 96 input channels (substrates) are added to succinate.

The flux distribution was averaged over 5,000 independent random choices of uptake metabolites.



→ **Yes, the flux distribution is independent of the external conditions.**

Almaar et al., Nature 427, 839 (2004)

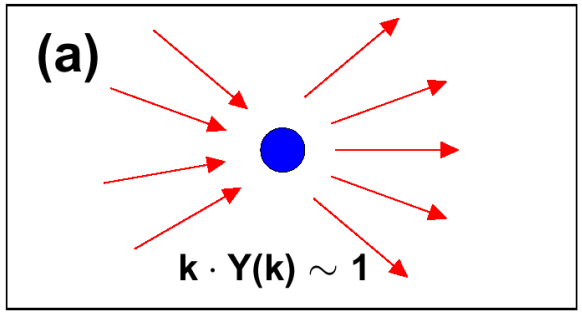
Use scaling behavior to determine local connectivity

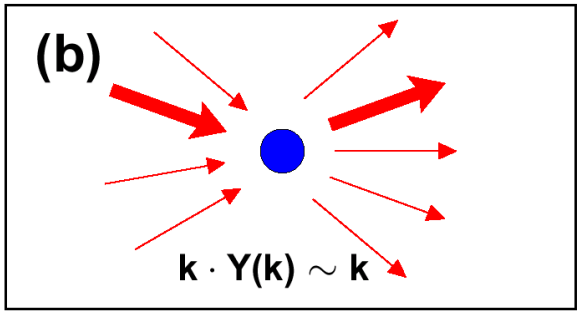
The observed flux distribution is compatible with two different potential local flux structures:

(a) a **homogenous local organization** would imply that all reactions producing (consuming) a given metabolite have **comparable fluxes**

(b) a more delocalized „**high-flux backbone** (HFB)“ is expected if the local flux organisation is heterogenous such that each metabolite has a dominant source (consuming) reaction.

$$Y(k,i) = \sum_{j=1}^{k} \left[\frac{\hat{v}_{ij}}{\sum_{l=1}^{k} \hat{v}_{il}} \right]^2$$





$$k \times \left(k \left(\frac{v}{k \cdot v} \right)^2 \right) = 1$$

All fluxes v_{ij} are the same, say v.

$$k \times \left(\left(\frac{v_{max}}{v_{max}} \right)^2 \right) = k$$

One flux dominates -> replace inner sum by this flux v_{max}.
Also in outer sum, only one j matters.

Almaar et al., Nature 427, 839 (2004)

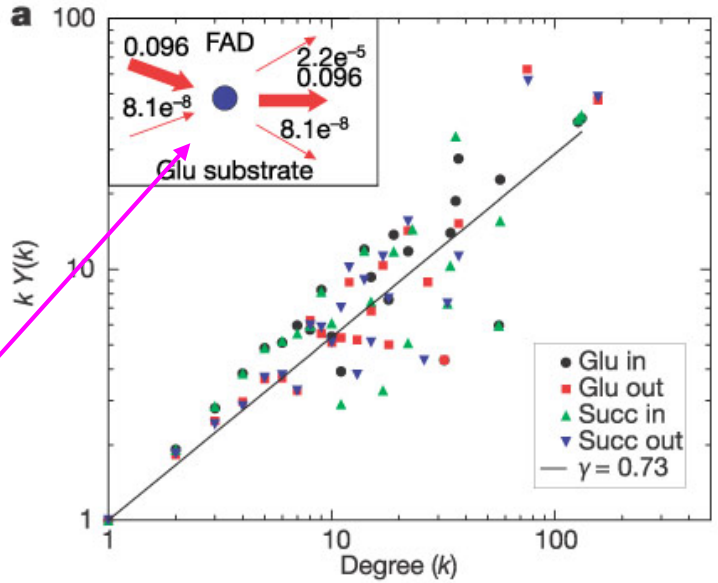
Characterizing the local inhomogeneity of the flux net

FBA-computed $kY(k)$ as a function of k, averaged over all metabolites shows linear dependence $k \times Y(k) \propto k^{0.73}$ with slope 0.73.
This is true for incoming and outgoing reactions.

