

# Escherichia coli Core Metabolism Model in LIM

**Karline Soetaert**

Royal Netherlands Institute of Sea Research  
Yerseke  
The Netherlands

---

## Abstract

R package **LIM** (Soetaert and van Oevelen 2009a) is designed for reading and solving linear inverse models (LIM).

A package vignette deals with the structure of the LIM input files and how to solve the problems (Soetaert and van Oevelen 2009b).

To open it, type:

```
vignette("LIM")
```

Here it is exemplified on a (relatively small) problem from systems biology, the core metabolism of E. coli (Edwards, Covert, and Palsson 2002) as from the following website:

[http://gcrp.ucsd.edu/Downloads/Flux\\_Balance\\_Analysis](http://gcrp.ucsd.edu/Downloads/Flux_Balance_Analysis)

The original input file can be found in the package subdirectory `/examples/Reactions/E_coli.lim`

If you use this package, please cite as: (van Oevelen, van den Meersche, Meysman, Soetaert, Middelburg, and Vezina 2009).

*Keywords:* Linear inverse models, flux balance analysis, linear programming, E coli, R.

---

## 1. the E. coli input file

The input file consists of several sections (see package vignette).

- The header of the file (ends at first line with `###`) is ignored
- The metabolic reactions
- A function to maximise
- The bounds on the reactions (inequalities)
- A measurement equation
- The name of the components
- The names of the externals

Everything following a "!" is ignored.

E.coli input file

#####

## REACTIONS

!gene:	Reaction	!	enzyme
GLK1:	GLC + ATP -> G6P + ADP	!	Glucokinase
PGI1:	G6P <-> F6P	!	Phosphoglucose isomerase
PFKA:	F6P + ATP -> FDP + ADP	!	Phosphofructokinase
FBP:	FDP -> F6P + PI	!	Fructose-1,6-bisphosphatase
FBA:	FDP <-> T3P1 + T3P2	!	Fructose-1,6-bisphosphatase a
TPIA:	T3P2 <-> T3P1	!	Triosphosphate Isomerase
GAPA:	T3P1 + PI + NAD <-> NADH + 13PDG	!	Glyceraldehyde-3-phosphate de
PGK:	13PDG + ADP <-> 3PG + ATP	!	Phosphoglycerate kinase
GPMA:	3PG <-> 2PG	!	Phosphoglycerate mutase 1
ENO:	2PG <-> PEP	!	Enolase
PPSA:	PYR + ATP -> PEP + AMP + PI	!	Phosphoenolpyruvate synthase
PYKA:	PEP + ADP -> PYR + ATP	!	Pyruvate Kinase II
!	PYKF: PEP + ADP -> PYR + ATP	!	Pyruvate Kinase I
ACEE:	PYR + COA + NAD -> NADH + CO2 + ACCOA	!	Pyruvate dehydrogenase
!Pentose Phosphate Pathway			
ZWF:	G6P + NADP <-> D6PGL + NADPH	!	Glucose 6-phosphate-1-dehydro
PGL:	D6PGL -> D6PGC	!	6-Phosphogluconolactonase
GND:	D6PGC + NADP -> NADPH + CO2 + RL5P	!	6-Phosphogluconate dehydrogen
RPIA:	RL5P <-> R5P	!	Ribose-5-phosphate isomerase
RPE:	RL5P <-> X5P	!	Ribulose phosphate 3-epimeras
TKTA1:	R5P + X5P <-> T3P1 + S7P	!	Transketolase I
!	TKTB1: R5P + X5P <-> T3P1 + S7P	!	Transketolase II
TKTA2:	X5P + E4P <-> F6P + T3P1	!	Transketolase I
!	TKTB2: X5P + E4P <-> F6P + T3P1	!	Transketolase II
TALA:	T3P1 + S7P <-> E4P + F6P	!	Transaldolase A
!The Tricarboxylic Acid Cycle			
GLTA:	ACCOA + OA -> COA + CIT	!	Citrate synthase
ACNA:	CIT <-> ICIT	!	Aconitase A
ICDA:	ICIT + NADP <-> CO2 + NADPH + AKG	!	Isocitrate dehydrogenase
SUCA:	AKG + NAD + COA -> CO2 + NADH + SUCCOA	!	2-Ketoglutarate dehydrogenase
SUCC1:	SUCCOA + ADP + PI <-> ATP + COA + SUCC	!	Succinyl-CoA synthetase
SDHA1:	SUCC + FAD -> FADH + FUM	!	Succinate dehydrogenase
FRDA:	FUM + FADH -> SUCC + FAD	!	Fumarate reductase
FUMA:	FUM <-> MAL	!	Fumarase A

