Hierarchical decomposition of metabolic networks using *k*-modules

Arne C. Reimers[*1]
[*]Centre for Mathematics and Computer Science (CWI), Science Park 123, 1098 XG Amsterdam, The Netherlands

Abstract

The optimal solutions obtained by flux balance analysis (FBA) are typically not unique. Flux modules have recently been shown to be a very useful tool to simplify and decompose the space of FBA-optimal solutions. Since yield-maximization is sometimes not the primary objective encountered *in vivo*, we are also interested in understanding the space of sub-optimal solutions. Unfortunately, the flux modules are too restrictive and not suited for this task. We present a generalization, called k-module, which compensates the limited applicability of flux modules to the space of sub-optimal solutions. Intuitively, a k-module is a sub-network with low connectivity to the rest of the network. Recursive application of *k*-modules yields a hierarchical decomposition of the metabolic network, which is also known as branch decomposition in matroid theory. In particular, decompositions computed by existing methods, like the null-space-based approach, introduced by Poolman et al. [(2007) J. Theor. Biol. 249, 691–705] can be interpreted as branch decompositions. With *k*-modules we can now compare alternative decompositions of metabolic networks to the classical sub-systems of glycolysis, tricarboxylic acid (TCA) cycle, etc. They can be used to speed up algorithmic problems [theoretically shown for elementary flux modes (EFM) enumeration] and have the potential to present computational solutions in a more intuitive way independently from the classical sub-systems.

Introduction

Constraint based methods have proven to be very successful in the analysis of metabolic networks [1,2], which are used to model metabolic capabilities and predict behaviours of organisms. In contrast with kinetic models, constraint-based metabolic network models do not aim to predict a single phenotype, but a space of biologically possible phenotypes. This is achieved by excluding unrealistic phenotypes using constraints. This reduces the data requirements enormously such that also large models with thousands of reactions can be built.

Because of the size of the networks, however, even the interplay of very simple constraints can yield very complex and high-dimensional solution spaces that are very hard to comprehensively analyse. This is already the case for networks solely based on the steady-state assumption and irreversibility constraints, which are the basic assumptions for methods like flux balance analysis (FBA) [3,4] and related methods. The steady-state assumption states that every metabolite must be produced at the same rate as it is consumed. Formally, a vector of reaction rates (flux vector) $v \in \mathbb{R}^{\mathcal{R}}$ is in steady-state if it satisfies

$$Sv = 0,$$

where S is the stoichiometric matrix. We use $\mathcal{R}$ to denote the set of all reactions and $\mathcal{M}$ to denote the set of all metabolites.

Key words: branch decomposition, connectivity, flux module, k-module, matroid, metabolic network.

Abbreviations: EFM, elementary flux modes; FBA, flux balance analysis; NP, non-deterministic polynomial time.

[1] To whom correspondence should be addressed (email arne.c.reimers@gmail.com).

With a set Irrev $\subseteq \mathcal{R}$ of reactions that are only allowed to operate in forward direction, the full steady-state flux space is obtained, as given below:

$$\{v \in \mathbb{R}^{\mathcal{R}} : S_{\mathrm{V}} = 0,\ v_{\mathrm{Irrev}} \geqslant 0\}$$

Although extreme pathways [5] or elementary flux modes (EFM) [6,7] can comprehensively characterize the solution space based on easily understandable pathways, the number of pathways explodes with the size of the network. This makes these approaches only applicable to small networks.