→ an **intermediate behavior** is found between the two extreme cases discussed before.

→ the large-scale inhomogeneity observed in the overall flux distribution is also valid at the level of the individual metabolites.

The more reactions consume (produce) a given metabolite, the more likely a single reaction carries most of the flux, see inset (FAD).



Inset shows non-zero mass flows producing (consuming) FAD on a glutamate-rich substrate.

Almaar et al., Nature 427, 839 (2004)

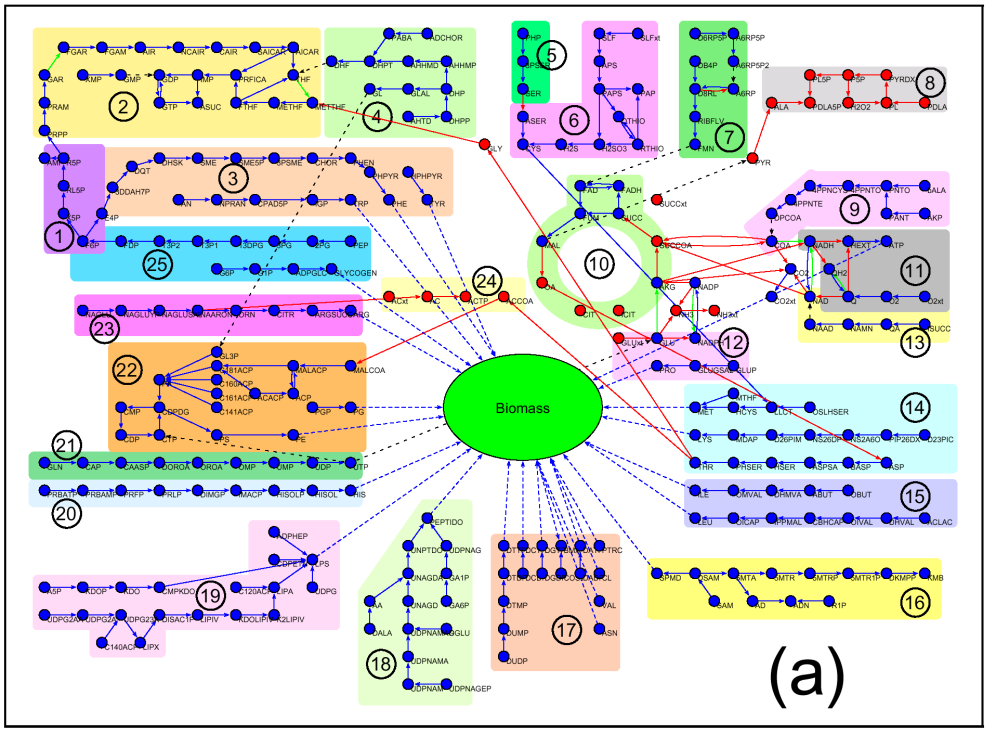
Clean up metabolic network

Use simple algorithm that removes for each metabolite systematically all reactions but the one providing the largest incoming (outgoing) flux distribution.