MDH: MAL + NAD <-> NADH + OA ! Malate dehydrogenase

!Pyruvate Metabolism

DLD1: PYR + NADH <-> NAD + LAC ! D-Lactate dehydrogenase 1

ADHE2: ACCOA +2\*NADH <-> ETH +2\*NAD + COA ! Acetaldehyde dehydrogenase

PFLA: PYR + COA -> ACCOA + FOR ! Pyruvate formate lyase 1

PTA: ACCOA + PI <-> ACTP + COA ! Phosphotransacetylase

ACKA: ACTP + ADP <-> ATP + AC ! Acetate kinase A

ACS: ATP + AC + COA -> AMP + PPI + ACCOA ! Acetyl-CoA synthetase

!Anaplerotic Reactions

PCKA: OA + ATP -> PEP + CO2 + ADP ! Phosphoenolpyruvate carboxykinase

PPC: PEP + CO2 -> OA + PI ! Phosphoenolpyruvate carboxylase

MAEB: MAL + NADP -> CO2 + NADPH + PYR ! Malic enzyme (NADP)

SFCA: MAL + NAD -> CO2 + NADH + PYR ! Malic enzyme (NAD)

ACEA: ICIT -> GLX + SUCC ! Isocitrate lyase

ACEB: ACCOA + GLX -> COA + MAL ! Malate synthase A

PPA: PPI -> 2\*PI ! Inorganic pyrophosphatase

GLPK: GL + ATP -> GL3P + ADP ! Glycerol kinase

GPSA1: GL3P + NADP <-> T3P2 + NADPH ! Glycerol-3-phosphate-dehydrogenase-

RBSK: RIB + ATP -> R5P + ADP ! Ribokinase

!Respiration Note: the P/O ratio is set to 1.5 currently

NUOA: NADH + Q -> NAD + QH2 +3.5\*HEXT ! NADH dehydrogenase I

FDOH: FOR + Q -> QH2 + CO2 +2\*HEXT ! Formate dehydrogenase-0

GLPD: GL3P + Q -> T3P2 + QH2 ! Glycerol-3-phosphate dehydrogenase 0

CYOA: QH2 +0.5\*O2 -> Q + 2.5\*HEXT ! Cytochrome oxidase bo3

SDHA2: FADH + Q <-> FAD + QH2 ! Succinate dehydrogenase complex

PNT1A: NADPH + NAD -> NADP + NADH ! Pyridine nucleotide transhydrogenase

PNT2A: NADP + NADH +2\*HEXT -> NADPH + NAD ! Pyridine nucleotide transhydrogenase

