Enumerating Chemical Organisations in Consistent Metabolic Networks: Complexity and Algorithms

Paulo Vieira Milreu, Vicente Acuña, Etienne E. Birmelé, Pierluigi Crescenzi, Alberto Marchetti-Spaccamela, Marie-France Sagot, Leen Stougie, Vincent Lacroix

Enumerating Chemical Organisations in Consistent Metabolic Networks: Complexity and Algorithms

P. V. Milreu[1,2], V. Acuña[1,2], E. Birmelé[3], P. Crescenzi[4], A. Marchetti-Spaccamela[5], M.-F. Sagot[1,2], L. Stougie[6,7], and V. Lacroix[1,2]

[1] Université de Lyon, F-69000 Lyon; Université Lyon 1; CNRS, UMR5558, Laboratoire de Biométrie et Biologie Evolutive, F-69622 Villeurbanne, France
[2] INRIA Rhône-Alpes, 38330 Montbonnot Saint-Martin, France
{milreu,viacuna,lacroix}@biomserv.univ-lyon1.fr, marie-france.sagot@inria.fr
[3] Lab. Statistique et Génome, CNRS UMR8071 INRA1152, Université d'Évry, France
etienne.birmele@genopole.cnrs.fr
[4] Università di Firenze, Dipartimento di Sistemi e Informatica, I-50134 Firenze, Italy
pierluigi.crescenzi@unifi.it
[5] Sapienza University of Rome, Italy, alberto.marchetti@dis.uniroma1.it
[6] VU University and CWI, Amsterdam, The Netherlands, lstougie@feweb.vu.nl

Abstract. The structural analysis of metabolic networks aims both at understanding the function and the evolution of metabolism. While it is commonly admitted that metabolism is modular, the identification of metabolic modules remains an open topic. Several definitions of what is a module have been proposed. We focus here on the notion of chemical organisations, i.e. sets of molecules which are closed and self-maintaining. We show that finding a reactive organisation is NP-hard even if the network is mass- and flux-consistent and that the hardness comes from blocking cycles. We then propose new algorithms for enumerating chemical organisations that are theoretically more efficient than existing approaches.

1 Introduction

Until recently, metabolism was analysed via the pathways composing it, which were traditionally established in a non automatic way by experts interested in some specific function (glycolysis for instance, or anaerobic respiration). Metabolic pathways may thus be seen as functional modules. One may wonder, however, about a possible arbitrariness in the definitions of the exact frontiers of each pathway, that may have been introduced by the experts because of some specific objectives they had in mind and of a partial lack of knowledge at the time the limits of the pathways were first drawn. Indeed, the advent of full genome sequences, which has also allowed inferring whole metabolic networks, has revealed the existence and actual use by different organisms of alternative metabolic routes to a same final overall product goal. The question thus rises

whether automatic methods for inferring functional modules from whole networks would enable us to preserve the expert knowledge that led to the first pathways, while at the same time providing an insight into alternative functional ones. This is the biological motivation underlying the work presented in this paper. Its computational motivation is to explore one model for metabolic modules, both in terms of the complexity of enumerating such modules and of exact algorithms for performing the enumeration.

Several formal definitions of pathways and modules can be found in the literature on metabolism, the best known of which may be elementary modes [12] or any of its close cousins (see [8, 9] for a survey). Elementary modes may be informally described as metabolic subnetworks that can function at steady state, meaning that all internal metabolites are produced and consumed in equal rates (that is, nothing accumulates internally). This is a fine definition, but has at least one drawback: it is restricted to the analysis of the system at steady state and does not allow to describe states of the system where metabolites can accumulate. However, such states are relevant as they could correspond to intermediary steps in the evolution of metabolism, or temporary states in the dynamics of metabolism.

As far as we know, two models in the literature enable to study such states. One is Petri nets and the other is a more recent model called *chemical organisations*. Because it is algebraically easier to manipulate chemical organisations as their formulation follows closely that of (hyper)graphs and matrices, we focus our attention in this paper on chemical organisations. The concept was introduced in 2005 by Peter Dittrich and his group [4], building on earlier work by Fontana and Buss [6] and can be used not only for metabolism, but also for any kind of reaction system, including regulatory networks. In this paper, however, we focus exclusively on metabolism.

