Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks: supplemental information

Steven M. Kelk[1,2], Brett G. Olivier[2,3,5], Leen Stougie[2,4], Frank J. Bruggeman[2,3,5,6,*]

July 20, 2012

1. Knowledge Engineering, Maastricht University, P.O. Box 616, 6200 MD, Maastricht, The Netherlands.
2. Life Science, Centre for Mathematics and Computer Science (CWI), Science Park 123, 1098 XG Amsterdam, The Netherlands.
3. Molecular Cell Physiology, VU University, De Boelelaan 1087, 1081 HV, Amsterdam, The Netherlands.
4. Operations Research, VU University, De Boelelaan 1085, 1081 HV, Amsterdam, The Netherlands.
5. Netherlands Institute for Systems Biology, Amsterdam, The Netherlands.
6. Swammerdam Institute for Life Sciences, University of Amsterdam, Amsterdam, The Netherlands.
* To whom correspondence should be addressed: f.j.bruggeman@vu.nl

1 Supplementary Information

1.1 Overall stoichiometries of the five subnetworks of *Escherichia coli* iJR904 growing on glucose

Subnetwork 1

```
- 0.007 ade + 6.92797 adp - 2.3025 amp - 4.64317 atp + 0.0247 datp
+ 0.0254 dgtp + 0.0247 dump - 0.522558 gdp - 0.27841 gmp + 0.775568 gtp
+ 0.524 h + 0.0748 h2o + 0.517 n2dr1p + 0.524 pi - 0.524 r1p + 0.5918 trdox
- 0.5918 trdrd - 0.0528 udp - 0.3511 ump + 0.3792 utp
```

Subnetwork 2

```
- 0.00273 aspL + 0.367362 co2c - 0.3151 dhorS - 0.367362 forc + 2.45148 fum
+ 0.05 glx - 0.05 glyclt - 1.10209 hc + 0.734724 he + 0.00273 iasp
+ 0.3151 orot - 2.81931 q8 - 0.367362 q8c + 2.81931 q8h2 + 0.367362 q8h2c
- 2.45148 succ
```

Subnetwork 3

```
7.1434 adp - 7.1434 atp + 8.06593 dhap - 8.06593 f6p + 8.06593 g3p
+ 7.1434 h - 0.922533 pep + 0.922533 pyr
```

Subnetwork 4

```
0.194309 adp - 0.194309 atp - 0.194309 cdp + 0.168909 ctp + 0.0254 dctp
+ 0.0254 h2o + 0.0254 trdox - 0.0254 trdrd
```

Subnetwork 5

```
- 0.3213 accoa + 0.3213 actACP + 0.3213 co2 + 0.3213 coa - 0.3213 h
- 0.3213 malACP
```

1.2 The subnetworks of iJR904 and iAF1260 growing on glucose

The subnetworks, rays and linealities for all the genome scale reconstructions mentioned in the main article text are available online (`http://code.google.com/p/memesa-tools`).

Subnetwork 1 (24 vertices)					
R_UMPK	R_URIK2	R_DGK1	R_NDPK8	R_NDPK6	R_NDPK5
R_NDPK2	R_NDPK1	R_RNDR4	R_RNDR2	R_RNDR1	R_URIDK2r
R_DURIPP	R_GK1	R_NTD8	R_NTD1	R_DADK	R_NTD6
R_PUNP4	R_PUNP1	R_PUNP3	R_PYNP2r	R_ADNK1	R_ADK1
R_RNTR4	R_PUNP2	R_RNTR2	R_RNTR1	R_GSNK	
Subnetwork 2 (90 vertices)					
R_DHORD5	R_FDH3	R_FDH2	R_DHORD2	R_SUCD4	R_GLYCTO4
R_ASPO4	R_GLYCTO2	R_GLYCTO3	R_FRD3	R_FRD2	R_HYD3
R_HYD2	R_FHL	R_SUCD1i	R_HYD1	R_ASPO5	R_ASPO3
Subnetwork 3 (2 vertices)					
R_PYK	R_PFK	R_DHAPT	R_FBA	R_F6PA	
Subnetwork 4 (2 vertices)					
R_NDPK7	R_NDPK3	R_RNDR3	R_RNTR3		
Subnetwork 5 (2 vertices)					
R_KAS14	R_KAS15	R_ACOATA			

Table 1: The fluxes and subnetworks of iJR904 (with glucose as a sole carbon source and optimal growth rate of one) and the number of unique vertex flux distributions that that occur in each subnetwork (total number of vertices 17280).