ATPA: ATP <-> ADP + PI +4\*HEXT ! FOF1-ATPase

!Membrane Transport

GLCUP: GLCxt + HEXT -> GLC ! Glucose/galactose transporter

GLCPTS:GLCxt + PEP -> G6P + PYR ! Glucose

GLUP: GLxt <-> GL ! Glycerol

RIBUP: RIBxt + ATP <-> RIB + ADP + PI ! Ribose

ACUP: ACxt + HEXT <-> AC ! Acetate transport

LACUP: LACxt + HEXT <-> LAC ! L-Lactate

FORUP: FORxt <-> FOR ! Formate transport

ETHUP: ETHxt + HEXT <-> ETH ! Ethanol transport

SUCCUP:SUCCxt + HEXT <-> SUCC ! Succinate transport

PYRUP: PYRxt + HEXT <-> PYR ! Pyruvate transport

```

PIUP: PIxt <-> PI           !      Phosphate transport
O2TX: O2xt <-> O2          !      Oxygen transport
CO2TX: CO2xt <-> CO2       !      Carbon dioxide transport

ATPM: ATP -> ADP + PI       !      ATP drain flux for constant m
ADK:  ATP + AMP-> 2*ADP     !      ADK

Growth:                                &
41.25*ATP +3.54*NAD +18.22*NADPH +0.2*G6P      &
+0.07*F6P +0.89*R5P +0.36*E4P +0.12*T3P1      &
+1.49*3PG +0.51*PEP +2.83*PYR +3.74*ACCOA +1.78*OA +1.07*AKG  &
-> 3.74*COA +41.25* ADP +41.25* PI            &
+3.54* NADH +18.22* NADP +1.00* Biomass
### END REACTION

## MAXIMISE
maxgrowth: Growth
## END MAXIMISE

### INEQUALITIES
! Carbon sources...
O2TX = [0,20]    ! Oxygen input
GLCUP = [0,10]   ! glucose input
GLUP  = [-1000,0] ! glycerol
RIBUP = [-1000,0] ! Ribose uptake - strange!
SUCCUP= [-1000,0] ! succinate
ACUP  = [-1000,0] ! acetate
LACUP = [-1000,0] ! lactate
PYRUP = [-1000,0] ! pyruvate
! Carbon byproducts
FORup = [-1000,0] ! formate
ETHup = [-1000,0] ! ethanol
CO2TX = [-1000,0] ! CO2
! phosphate
PIUP = [-1000,1000]

SDHA1 <100
FRDA <100
FORup+ LACUP=[-10,-10]
### END INEQUALITIES

### EQUATIONS
ATPM = 5.87 ! Non-growth associated ATP drain flux for constant maintenance require

```

### END EQUATIONS

### COMPONENTS

GLC ! a-D-Glucose  
G6P ! Glucose 6-phosphate  
F6P ! Fructose 6-phosphate  
FDP ! Fructose 1,6-diphosphate  
T3P2 ! /DHAP Dihydroxyacetone phosphate  
T3P1 ! Glyceraldehyde 3-phosphate  
13PDG ! 1,3-bis-Phosphoglycerate  
3PG ! 3-Phosphoglycerate  
2PG ! 2-Phosphoglycerate  
PEP ! Phosphoenolpyruvate  
PYR ! Pyruvate  
ACCOA ! Acetyl-CoA  
CIT ! Citrate  
! ACO ! cis-Aconitate  
ICIT ! Isocitrate  
AKG ! a-Ketoglutarate  
SUCCOA ! Succinate CoA  
SUCC ! Succinate  
FUM ! Fumarate  
MAL ! Malate  
OA ! Oxaloacetate  
! ACAL ! Acetaldehyde  
ACTP ! Acetyl-phosphate  
ETH ! Ethanol  
AC ! Acetate  
LAC ! D-Lactate  
FOR ! Formate  
D6PGL ! D-6-Phosphate-glucono-delta-lactone  
D6PGC ! D-6-Phosphate-gluconate  
RL5P ! Ribulose 5-phosphate  
X5P ! Xylulose-5-phosphate  
R5P ! Ribose 5-phosphate  
S7P ! sedo-Heptulose  
E4P ! Erythrose 4-phosphate  
RIB ! Ribose  
GLX ! Glyoxylate  
NAD ! Nicotinamide adenine dinucleotide  
NADH ! Nicotinamide adenine dinucleotide (reduced)  
NADP ! Dihyronicotinamide adenine dinucleotide phosphate  
NADPH ! Dihyronicotinamide adenine dinucleotide phosphate (reduced)

```
HEXT    ! External Hydrogen Ion (Proton)
Q       ! Ubiquinone

FAD     ! Flavin adenine dinucleotide
FADH    ! Flavin adenine dinucleotide (reduced)
AMP     ! Adenosine monophosphate
ADP     ! Adenosine diphosphate
ATP     ! Adenosine triphosphate
GL3P    ! Glycerol 3-phosphate
CO2     ! Carbon dioxide
PI      ! Inorganic Phosphate
PPI     ! Pyrophosphate

O2      ! Oxygen
COA
GL
QH2     !
### END COMPONENTS

### EXTERNALS
Biomass
GLCxt
GLxt
RIBxt
ACxt
LACxt
FORxt
ETHxt
SUCCxt
PYRxt
PIxt
O2xt
CO2xt
### END EXTERNALS
```