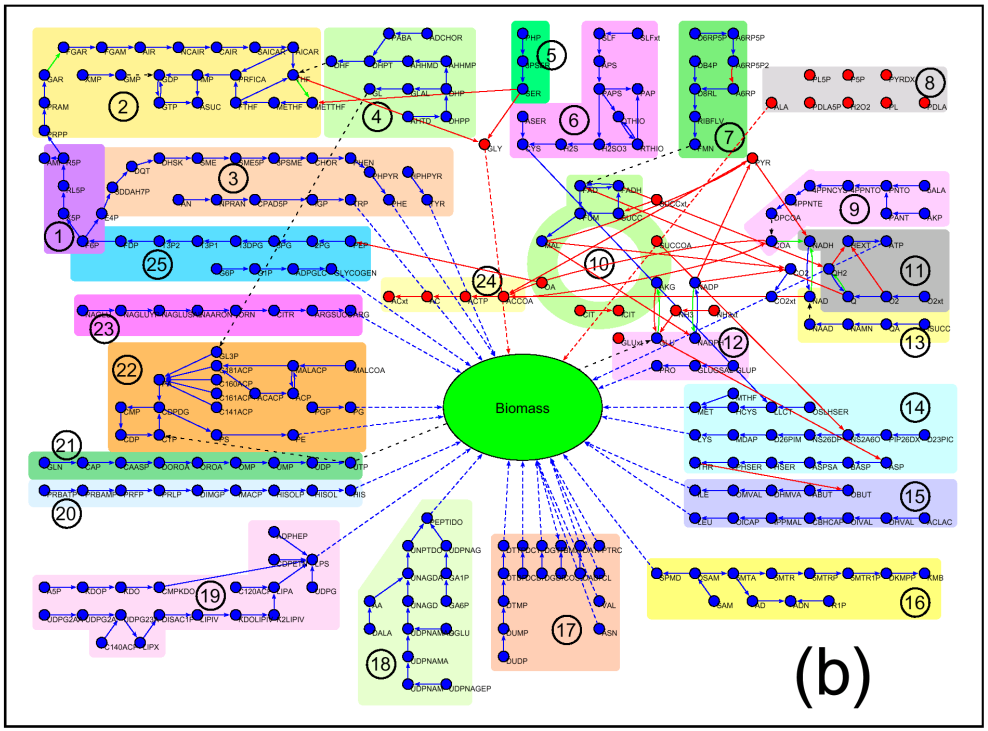
This algorithm uncovers the „**high-flux-backbone**“ of the metabolism.

Almaar et al., Nature 427, 839 (2004)

High-flux backbone of *E.coli* metabolic network



glutamate rich medium



succinate rich medium

Directed links: Metabolites A and B are connected with an arc from A to B if the reaction with maximal flux consuming A is the reaction with maximal flux producing B.
Shown are all metabolites that have at least one neighbour after completing this procedure.

Background colours : known biochemical pathways.

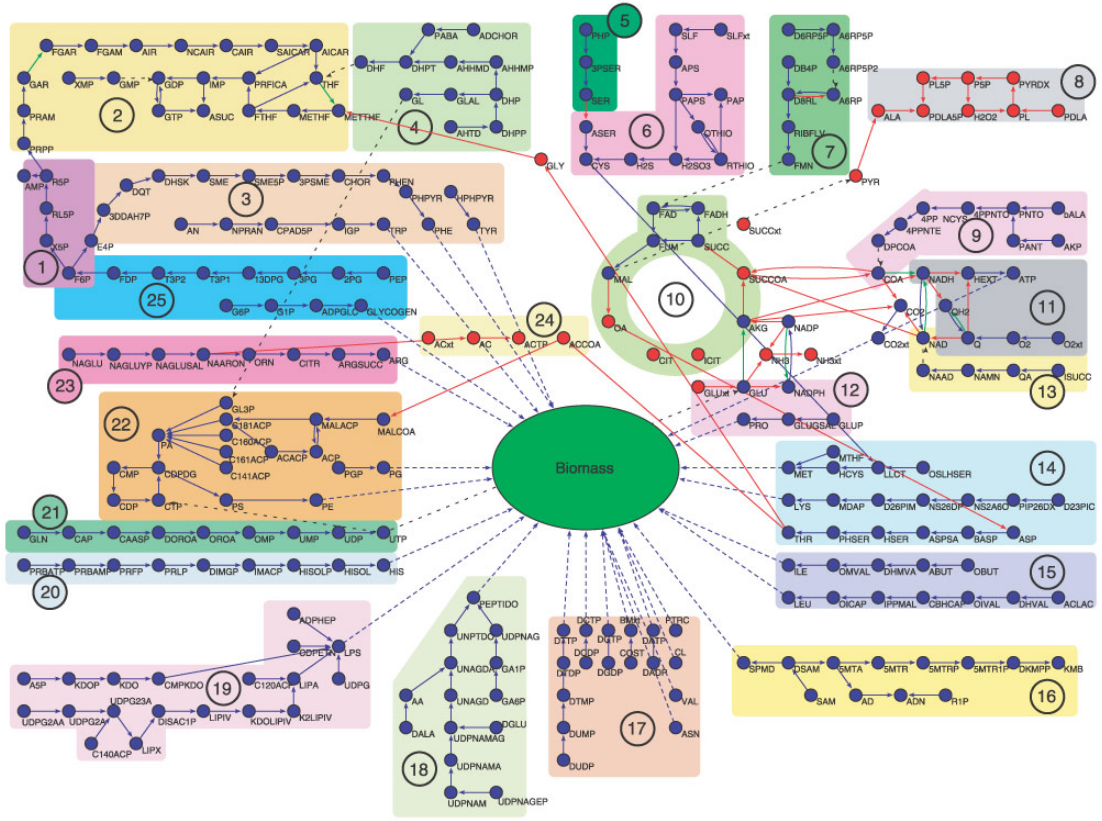
Almaar et al., Nature 427, 839 (2004)