Subnetwork 1 (1296 vertices)					
R_RPE	R_ACt2rpp	R_GLYCLTt4pp	R_THRt4pp	R_GLUt4pp	R_EX_for_e
R_RPI	R_PTAr	R_GLUt2rpp	R_GLYCTO2	R_GLYCTO3	R_GLYCTO4
R_GHMT2r	R_PYK	R_PGM	R_PGL	R_PGK	R_PGI
R_NADH18pp	R_CO2tex	R_TPI	R_SERt4pp	R_TKT1	R_TKT2
R_PGCD	R_DHORD2	R_MTHFD	R_MTHFC	R_GAPD	R_DHORD5
R_NADH16pp	R_PSERT	R_THRt2rpp	R_GART	R_NADH17pp	R_GLYCL
R_F6PA	R_MDH	R_CO2tpp	R_PDH	R_SUCOAS	R_H2Otpp
R_PSP_L	R_ACtex	R_ICDHyr	R_H2Otex	R_ACKr	R_G6PDH2r
R_DHAPT	R_PROt2rpp	R_PROt4pp	R_ASPO4	R_EX_ac_e	R_GLYCLTt2rpp
R_ENO	R_ACt4pp	R_Htex	R_SERt2rpp	R_ASPO3	R_PFK
R_ASPO5	R_CS	R_EX_h2o_e	R_EX_co2_e	R_AKGDH	R_FBA
R_GLUDy	R_ACONTb	R_ACONTa	R_FUM	R_GND	R_GARFT
R_FORtex	R_FRD3	R_FRD2	R_SUCDi	R_TALA	R_ATPS4rpp
R_FORtppi	R_EX_h_e				
Subnetwork 2 (324 vertices)					
R_GTHOr	R_PRPPS	R_RNDR2b	R_PPKr	R_RNDR4b	R_NDPK8
R_NDPK7	R_NDPK6	R_GRXR	R_NDPK3	R_NDPK2	R_NDPK1
R_RNDR4	R_RNDR2	R_RNDR3	R_RNDR1	R_RNDR1b	R_FLDR
R_RNDR3b	R_PAPSR2	R_RNTR1c	R_RNTR3c	R_ADPT	R_TRDR
R_PUNP1	R_ADK3	R_ADNK1	R_PAPSR	R_RNTR2c	R_RNTR4c
R_PPM	R_NDPK5				
Subnetwork 3 (2 vertices)					
R_DMPPS	R_IPDPS	R_IPDDI			
Subnetwork 4 (2 vertices)					
R_KAS14	R_KAS15	R_ACOATA			

Table 2: The identity of the fluxes that appear in the subnetworks of iAF1260 (with glucose as a sole carbon source and optimal growth rate of one) and the number of unique vertex flux distributions that that occur in each subnetwork (total number of vertices 1679616).

Mathematical background: polyhedra

Consider the LP representing the FBA as expressed in the Methods section. We represent the set of inequalities here by $\mathbf{Av} \geq \mathbf{b}$.

$$\begin{array}{ll} \textit{maximize} & f(\mathbf{v}) = \mathbf{cv} \\ \textit{subject to} & \mathbf{Av} \geq \mathbf{b} \end{array} \tag{1}$$

Optimal solutions of FBA programs are hardly ever unique. For analyzing the structure of the space of all optimal fluxes, we define some concepts. A *linear combination* of vectors $v_1, v_2, \ldots, v_k$ is defined as any vector $v = \sum_{i=1}^{k} \lambda_i v_i$ for any constants $\lambda_1, \lambda_2, \ldots, \lambda_k$. It is a *positive linear combination* if $\lambda_i \geq 0$, $i = 1, \ldots, k$ and a *convex combination* if $0 \leq \lambda_i \leq 1$, $i = 1, \ldots, k$ and $\sum_{i=1}^{k} \lambda_i = 1$.