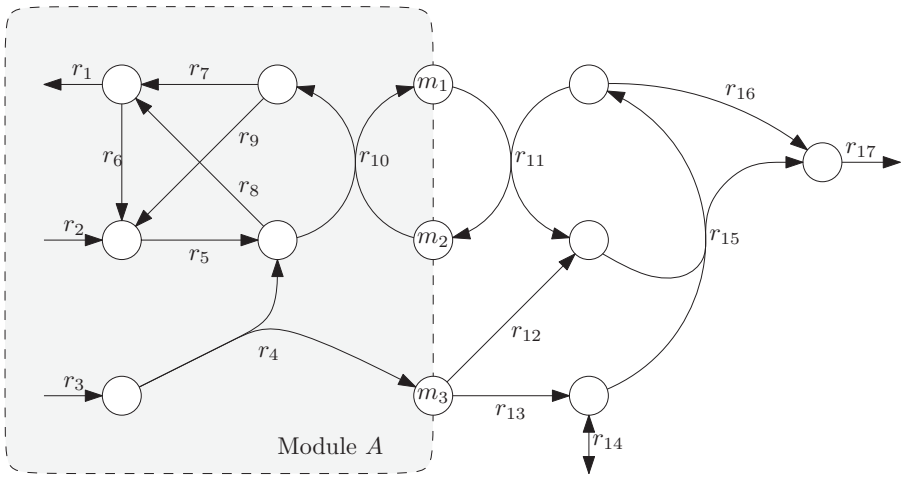
Therefore, many methods try to determine only special properties of the network. For example, FBA asks for the maximal biomass yield for a given uptake of nutrients [3,4]. Although the space of optimal yield fluxes (optimal yield flux space) also contains many solutions [8,9]; Kelk et al. [10] discovered a method that allows a comprehensive pathway-based description for the optimal yield flux space of many genome-scale networks. They observed that the optimal yield flux space can be decomposed into flux modules. For a flux space $P \subseteq \mathbb{R}^{\mathcal{R}}$, a P-module is a set of reactions $A \subseteq \mathcal{R}$ for which there exists a vector $d \in \mathbb{R}^{\mathcal{M}}$ with

$$S_A v_A = d \quad \text{for all } v \in P,$$

where S_A denotes the sub-matrix of S with only columns corresponding to the reactions in A. Similarly, v_A is the sub-vector of v with only entries corresponding to reactions in A. With this definition, originally introduced by Müller and Bockmayr [11], the flux modules can be efficiently computed [12]. By computing the pathways through each module, a comprehensive pathway-based description can be obtained

Figure 1 | **Example of a toy network**

The reactions in $A = \{r_1, \ldots, r_{10}\}$ have three boundary metabolites and form a 2-module.



efficiently [13]. However, this unfortunately only works for the optimal yield space, because for the full steady-state flux space (without yield-optimality condition) no interesting flux modules can typically be found.

In the study by Reimers and Stougie [14], we introduced the concept of k-modules to overcome the limitations of flux modules. There, we followed a mathematical approach and considered the general problem of vertex enumeration of polyhedra. Here, we will now focus on the application to metabolic networks and the biological interpretation of k-modules, while keeping the mathematical overhead to a minimum.

With k-modules, we can define a hierarchical decomposition of metabolic networks that is similar to tree-decompositions and tree-width in graph theory [14]. In parameterized complexity theory, there exist many results that show that, if a graph has low tree-width, many otherwise NP-hard (problems that can not be solved in polynomial time unless all problems that can be solved in non-deterministic polynomial time (NP) can be solved in polynomial time) problems can be solved efficiently in polynomial time [15,16]. This gives us the chance to also obtain similar results for metabolic networks.

In the section on 'k-modules', we will introduce and define k-modules. They will then form the basis for the hierarchical decompositions discussed in the section 'Branch decomposition'. Finally, an application is given in the section 'Application: EFM enumeration' by considering the problem of EFM enumeration.

k-modules

Let us consider the network shown in Figure 1 and the set $A = \{r_1, \ldots, r_{10}\}$ of reactions. The metabolites $B = \{m_1, m_2, m_3\}$ are each involved in a reaction of A and also in one of the other reactions. These metabolites form the boundary of A and therefore connect A to the rest of the network. Therefore, we call them the boundary metabolites $B(A)$ of A:

$$B(A) := \{m \in \mathcal{M} : S_{mr} \neq 0 \neq S_{ms}\ \exists r \in A, s \notin A\}$$

We observe that for any set of reactions A, we can compute how well A is connected to the rest of the network using the formula below:

$$\mu(A) := |B(A)|.$$

We argue that a set of reactions A with low connectivity $\mu(A)$ should be easy to analyse by itself, because the interaction with the rest of the network that could influence the interpretation of A is low.

However, we also observe for the set A from Figure 1 that m_1 is always produced at the rate by which m_2 is consumed (in real networks this can happen for example with currency metabolites like ATP and ADP). Hence, the interaction of A through m_2 is already given by the interaction through m_1 with the rest of the network. To deal with such redundancies, we use the concept of k-modules.