Chemical organisations are sets of molecules that are self-maintaining and closed (in this paper, we use the terms metabolite and molecule with no distinction). Informally, a self-maintaining set is a set where molecules can accumulate – the feature we were seeking – provided no molecule vanishes. A set is closed if all metabolites produced from reactions for which all the inputs are present in the set will also be present and thus part of the set. By convention, this includes all reactions that take their input from the environment, *i.e.* are external. All external inputs are therefore considered as being available *and used*. This introduces a second contrast with elementary modes (EMs). Indeed, EMs may use only part of the externally available inputs. More generally, EMs are not closed.

Finally, as we have already said, the theory of chemical organisations has been proposed for general reaction systems. Its application to metabolic networks raises new specific questions, as the networks have specific properties. They are indeed expected to be mass-consistent (reactions preserve mass) and flux-consistent (each reaction belongs to at least one elementary mode).

The objectives of this paper are thus twofold. The first is to revisit chemical organisations in the context of mass- and flux-consistent networks. In particular, finding a chemical organisations was shown to be hard in [2]. A legitimate and

non trivial question is whether this remains true in biologically more realistic mass- and flux-consistent networks. Section 2 presents the main definitions on chemical organisations and consistency of networks. Section 3 shows that even for consistent networks the enumeration problem is hard. We go however further by identifying the specific structural properties of the network that account for this hardness. Those are discussed in Section 4, while Section 5 fulfills the second objective of this paper. This is to describe a new algorithm that takes advantage of such properties to obtain an exact method that is in all cases theoretically more efficient for consistent networks than the enumeration algorithms presented in [2] because, at best, a smaller part of the solution space needs to be explored.

2 Preliminaries

A metabolic network, like any reaction system, can be modelled as a *weighted directed hypergraph* $G = (M, R)$ with M the set of *vertices* corresponding to the metabolites and R the set of *hyperarcs* corresponding to the reactions. A directed hyperarc (*i.e.* a reaction) $r \in R$ is an ordered pair of sets of vertices (*i.e.* metabolites) $r = (subs(r), prod(r))$ where $subs(r)$ is the set of substrates of r and $prod(r)$ is the set of products of r. For each x in $subs(r)$ (in $prod(r)$) the weight of x with respect to r denotes the stoichiometric coefficient of x in r, that is, the number of units of x consumed (or produced) when r fires. Note that x can belong to both $subs(r)$ and $prod(r)$; in this case there are two weights associated to x w.r.t. r. Note also that, according to the above definitions, the set of substrates of a reaction r can be empty: in this case, we say that the metabolites in $prod(r)$ are *inputs* of the network.

Metabolic networks have also been often modelled using matrices [11]. The *stoichiometric matrix* S has $|M|$ rows and $|R|$ columns where $S_{i,j}$ is the stoichiometric coefficients of molecule i in reaction j. $S_{i,j}$ is negative if i is consumed and it is positive if i is produced. We notice here that while the stoichiometric matrix can always be derived from the weighted hypergraph, the reverse is not true. Indeed, metabolites involved as substrates and products of the same reaction cannot be handled in the matrix representation.

For some of the results presented, we also use the concept of the *underlying graph* of G, which is a directed multigraph with the same set of vertices of G and arcs $x \to y$ for every pair of vertices x, y for which there is an hyperarc r such that $x \in subs(r)$ and $y \in prod(r)$. A reaction is said to be on a path/cycle of the underlying directed graph if any of its (substrate,product)-pairs is an arc of the path/cycle.

In the context of metabolic networks, we say that a *flux* over the network is the rate at which each reaction occurs. A flux can be represented as a *flux vector* $v \in \mathbb{R}^{|R|}$ with $v[i]$ denoting the rate of reaction i. We also define a *mass vector* $m \in \mathbb{R}^{|M|}$ with $m[j]$ denoting the mass of metabolite j.

A metabolic network is **flux-consistent** if there exists a flux vector $v > 0$ such that $Sv = 0$ [1]. This is the same as saying that every reaction of the network

belongs to at least one elementary mode, thus checking for the usefulness of each reaction. For more information on elementary modes, see [12] and [11].

A metabolic network is **mass-consistent** if there exists a mass vector $m > 0$ such that $m^T S = 0$, where m^T denotes the transposed vector. This is the same as saying that there exists some mass distribution for all metabolites such that the whole set of reactions is mass balanced.

An example of a network with two reactions $r_1 : a+c \to b$ and $r_2 : b+d \to d+c$ that is not mass-consistent is shown in Figure 1. Since reaction r_2 requires that b and c have the same mass, then the mass of a in r_1 has to be 0, which is inconsistent. If we replace r_1 by $r_1' : a + c \to 2b$, then the system becomes mass-consistent (indeed, every positive mass vector with equal components is consistent).