FBA-optimized high-flux backbone on glutamate-rich medium

Blue colored **Metabolites** (vertices) have at least one neighbour in common in glutamate- and succinate-rich substrates.

Red colored nodes have no common neighbors („rewiring“)

Reactions (lines) are coloured
blue if they are identical in glutamate- and succinate-rich substrates,
green if a different reaction connects the same neighbour pair, and
red if this is a new neighbour pair („rewiring“).



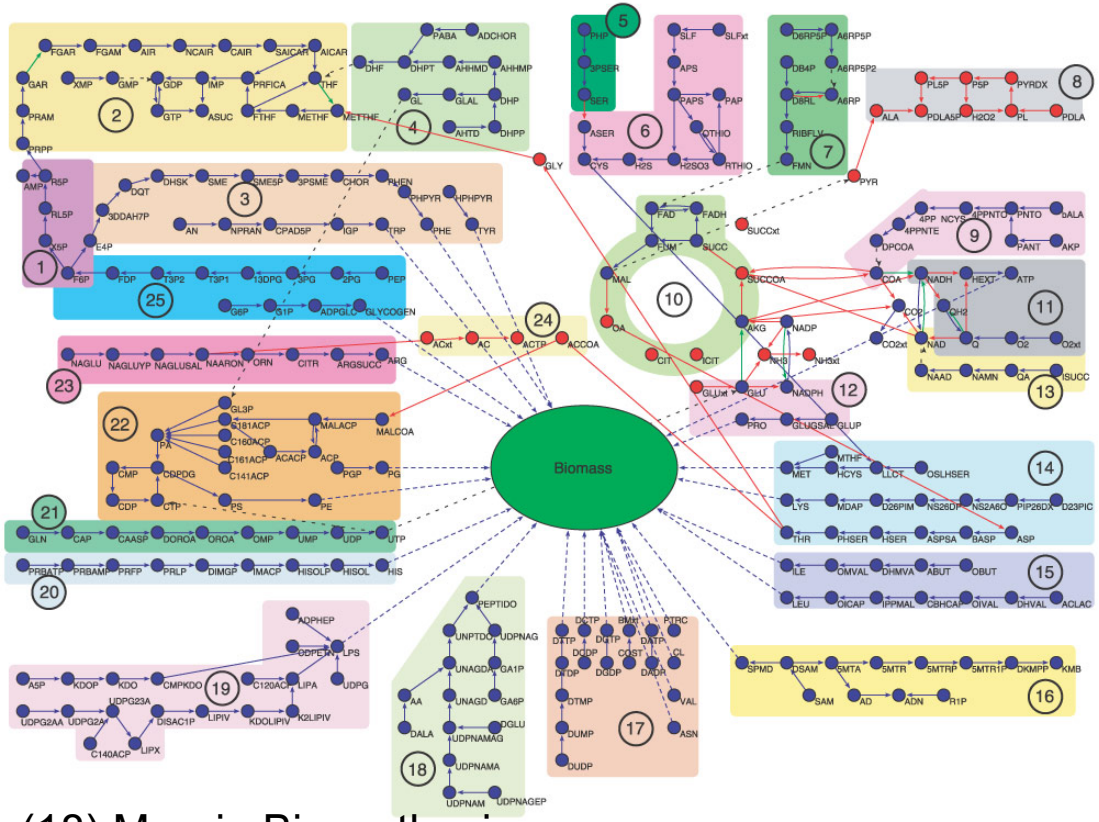
Black dotted lines indicate where the disconnected pathways, e.g., folate biosynthesis (4), would connect to the cluster through a link that is not part of the HFB.