## 2. Reading the E.coli input file

Assuming that the input file is called "E\_coli.lim" and the working directory has been set, it can be read as follows:

```
require(LIM)
LIMEcoli <- Setup("E_coli.lim")
```

This creates a list of type `lim`, that contains all information necessary to solve the problem

## 3. The parsimonious and optimized solution, ranges

Once the input file has been read, we can generate the "simplest" solution, i.e. the one where  $\sum(x^2)$  is minimal, where  $x$  are the unknown reaction rates. This is called the "parsimonious solution". It is common to calculate this in foodweb ecology (where it is unclear which characteristic of a foodweb is optimized); it may be less relevant from a system's biology perspective.

Function `Ldei` estimates the parsimonious solution

```
> pars <- Ldei(LIMEcoli)
```

It makes more sense to optimize the growth. That growth has to be maximised was inputted in the file (by the `## maximize` statement).

The optimal value is found by linear programming, using function `Linp`:

```
> LP <- Linp(LIMEcoli)
```

It is also simple to estimate the ranges of all unknown reaction rates:

```
> xr <- Xranges(LIMEcoli)
```

Now for every reaction rate, the "simplest", "optimal", "minimal" and "maximal" value has been estimated:

```
> data.frame(simplest = pars$X, optimal = LP$X, xr)
```

	simplest	optimal	min	max
GLK1	1.2508063751	0.000000	0.0000000	10.000000
PGI1	4.1688322362	807.532745	-15.8333333	807.532745
PFKA	4.6630482111	781.590686	0.8333333	2229.130000
FBP	0.4831134621	0.000000	0.0000000	1604.130000
FBA	4.1799347490	781.590686	0.8333333	781.590686
TPIA	4.1675392962	781.590686	0.8333333	781.590686
GAPA	8.3531445746	1541.434199	5.0000000	1541.434199

PGK	8.3531445746	1541.434199	5.0000000	1541.434199
GPMA	8.3538152036	1492.089090	5.0000000	1492.089090
ENO	8.3538152036	1492.089090	5.0000000	1492.089090
PPSA	-0.0012096961	0.000000	0.0000000	1604.130000
PYKA	2.5684657495	466.657964	0.0000000	2136.630000
ACEE	0.0007587511	1149.295284	0.0000000	1158.949190
ZWF	-0.0035756501	0.000000	0.0000000	75.000000
PGL	-0.0035756501	0.000000	0.0000000	75.000000
GND	-0.0035756501	0.000000	0.0000000	75.000000
RPIA	-0.0146466569	23.623833	0.0000000	28.202015
RPE	0.0110710067	-23.623833	-23.6238328	50.000000
TKTA1	0.0054544878	-5.850762	-5.8507623	25.000000
TKTA2	0.0056165190	-17.773070	-17.7730705	25.000000
TALA	0.0054544878	-5.850762	-5.8507623	25.000000
GLTA	1.7692398473	35.435749	0.0000000	40.847149
ACNA	1.7692398473	35.435749	0.0000000	40.847149
ICDA	-0.0063769772	35.435749	0.0000000	40.847149
SUCA	-0.0058953845	0.000000	0.0000000	30.000000
SUCC1	-0.0058953845	0.000000	0.0000000	30.000000
SDHA1	1.7656919880	0.000000	0.0000000	100.000000
FRDA	-0.0018735606	100.000000	0.0000000	100.000000
FUMA	1.7675655486	-100.000000	-100.0000000	8.333333
MDH	-1.1015704154	-100.000000	-1168.3150000	16.666667
DLD1	3.0297277502	0.000000	0.0000000	10.000000
ADHE2	3.5846082271	1000.000000	0.0000000	1000.000000
PFLA	7.1027439722	10.000000	0.0000000	150.000000
PTA	1.7921570508	0.000000	0.0000000	1660.380000
ACKA	1.7921570508	0.000000	0.0000000	1660.380000
ACS	1.8164359024	0.000000	0.0000000	1604.130000
PCKA	0.0008317580	0.000000	0.0000000	1604.130000
PPC	2.8708408666	194.384939	0.0000000	1704.130000
MAEB	1.5433419402	0.000000	0.0000000	1068.315000
SFCA	3.1014108483	0.000000	0.0000000	1068.315000
ACEA	1.7756168245	0.000000	0.0000000	30.000000
ACEB	1.7756168245	0.000000	0.0000000	30.000000
PPA	1.8164359024	0.000000	0.0000000	1604.130000
GLPK	-0.0123954528	0.000000	0.0000000	0.000000
GPSA1	-1.6917400737	0.000000	-140.0000000	0.000000
RBSK	0.0197005675	0.000000	0.0000000	0.000000
NUOA	-0.0064149697	140.000000	0.0000000	140.000000
FDOH	0.0068803568	0.000000	0.0000000	140.000000
GLPD	1.6793446208	0.000000	0.0000000	140.000000
CYOA	3.4473755565	40.000000	0.0000000	40.000000