The set of optimal solutions of a FBA problem is a convex set, which means that it contains every convex combination of every set of its points. Because it is defined by a finite number of linear equalities and inequalities (the constraints in the LP (1) plus the constraint that the objective value of the LP is equal to the optimal value) it is a polyhedron (or polytope, when bounded, i.e., when none of the fluxes can reach minus or plus infinity) [12]. This description of a polyhedron is called the "outer description."

Any polyhedron has another description, the "inner description," as the Minkowsky sum of three sets. The first is a *lineality space*, which are all the flux vectors that satisfy the steady state constraints and have zero flux on all irreversible and all bounded reactions; i.e., in terms of the compressed description of the feasible set of (1), all flux vectors in $\{\mathbf{v} \mid \mathbf{Av} = \mathbf{0}\}$, the *null-space* of the matrix $\mathbf{A}$. For any flux $\mathbf{v}$ in this space and any optimal flux $\mathbf{v}'$, the flux $\mathbf{v}' + \mu\mathbf{v}$ is also optimal for every value of μ. This space is a linear subspace and is characterized by a, usually non-unique, set of basis vectors, their number being equal to the dimension of this subspace. We call the basis vectors linealities and they can be chosen to correspond to fully matter-preserving reversible cycles in the network.

Projecting the polyhedron onto the orthogonal of the lineality space through any feasible point gives a polyhedron with extreme points and extreme rays. Given this projection, the second set is a *cone*, which is the set of flux vectors that satisfy the steady state constraints and have positive value on reactions bounded from below but not from above, in particular irreversible reactions, and zero flux on reactions with fluxes bounded from above, in particular the fluxes that are prefixed (e.g. the prefixed growth rate). For any flux $\mathbf{v}$ in this cone and any optimal flux $\mathbf{v}'$, the flux $\mathbf{v}' + \nu\mathbf{v}$ is also optimal for every value of $\nu > 0$. A cone is characterized by a set of extreme rays, elements of the cone that can not be expressed as a positive linear combination of any other elements of the cone. Every element of the cone is a positive linear combination of rays. Rays correspond to fully matter-preserving irreversible cycles in the network.

Again given the projection, the third set is a *polytope*, which is the set of all convex combinations of its vertices, which are feasible and optimal flux vectors, i.e. optimal vectors in $\{\mathbf{v} \mid \mathbf{Av} \geq \mathbf{b}\}$, that cannot be written themselves as the convex combination of any other two vectors with this property. Hence, vertices do belong to the optimal solution space. The polytope is completely characterized by the set of vertices in the sense that any element of the polytope can be expressed as a convex combination of its vertices. Any optimal flux vector can then be written as a linear combination of the linealities plus a positive linear combination of the rays of the cone plus a convex combination of the vertices of the polytope. For more details about polyhedra we refer the reader to standard mathematical texts such as [14, 12].

As a final note, linealities and rays could in principle also correspond to paths in the network from some non-nutrient metabolite to some non-biomass metabolite, which are modeled to be present in fixed concentration. Such parts of the network are at best irrelevant in FBA analysis and therefore should be ignored or even deleted. In the networks we analyzed we never encountered linealities or rays of this kind. We emphasize that the elements of the lineality space and the cone do not belong to the optimal solution space; they merely tell the directions in which the optimal solution space is unbounded.

1.3 Analysing genome scale models

1.3.1 Preparing models for characterization

The following procedure was followed that prepared genome scale stoichiometric models for enumeration. All models and software referred to in this document, as well as detailed instructions on how to install and use the enumeration pipeline are available on the MEMESA Tools web site (`http://code.google.com/p/memesa-tools`).