A set of reactions $A \subseteq \mathcal{R}$ is a P-k-module, if there exists a $d \in \mathbb{R}^k$ and a matrix $D \in \mathbb{R}^{\mathcal{M} \times k}$ (interface) such that for every $v \in P$ there exists an $\alpha \in \mathbb{R}^k$ with

$$S_A v_A = d + D\alpha$$

We simply write k-module if P is the full steady-state flux space and the network contains no blocked reactions.

The set A from the example of Figure 1 is a 2-module, because we can choose

$$D = \begin{matrix} m_1 \\ m_2 \\ m_3 \end{matrix} \begin{pmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \end{pmatrix}$$

The notion of k-module allows us now to define an alternative connectivity function that does not only consider network topology but also stoichiometries:

$$\lambda(A) := min\{k : A \text{ is a } k - \text{module}\}$$

To improve the reader's understanding of k-modules and their relation to the previously defined P-modules, we here list a few properties that hold in general:

- $\mu(A) = \mu(\mathcal{R} \setminus A)$
- $\lambda(A) = \lambda(\mathcal{R} \setminus A)$
- $\lambda(A) \leqslant \mu(A)$
- $\lambda(A) \leqslant |A|$ (note that $\mu(A) \leqslant |A|$ does not hold in general)
- $\lambda(A) \leqslant |\mathcal{R} \setminus A|$
- A is a P-module if and only if A is a P-0-module (i.e., A is a 0-module if P is the full steady-state flux space without blocked reactions).

Furthermore, we want to remark that the simplifications [12] that make an efficient computation of flux modules possible, also apply to k-modules. This means that as soon as all reactions with fixed reaction rate have been identified, the connectivity function λ can be computed based on linear algebra alone [14], i.e. we have:

$$\lambda(A) = \lambda(A \cap V),$$

where V is the set of reactions with variable flux. Furthermore, if all reactions can carry variable flux, then λ depends on the stoichiometric matrix alone. For $V = \mathcal{R}$, this leads to the surprising result that [14]:

$$\lambda(A) = \operatorname{rank}(S_A) + \operatorname{rank}(S_{\mathcal{R}\setminus A}) - \operatorname{rank}(S_{\mathcal{R}}).$$

Since the rank of a matrix can be efficiently computed, we can also compute λ efficiently. For example, in the case of the *Escherichia coli* iAF1260 network and its sub-system annotations, we computed that glycolysis/gluconeogenesis has a connectivity of 15 and the citric acid cycle has a connectivity of 12. A complete list can be found in the Supplementary Material.

Branch decomposition

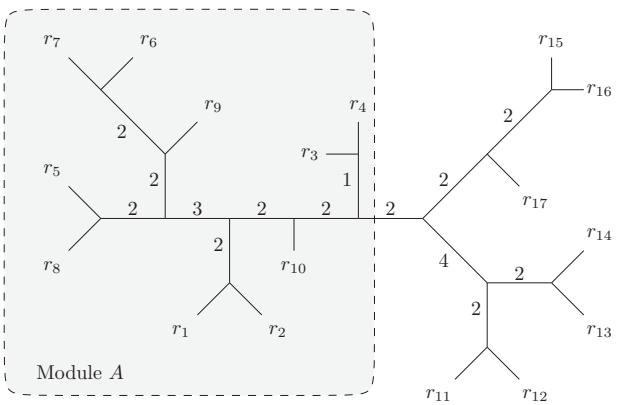
Although we can compute $\lambda(A)$ efficiently for a given A, we are still left with the problem of finding 'interesting' sets of reactions A with low $\lambda(A)$. In particular, it is not really clear what an 'interesting' set of reactions is. We observe by the properties mentioned above that there are many sets of reactions, where $\lambda(A)$ is low (for example if A contains only one reaction or if A contains all but one reaction), but which are clearly not interesting.

We conclude that we want to find sets of reactions A where A is large, the complement $\mathcal{R} \setminus A$ is large and $\lambda(A)$ is low. However, if A contains many reactions, we will be interested to understand A more deeply. Hence, we want to be able to split A recursively into smaller k-modules. This leads us to the concept of branch decompositions and branch width [17,18]. Branch width is related to the more popular concept of tree width, which is used to measure how tree-like a graph is. In contrast with tree width, branch-width has a natural extension to matroids and, thus, also to metabolic networks.