SDHA2	1.7675655486	-100.000000	-100.000000	8.333333
PNT1A	1.3888541919	0.000000	0.000000	3208.260000
PNT2A	1.5425800246	567.965512	0.000000	3208.260000
ATPA	-2.7157063757	-145.466329	-460.000000	1144.130000
GLCUP	1.2508063751	0.000000	0.000000	10.000000
GLCPTS	2.9143601936	814.156250	0.000000	814.156250
GLUP	-0.0123954528	0.000000	0.000000	0.000000
RIBUP	0.0197005675	0.000000	0.000000	0.000000
ACUP	0.0242788516	0.000000	-75.000000	0.000000
LACUP	-3.0297277502	0.000000	-10.000000	0.000000
FORUP	-7.0958636154	-10.000000	-10.000000	0.000000
ETHUP	-3.5846082271	-1000.000000	-1000.000000	0.000000
SUCCUP	-0.0021558913	-100.000000	-130.000000	0.000000
PYRUP	0.0031683006	-27.796342	-150.000000	0.000000
PIUP	-0.0016383153	120.547782	0.000000	120.547782
O2TX	1.7236877783	20.000000	0.000000	20.000000
CO2TX	-1.7665347761	-990.346093	-1000.000000	0.000000
ATPM	5.8700000000	5.870000	5.870000	5.870000
ADK	1.8152262063	0.000000	0.000000	1604.130000
Growth	-0.0004500866	33.117523	0.000000	33.117523

The range solutions can be plotted; as there are many reactions, we plot them in two figures. The "optimal" solution is added as a black dot.

```
> par(mfrow = c(1, 2))
> nr <- LIMEcoli$NUnknowns
> ii <- 1:(nr/2)
> dotchart(LP$X[ii], xlim = range(xr), pch = 16, cex = 0.8)
> segments(xr[ii,1], 1:nr, xr[ii,2], 1:nr)
> ii <- (nr/2+1):nr
> dotchart(LP$X[ii], xlim = range(xr), pch = 16, cex = 0.8)
> segments(xr[ii,1], 1:nr, xr[ii,2], 1:nr)
> mtext(side = 3, cex = 1.5, outer = TRUE, line = -1.5,
+       "E coli Core Metabolism, optimal solution and ranges")
```