First the reconstruction (e.g. E. coli iJR904 [9] obtained from the BiGG database [11]) was downloaded as a fully compartmentalized model in the SBML dialect compatible with the COBRA Toolkit [6, 10] and then converted into SBML Level 2 with FBA annotation [3] suitable for use in the PySCeS-CBM constraint based modelling framework [8, 7]. Next it was scanned for reactions that have an exact matching stoichiometry (ignoring reversibility and reagent order). For example, in iJR904 one of the nine pairs of redundant reactions involves adenosine (adn) transport via a proton (h) symport:

$$\begin{array}{lccc} \texttt{R\_ADNt2r:} & \mathrm{adn}_e + \mathrm{h}_e & \longleftrightarrow & \mathrm{adn}_c + \mathrm{h}_c \\ \texttt{R\_ADNt2:} & \mathrm{adn}_e + \mathrm{h}_e & \longrightarrow & \mathrm{adn}_c + \mathrm{h}_c \end{array}$$

Note how the two reactions differ only in the one being reversible (`R_ADNt2r`) and the other irreversible (`R_ADNt2`). When such a redundant pair of reactions was found, the irreversible reaction was deleted.

For each model thus processed an uptake flux of a single carbon source was determined such that an optimal biomass production flux of one unit was obtained. In iJR904 this resulted in the maximum uptake rates of glucose (`R_EX_glc_(e)`) and threonine (`R_EX_thr_L_(e)`) being set as -11.64 and -69.80 units respectively. Any artificial infinity constraints (see 1.5) were removed by scanning though all reaction flux bounds and deleting any one found to have an "infinite" value (for iJR904 this is ±999999). The model was then exported and saved to disk as a rational linear program in H-format compatible with (amongst others) the CDDR+ software [4] which could then be used as an input for the enumeration pipeline (as described in 1.3.2).

1.3.2 Characterization of polyhedra

For sake of completeness, we repeat here a detailed description of the method to enumerate all vertices, rays and linealities of the optimal flux balance solution space.

1. **Compute the FBA optimum.** We formulate the FBA program as the linear programming (LP) problem described in the main text. We solve the LP using QSOPT_EX version 2.5.0 [2], a rational LP-solver. Let Z^* be the optimal FBA value.

2. **Formulate the optimal FBA set.** This is done simply by replacing the objective in the LP by the optimality restriction $f(\mathbf{v}) \geq Z^*$. We write this constraint together with the set $\mathbf{Av} \geq \mathbf{b}$ of all constraints (as described in the Methods section) shortly as $\mathbf{Dv} \geq \mathbf{d}$.

3. **Perform Flux Variability Analysis (FVA).** For each flux v_j, $j = 1, \ldots, r$ we solve, using QSOPT_EX, two linear programs: $F_j^+ = \{\max v_j \mid \mathbf{Dv} \geq \mathbf{d}\}$ and $F_j^- = \{\min v_j \mid \mathbf{Dv} \geq \mathbf{d}\}$.

4. **Remove fixed fluxes.** For each variable v_j for which $F_j^+ = F_j^-$, remove from $\mathbf{D}$ the corresponding column $\mathbf{D_j}$ and subtract $F_j^+\mathbf{D_j}$ from $\mathbf{d}$. Delete the rows that have now become all-0-rows. Let the new system be $\mathbf{D'v'} \geq \mathbf{d'}$.

5. **Compute a basis for the lineality space.** The lineality space of the polyhedron is given by the null-space of $\mathbf{D'}$, i.e., all solutions to the system $\mathbf{D'v'} = \mathbf{0}$. Compute a basis for this linear subspace using a linear algebra library (such as JLINALG [1]).

6. **Compute rays and vertices of the system $\mathbf{D'v'} \geq \mathbf{d'}$.** For genome-scale systems we use the enumeration program POLCO (version 4.2.0) for this [13]. Note that POLCO automatically detects whether the system has a lineality space, but it does not report a basis for it, it only returns rays and vertices.

7. **Reintroduce the fixed fluxes that were removed earlier.** In each of the vertices reintroduce the fluxes that are fixed across all optima and were removed. Note the latter fluxes have value 0 in rays and linealities.

1.4 Determining metabolic modules/subnetworks

The analysis of the enumeration pipeline output, as described above, was performed with custom scripts developed using the Python based PYSCES-CBM framework [7].

One of the greatest challenges in this process was developing efficient analysis algorithms that could deal with the size and complexity of genome scale reconstruction enumerations. While core and reduced models may be analyzed by hand, genome scale reconstructions typically have large and complex metabolic networks e.g. iAF1260 has over 2000 reactions and over a million vertices. In order to overcome this problem the data generated in the enumeration pipeline was converted to and stored using the Hierarchical Data Format version 5 (`http://www.hdfgroup.org/HDF5`).