Thus, the red nodes and links highlight the predicted changes in the HFB when shifting *E. coli* from glutamate- to succinate-rich media.

Dashed lines indicate links to the biomass growth reaction.

Almaar et al., Nature 427, 839 (2004)

FBA-optimized high-flux backbone on glutamate-rich medium



(1) Pentose Phospate
(2) Purine Biosynthesis
(3) Aromatic Amino Acids
(4) Folate Biosynthesis
(5) Serine Biosynthesis
(6) Cysteine Biosynthesis
(7) Riboflavin Biosynthesis
(8) Vitamin B6 Biosynthesis
(9) Coenzyme A Biosynthesis
(10) TCA Cycle
(11) Respiration
(12) Glutamate Biosynthesis
(13) NAD Biosynthesis
(14) Threonine, Lysine and Methionine Biosynthesis
(15) Branched Chain Amino Acid Biosynthesis
(16) Spermidine Biosynthesis
(17) Salvage Pathways
(18) Murein Biosynthesis
(19) Cell Envelope Biosynthesis
(20) Histidine Biosynthesis
(21) Pyrimidine Biosynthesis
(22) Membrane Lipid Biosynthesis
(23) Arginine Biosynthesis
(24) Pyruvate Metabolism
(25) Glycolysis

Almaar et al., Nature 427, 839 (2004)

Interpretation

Only a few pathways appear disconnected.

This indicates that although these pathways are part of the HFB, their end product is only the second-most important source for another HFB metabolite.

Groups of individual **HFB reactions largely overlap**
with traditional biochemical partitioning of cellular metabolism ☺

Almaar et al., Nature 427, 839 (2004)

How sensitive is the HFB to changes in the environment?

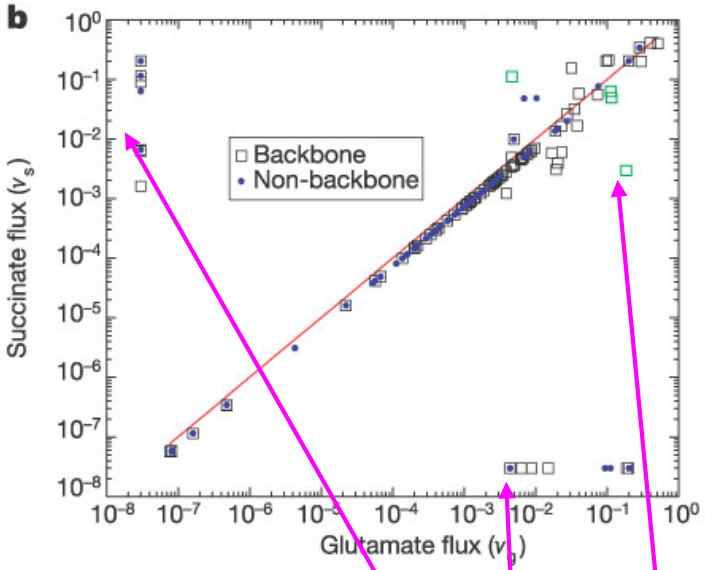
Fluxes of individual reactions on glutamate-rich and succinate-rich medium.

Black squares: reactions belonging to the HFB,
blue dots : remaining reactions

Green squares : reactions in which the direction of the flux is reversed.

Reactions with **negligible** flux changes follow the diagonal (solid line).

Some reactions are turned off in only one of the conditions (shown close to the coordinate axes).



Only reactions in the high-flux territory undergo noticeable differences!

Type I: reactions turned on in one conditions and off in the other.

Type II: reactions remain active but show an orders-in-magnitude shift in flux under the two different growth conditions.