## E coli Core Metabolism, optimal solution and ranges

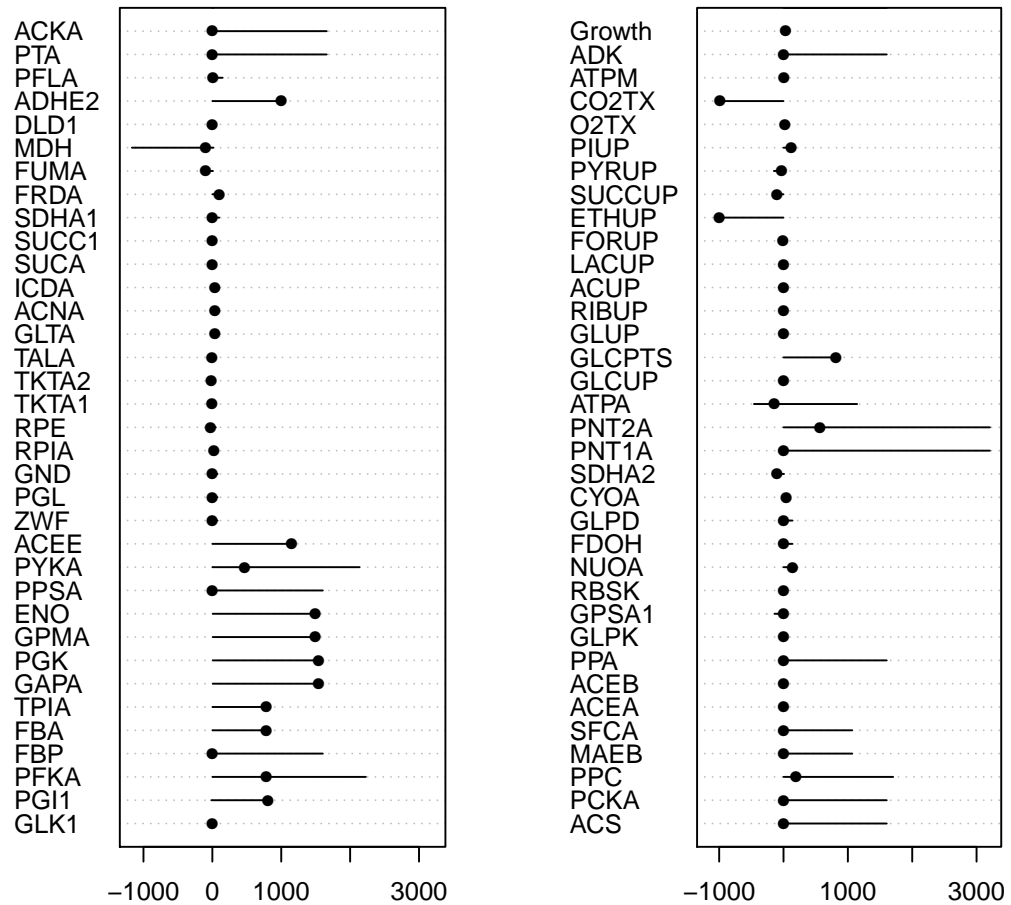


Figure 1: Ranges, and optimal solution of the E.coli central core metabolism - see text for R-code

## 4. Generating multiple plausible solutions

The E.coli model is underdetermined, such that an infinite amount of solutions are likely, given the data. By optimising growth, we selected one "optimal" solution; by estimating the ranges, we calculated the minimal and maximal values of each reaction.

It is also possible to sample the solution space in a random way. Function `xsample` does that; each point it generates is equally valid and equally likely.

We take 500 random samples; it takes a while to do this; `print(system.time())` estimates the time, in seconds.

```
> print(system.time(  
+   xs <- Xsample(LIMEcoli, iter = 500, type = "mirror", test = TRUE) #))  
+ ))
```

```
   user  system elapsed  
  8.55   0.79   9.44
```

```
>
```

With 70 variables, it is not possible to plot all pairwise relationships.

Here we plot them for 12 of them.

```
> pairs(xs[, 1:12], pch = ".", cex = 2, gap = 0, upper.panel = NULL)
```