The complete metabolic subnetwork/module analysis was performed in three steps and the tools, data files and a worked example based on the analysis of iJR904 are available on the MEMESA Tools web site (`http://code.google.com/p/memesa-tools`).

1.4.1 Data transformation and integration

The first step takes as input the rational-arithmetic FVA and enumeration results generated by the enumeration pipeline (see Section 1.3.2) as well as the original SBML model definition used to generate the H-format model descriptions. This data is then converted into a floating point HDF5 data structure containing the ray, lineality and vertex vectors using the H5Py package (`http://alfven.org/wp/hdf5-for-python`). Once translated the vertex array $\mathbf{K}$ is scanned for constant/variable fluxes and this information is exported as meta-data to be used in subsequent analyses.

1.4.2 Calculating the correlation between variable fluxes

We now generate a sub-matrix $\mathbf{K}'$ by removing the fixed fluxes from the vertex matrix $\mathbf{K}$. Using $\mathbf{K}'$ the correlation coefficients are calculated using the Numerical Python (`http://numpy.scipy.org`) algorithm *numpy.corrcoef* which are then stored as the correlation coefficient matrix, $\mathbf{P}$.

1.4.3 Determining metabolic subnetworks/modules

To determine the metabolic modules/subnetworks we first define the adjacency matrix $\mathbf{A}$ of a graph with nodes the indices of the rows (columns) of $\mathbf{P}$: $\mathbf{A}_{m,n} = 1$ if and only if $\mathbf{P}_{m,n} \neq 0$, and $\mathbf{A}_{m,n} = 0$ otherwise. From $\mathbf{A}$ the connected component subgraphs $\mathbf{S_1} \ldots\ \mathbf{S_n}$ are determined and extracted using the NetworkX package [5]. Each subgraph, corresponds to a metabolic module/subnetwork whose nodes are fluxes. For each metabolic module/subnetwork a pattern matching algorithm is used to determine the number of unique flux distributions that occur within a particular module, across all vertices. These vertices represent the intra-module variability.

1.5 The consequences of artificially transforming the lineality space and rays into vertices

Often, flux infinity constraints ($v_j \leq \infty$ and $-\infty \leq v_j$) are encoded by choosing a sufficiently large constant M to represent infinity, we henceforth call such constraints "artificial infinity bounds". This procedure will not affect the solution to the FBA program as long as M is larger than any realistic values fluxes in the system could take. We simulated this by augmenting the FBA program for the toy model described in the main text with constraints to place an upper bound of 1000 on all fluxes, and a lower bound of -1000. Clearly, the FBA optimum remains 1. We let P' denote the space of all solutions which attain this optimum. Recall that P, the polyhedron defined in the previous system, had a lineality space of dimension 2, 1 ray and 4 vertices. In contrast, P' has neither a lineality space nor rays, because all reactions are bounded above and below by 1000 and -1000 respectively. P' is a polytope and its inner description comprises of only vertices. For this we used the rational-arithmetic variant of the CDD software (CDDR+) [4] to generate the vertices which reports 32 vertices. In this case, the origin of the vertices of P' is clear. Informally they are obtained by choosing one of the $2^2 = 4$ vertices of P, then deciding whether to have the single ray of P contribute as little as possible (i.e. multiplicative coefficient 0) or as much flux as possible (coefficient 999), and then deciding whether to run each basis vector of the lineality space of P as much as possible in one direction (coefficient 1000) or the other (coefficient -999). This gives rise to $2^2 2^1 2^2 = 32$ vertices. The details of the this analysis and results are available on the MEMESA Tools website `http://code.google.com/p/memesa-tools`.

The relationship between the (number of) vertices of a system with artificial infinity bounds, and the lineality space, rays and vertices of the same system without those bounds, will in general be more complex than in this simple example. However, it is clear that the former can very easily grow exponentially quickly with regard to the latter, and obfuscate the true structure of the unbounded polyhedron. For this reason artificial infinity bounds should be avoided to allow for explicit identification of the lineality space and rays.