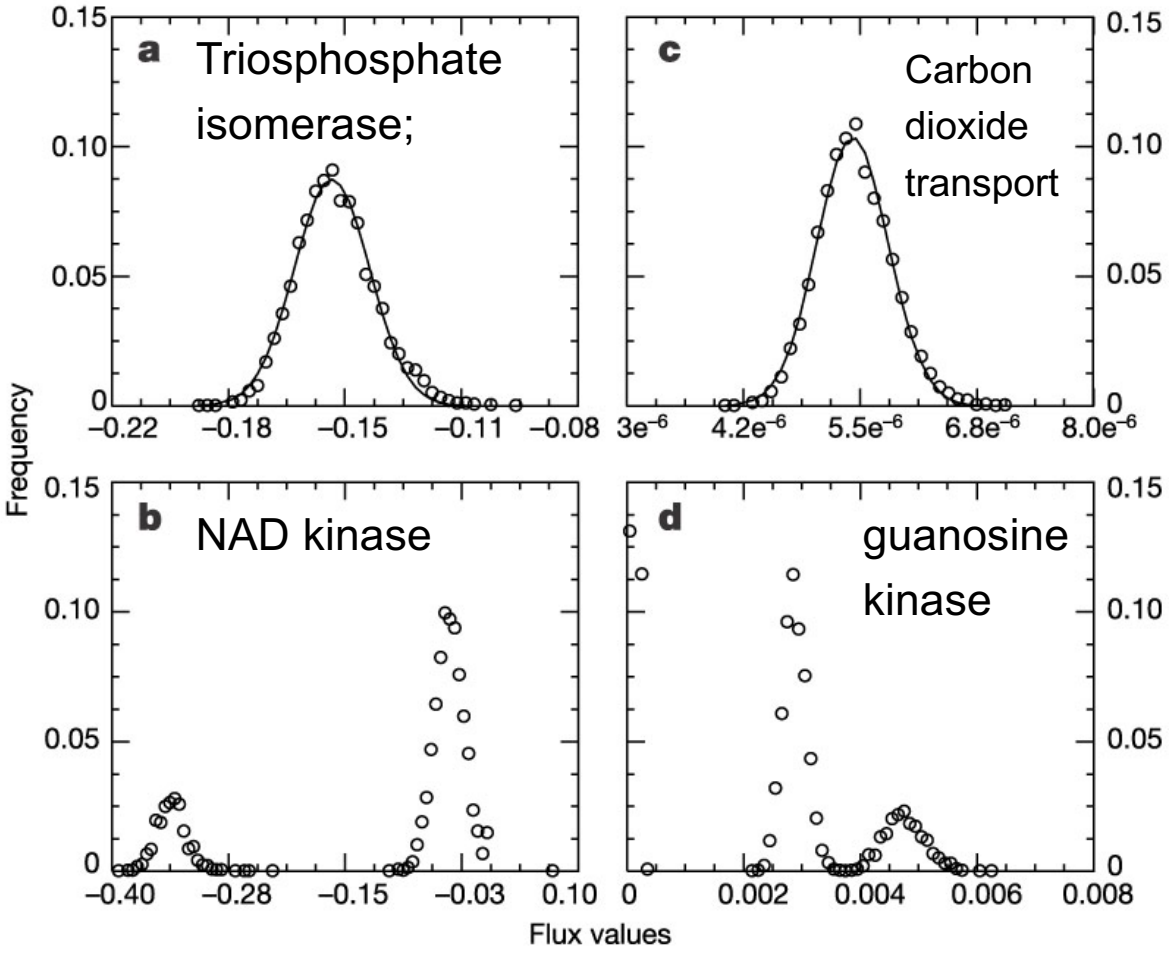
Almaar et al., Nature 427, 839 (2004)

Flux distributions for individual reactions

Shown is the flux distribution for 4 selected *E. coli* reactions on a 50% random medium.

Reactions with small fluxes have **unimodal/gaussian distributions** (a and c).
Shifts in growth-conditions only lead to small changes of their flux values.

Off-diagonal reactions have **multimodal distributions** (b and d), showing several discrete flux values under diverse conditions.



Almaar et al., Nature 427, 839 (2004)

Summary

Metabolic network use is **highly uneven** (power-law distribution) at the global level and at the level of the individual metabolites.

Whereas most metabolic reactions have low fluxes, the overall activity of the metabolism is dominated by several reactions with very high fluxes.

E. coli responds to changes in growth conditions by reorganizing the rates of selected fluxes predominantly within this high-flux backbone.

Apart from minor changes, the use of the other pathways remains unaltered.