A branch-decomposition is a sub-cubic tree, i.e. all nodes have either degree 3 or they are leaves. Every leaf is uniquely associated to a reaction of the network. An example is shown in Figure 2. We observe that if we remove an edge e of the tree, we get two connected components. Let A be the set of reactions associated to the leaves of one of the connected components. We observe that $\mathcal{R} \setminus A$ is the set of reactions associated to the leaves of the other connected component. Hence, we can annotate the edge e with the value of the connectivity function $\lambda(A) = \lambda(\mathcal{R} \setminus A)$. Note that, alternatively, we can do the same with the connectivity function μ. Furthermore, we observe that by deleting the edge e, we get two rooted binary trees, each rooted at a vertex that was incident to e. These rooted binary trees give a straightforward rule on how to recursively split the k-module A (respectively the k-module $\mathcal{R} \setminus A$) into two smaller k-modules. In the example of Figure 2 the reaction set $A = \{r_1, \ldots, r_{10}\}$ would be split up into the reaction set $\{r_1, r_2, r_5, \ldots, r_{10}\}$ with connectivity 2 and the fully coupled reactions $\{r_3, r_4\}$ with connectivity 1.

Figure 2 | A branch decomposition of the example network in Figure 1

The edges are annotated with the corresponding value of the connectivity function, except edges incident to leaves. Edges incident to leaves correspond to k-modules containing only one reaction. Hence, they have connectivity 1.

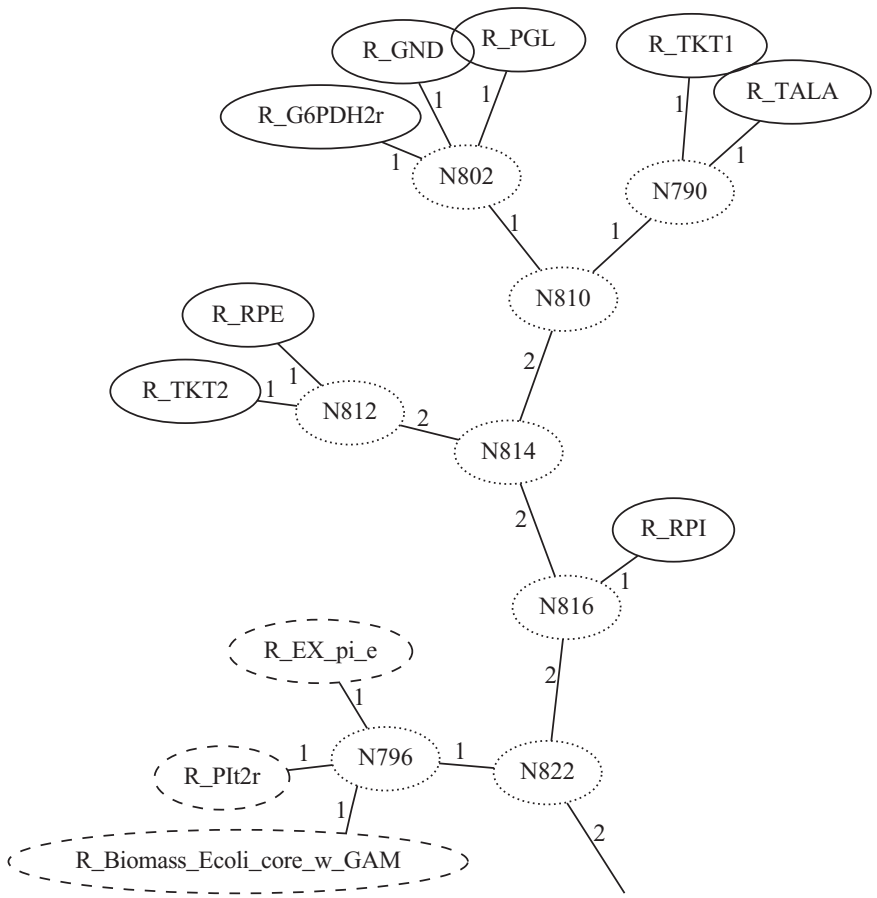


The largest value with which an edge of a branch-decomposition is annotated is called the branch width of the branch decomposition. For the example of Figure 1, we see in Figure 2 a branch-decomposition with branch width 4, because $A' := \{r_{11}, r_{12}, r_{13}, r_{14}\}$ is only a 4-module, i.e. $\lambda(A') = 4$. We can now turn the question of finding an interesting k-module into the question of finding a branch-decomposition with low branch width. In particular, we can consider a metabolic network modular if we can find a branch-decomposition with low branch width.