Identification of metabolic subnetworks

Here we describe how we computationally identify metabolic subnetworks from a description of all the vertices of a polyhedron representing an optimal FBA set. In the last section, we described how these vertices can be obtained. A vertex is a flux vector, i.e. it has flux values as entries, and it has as many entries as there exist reactions in the network, i.e. r. We define $\mathbf{K}$ as the matrix that contains all the vertices as its columns and $\mathbf{K}_i$ as its i-th column. We remove the rows of $\mathbf{K}$ that refer to the reactions that have fixed flux values across all FBA optima to obtain $\mathbf{K}'$. From $\mathbf{K}'$ we would like to identify subnetworks that display independent flux variability in the FBA optimum. These subnetworks have the following properties: (i) they contain reactions that inter-convert a set of metabolites, such that all these reactions in the subnetwork are linked via a path of reactions, (ii) they have input and output reactions with fixed fluxes (fixed to their values of the FBA optimum), (iii) they have fixed overall reaction stoichiometry (because subnetworks communicate with the remainder of the metabolic network through a set of fixed fluxes), (iv) subnetworks are connected by paths of reactions that carry fixed fluxes and (v) multiple routes exist inside subnetworks that give rise to the subnetwork's overall reaction stoichiometry.

Which reactions constitute the subnetworks and their overall reaction stoichiometry depends on the specific FBA problem. If two reactions occur in the same subnetwork (and are not input or output fluxes), they have flux variability in the FBA optima and their flux values across all the vertices will correlate: they have correlation coefficient 1 when they always occur in the same optimal flux route(s) of the subnetwork, correlation coefficient -1 when they exclusively carry flux in alternative routes within the subnetwork, and correlation coefficients between -1 and 1 otherwise. Two reactions that occur in different subnetworks will have correlation coefficient 0. Hence, we can identify the subnetworks by determining the Pearson's correlation coefficients of the flux values of all the pairs of reactions in $\mathbf{K}'$. In all the cases that we analyzed we found that the reactions that correlated formed a connected subnetwork of the network consisting of the variable reactions only.

References

[1] JLinAlg: An open source and easy-to-use java library for linear algebra. http://jlinalg.sourceforge.net/.

[2] David Applegate, William Cook, Sanjeeb Dash, and Daniel Espinoza. QSopt_ex: Rational LP solver. http://www2.isye.gatech.edu/~wcook/qsopt/ex/index.html.

[3] Frank Bergmann and Brett G. Olivier. SBML Level 3 Package Proposal: Flux, 2010. http://dx.doi.org/10.1038/npre.2010.4236.1.

[4] Komei Fukuda. cdd and cddplus homepage. http://www.cs.mcgill.ca/˜fukuda/soft/cdd%5Fhome/cdd.html.

[5] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA, August 2008.

[6] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, and H. Kitano. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[7] B G Olivier. PySCeS-CBM: a toolkit for Constraint Based Modelling in Python, 2011. http://pysces.sf.net/cbm.

[8] B G Olivier, J M Rohwer, and J H Hofmeyr. Modelling cellular systems with PySCeS. *Bioinformatics*, 21(4):560–561, Feb 2005.

[9] J L Reed, T D Vo, C H Schilling, and B O Palsson. An expanded genome-scale model of escherichia coli k-12 (ijr904 gsm/gpr). *Genome Biol*, 4(9), 2003.

[10] J Schellenberger, R Que, R M Fleming, I Thiele, J D Orth, A M Feist, D C Zielinski, A Bordbar, N E Lewis, S Rahmanian, J Kang, D R Hyduke, and B Ø Palsson. Quantitative prediction of cellular metabolism with constraint-based models: the cobra toolbox v2.0. *Nat Protoc*, 6(9):1290–1307, 2011.

[11] Jan Schellenberger, Junyoung Park, Tom Conrad, and Bernhard Palsson. Bigg: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions. *BMC Bioinformatics*, 11(1):213, 2010.

[12] Alexander Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons, June 1988.

[13] M Terzer and J Stelling. Large-scale computation of elementary flux modes with bit pattern trees. *Bioinformatics*, 24(19):2229–2235, Oct 2008.

[14] Gunter M. Ziegler. *Lectures on Polytopes (Graduate Texts in Mathematics).* Springer, 2001.