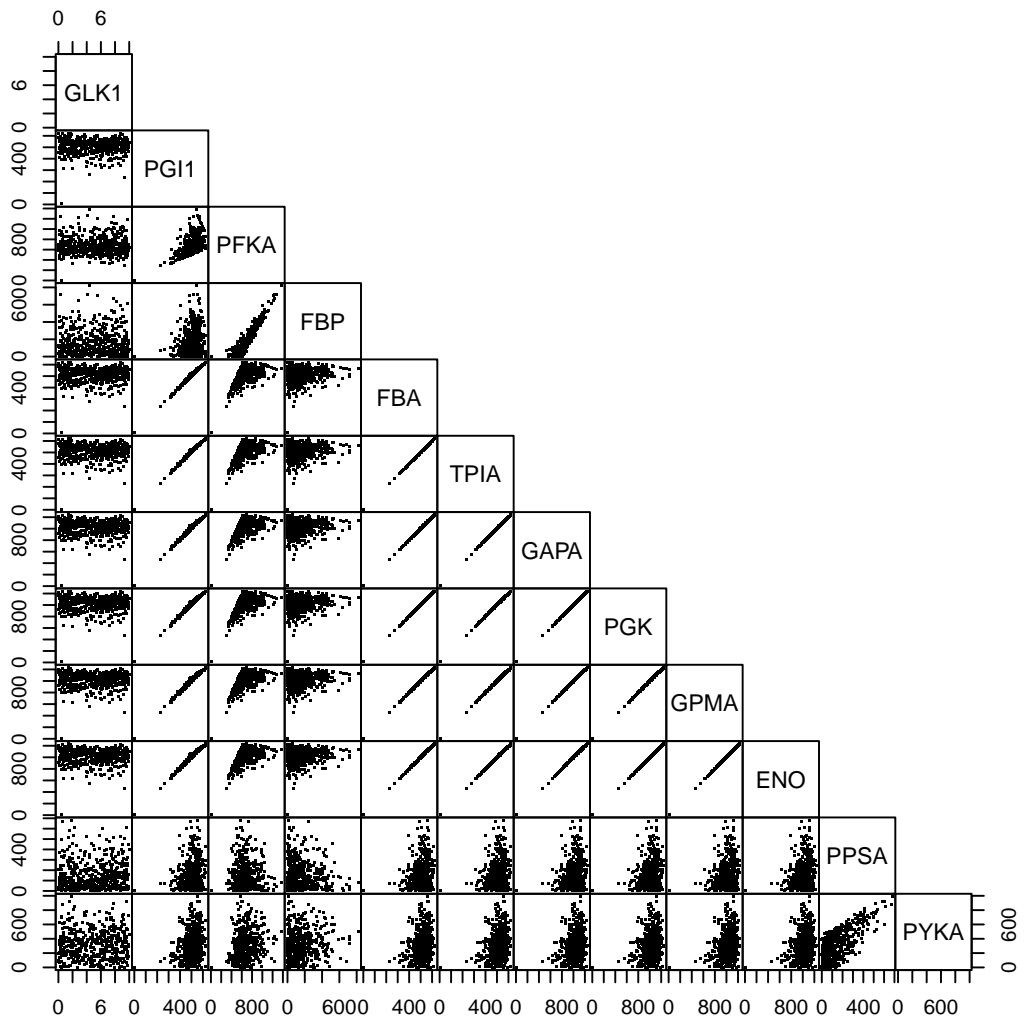


Figure 2: A random sample of plausible solutions of the E.coli central core metabolism - plotted as a pairwise plot for the first 12 reaction rates see text for R-code

## References

- Edwards J, Covert M, Palsson B (2002). “Metabolic Modeling of Microbes: the Flux Balance Approach.” *Environmental Microbiology*, **4(3)**, 133–140.
- Soetaert K, van Oevelen D (2009a). *LIM: Linear Inverse Model examples and solution methods*. R package version 1.4.
- Soetaert K, van Oevelen D (2009b). *Package LIM, implementing linear inverse models in R*. R package vignette.
- van Oevelen D, van den Meersche K, Meysman F, Soetaert K, Middelburg JJ, Vezina AF (2009). “Quantifying Food Web Flows Using Linear Inverse Models.” *Ecosystems*. doi:10.1007/s10021-009-9297-6. URL <http://www.springerlink.com/content/4q6h4011511731m5/fulltext.pdf>.

### Affiliation:

Karline Soetaert  
Royal Netherlands Institute of Sea Research (NIOZ)  
4401 NT Yerseke, Netherlands E-mail: [karline.soetaert@nioz.nl](mailto:karline.soetaert@nioz.nl)  
URL: <http://www.nioz.nl>