Computing branch decompositions for metabolic networks

Unfortunately, it is NP-hard to find a branch-decomposition with minimal branch width. There exist theoretical results on how to solve this problem in polynomial time for low branch-width instances [19,20]. However, these algorithms are not

Figure 3 | **Excerpt of branch decomposition computed using a variant of Poolman's method [22] for the *E. coli* core network**
All non-dashed leaves belong to the pentose phosphate pathway.



likely to be practical, which is why we need to use heuristics. Heuristics have been developed by Ma et al. [21]. Core idea is to compute a similarity matrix by computing the similarity for each pair of reactions. Interestingly, Ma et al. [21] use a similarity measure, which is closely related to the one introduced by Poolman et al. [22]. Indeed, the decomposition computed by Poolman et al. [22] is a branch decomposition. In Figure 3 an excerpt of the branch decomposition computed for an *E. coli* core network [23] is shown. The full version can be found in the Supplementary Material.

In Table 1 we have listed upper bounds on the branch widths for a set of genome-scale networks computed using variants of Poolman's method. The methods for computing the branch decompositions are described in the Supplementary Material and they are implemented in the cbmpy toolbox (cbmpy.sourceforge.net).

Table 1 | **Upper bounds on branch-width for some genome-scale metabolic networks**

Network	Reactions	Branch width
E. coli core	95	13
E. coli iJR904	1075	40
E. coli iAF1260	2382	59
Helicobacter pylori iIT341	554	26
Homo sapiens recon 1	3742	99
H. sapiens recon 2	7440	146
Methanosarcina barkeri iAF692	690	29
Mycobacterium tuberculosis iNJ661	1025	35
Staphylococcus aureus iSB619	743	39
Saccharomyces cerevisiae iND750	1266	53

Application: EFM enumeration

In the study by Reimers and Stougie [14], we have shown that the set of EFM can theoretically be efficiently enumerated if the branch width of the network is low. To be more precise, we show that given a branch decomposition of branch-width k, we can enumerate all EFM in time

$$O(|\mathcal{R}||\mathrm{EFM}|^{2k+2}t)$$

where t is the time needed to solve a linear program (LP) and |EFM| is the number of EFM [14]. Although this is the first result that shows that EFM can be enumerated in total polynomial time, it unfortunately is not very practical due to the $|\mathrm{EFM}|^{2k+2}$ term.

Without going into details (we refer to Reimers and Stougie's work [14] for that), the idea is to enumerate recursively pathways that correspond to EFM through the k-modules in the branch decomposition, i.e. the pathways of a k-module C, which is split into two k-modules A and B, can be computed from the pathways through A and B. The bad runtime bound arises from the fact that it is hard to bind the number of pathways through each k-module well enough.

Conclusion

In this article we have shown how we can use k-modules to measure how well connected sub-systems are to the rest of the network. Whereas this measure is similar to counting the number of metabolites on the boundary, it is stoichiometry-based and hence it smoothly deals with redundancies due to coupled metabolites.

By recursively decomposing a network using branch decompositions, k-modules give us a measure of modularity. Unfortunately, it is very hard to compute the best branch decomposition. Therefore, we use heuristics, such as the method by Poolman et al. [22]. Although this method gives us a branch decomposition from which we can recognize familiar sub-networks, its branch width is not very small. In addition, the connectivity of many subsystems as annotated in the *E. coli* iAF1260 model is also very large compared with their size. Therefore, we conclude that the high branch width computed by our algorithm is probably not due to its lack of a quality guarantee, but because metabolic networks are not very modular (in the k-modules sense).

Acknowledgements

I thank Timo Maarleveld, Frank Bruggeman, Brett Olivier, Marie-France Sagot and Leen Stougie for insightful discussions.

Funding

This work was supported by the European Research Consortium for Informatics and Mathematics through an Alain Bensoussan Fellowship.

References

1 Papin, A.J., Stelling, J., Price, N.D., Klamt, S., Schuster, S. and Palsson, B.Ø. (2004) Comparison of network-based pathway analysis methods. Trends Biotechnol. **22**, 400–405 CrossRef PubMed
2 Price, N.D., Reed, J.L. and Palsson, B.Ø. (2004) Genome-scale models of microbial cells: evaluating the consequences of constraints. Nat. Rev. Microbiol. **2**, 886–897 CrossRef PubMed
3 Orth, J.D., Thiele, I. and Palsson, B.Ø. (2010) What is flux balance analysis? Nat. Biotechnol. **28**, 245–248
4 Varma, A. and Palsson, B.Ø. (1994) Metabolic flux balancing: basic concepts, scientific and practical use. Nat. Biotechnol. **12**, 994–998 CrossRef
5 Schilling, C.H., Letscher, D. and Palsson, B.Ø. (2000) Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. J. Theor. Biol. **203**, 229–248 CrossRef PubMed
6 Schuster, S. and Hilgetag, C. (1994) On elementary flux modes in biochemical systems at steady state. J. Biol. Syst. **2**, 165–182 CrossRef
7 Schuster, S., Fell, D.A. and Dandekar, T. (2000) A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. Nat. Biotechnol. **18**, 326–332 CrossRef PubMed
8 Mahadevan, R. and Schilling, C. (2003) The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. Metab. Eng. **5**, 264–276 CrossRef PubMed
9 Khannapho, C., Zhao, H., Bonde, B.L., Kierzek, A.M., Avignone-Rossa, C.A. and Bushell, M.E. (2008) Selection of objective function in genome scale flux balance analysis for process feed development in antibiotic production. Metab. Eng. **10**, 227–233 CrossRef PubMed
10 Kelk, S.M., Olivier, B.G., Stougie, L. and Bruggeman, F.J. (2012) Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. Sci. Rep. **2**, 580 CrossRef PubMed
11 Müller, A.C. and Bockmayr, A. (2014) Flux modules in metabolic networks. J. Math. Biol. **69**, 1151–1179 CrossRef PubMed
12 Reimers, A.C., Bruggeman, F.J., Olivier, B.G. and Stougie, L. (2015) Fast flux module detection using matroid theory. J. Comput. Biol. **22**, 414–424 CrossRef PubMed
13 Maarleveld, T.R., Wortel, M., Olivier, B.G., Teusink, B. and Bruggeman, F.J. (2015) Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models. PLoS Comput. Biol. **11**, e1004166 CrossRef PubMed
14 Reimers, A.C. and Stougie, L. (2014) A decomposition theory for vertex enumeration of convex polyhedra., arXiv:1404.5584 [cs.CG]
15 Arnborg, S. (1985) Efficient algorithms for combinatorial problems on graphs with bounded decomposability - a survey. BIT Numerical Math. **25**, 1–23 CrossRef
16 Cook, W. and Seymour, P. (2003) Tour merging via branch-decomposition. Informs J. Comput. **15**, 233–248 CrossRef
17 Hicks, I.V., Koster, A.M.C.A. and Kolotoğlu, E. (2005) Branch and tree decomposition techniques for discrete optimization. In Tutorials in Operations Research. Emerging Theory, Methods, and Applications, pp. 1–29. INFORMS 2005
18 Hicks, I.V. and Oum, S.-I. (2011) Branch–width and tangles. In Wiley Encyclopedia of Operations Research and Management Science (Cochran, J.J., Cox, Jr, L.A., Keskinocak, P., Kharoufeh, J.P. and Smith, J.C., eds), Wiley
19 Oum, S.-I. and Seymour, P. (2006) Approximating clique-width and branch-width. J. Combin. Theory Ser. B **96**, 514–528 CrossRef
20 Oum, S.-I. and Seymour, P. (2007) Testing branch-width. J. Combin. Theory Ser. B **97**, 385–393 CrossRef
21 Ma, J., Margulies, S., Hicks, I.V. and Goins, E. (2013) Branch decomposition heuristics for linear matroids. Discrete Optimization. **10**, 102–119 CrossRef
22 Poolman, M.G., Sebu, C., Pidcock, M.K. and Fell, D.A. (2007) Modular decomposition of metabolic systems via null-space analysis. J. Theor. Biol. **249**, 691–705 CrossRef PubMed
23 Orth, J.D., Fleming, R.M. and Palsson, B.Ø. (2010) Reconstruction and use of microbial metabolic networks: the core *Escherichia coli* metabolic model as an educational guide. EcoSal Plus, doi: 10.1128/ecosalplus.10.2.1

Received 22 June 2015
doi:10.1042/BST20150143