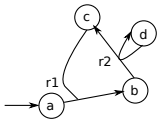

Fig. 1. Metabolic network whose mass-consistence depends on the stoichiometric coefficients of the first reaction.

We denote by $\mathcal{R}_A \subseteq R$ the subset of reactions that can be fired when the metabolites in set $A \subseteq M$ are present, *i.e.*, $\mathcal{R}_A = \{r \in R | subs(r) \subseteq A\}$. We now introduce the basic definitions that will be used throughout the paper.

Definition 1. *A set $C \subseteq M$ is **closed** if, for all reactions $r \in \mathcal{R}_C$, $prod(r) \subseteq C$. Moreover, given a set $C \subseteq M$, the **closure** of C, denoted by Cl_C, is the smallest closed set H that contains C.*

Note that if C is a closed set of molecules, then C must contain all inputs of the network (since the empty set is a subset of C and input reactions therefore belong to $\mathcal{R}_C$). In particular, the closure of the empty set will contain all inputs and whatever can be produced from them.

Definition 2. *A set of molecules $C \subseteq M$ is **self-maintaining** if there is a flux vector v such that:*

1. *for all reactions $r \in \mathcal{R}_C$, $v[r] > 0$;*
2. *for all reactions $r \notin \mathcal{R}_C$, $v[r] = 0$;*
3. *for all molecules $i \in C$, the production rate $(Sv)[i] \geq 0$.*

Definition 3. *A set of molecules $O \subseteq M$ is an **organisation** if it is closed and self-maintaining. O is said to be **reactive connected** if:*

- *(reactive) each metabolite in O takes part as substrate or product in at least one reaction inside $\mathcal{R}_O$;*

– *(connected) for any two molecules x and y in O, there is a path from x to y in the underlying undirected graph.*

The motivation to find only reactive connected organisations is illustrated in the example of Figure 2 where the network has 3 connected components and 8 organisations but only 2 of them are reactive connected organisations. The others are just combinations of organisations which cannot directly interact among them.

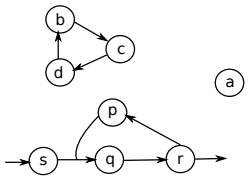

Fig. 2. A metabolic network with (a) 3 connected components, (b) 8 organisations: $\{\{s\},\{s,p,q,r\},\{s,a\},\{s,b,c,d\},\{s,a,b,c,d\},\{s,a,p,q,r\},\{s,b,c,d,p,q,r\},\{s,a,b,c,d,p,q,r\}\}$, and (c) 2 reactive connected organisations: $\{\{s\},\{s,p,q,r\}\}$.

For this reason, from now on we restrict our networks to have only one connected component and some input and output metabolites. For general networks, indeed, we can without loss of generality work on each connected component separately and then combine the results.

Since throughout the paper we need to compute closures of sets, we recall here the forward propagation procedure [10] that in an iterative process enables to obtain the closure of a given set C. Informally, this consists in starting from C itself, adding $prod(r)$ for every $r \in \mathcal{R}_C$, and repeating this procedure until no new metabolites are added to C. The detailed description of the forward propagation algorithm is presented in the Appendix.

As already mentioned, all inputs of the network need to be considered together in order to compute organisations. This is a modelling choice that implies that if one wished to compute organisations for different subsets of the inputs, then it would be necessary to edit the network and recompute the organisations for the subsets of interest.

3 Chemical Organisations in Consistent Networks

It has been shown that deciding whether a network contains one organisation is NP-complete [2]. However the proof was based on a network that was neither flux- nor mass-consistent. We now characterise organisations in consistent networks. First of all, we observe that it is easy to check whether a set C is an organisation by inspecting the reaction rules to check closure and self-maintenance using linear programming.

The following theorem shows how to compute two possible organisations.

Theorem 4. *If a network is flux-consistent then the whole network and the closure of the empty set are organisations.*

Proof. The whole network is always closed. By definition of flux-consistency, we have a flux vector v that covers the whole network, satisfying the condition of self-maintenance. Therefore the whole network is an organisation. Analogously, if the closure of the empty set produces the whole network then it is an organisation. Otherwise, since every metabolite is produced from the empty set, we can easily obtain a valid flux vector v satisfying the condition of self-maintenance. An algorithm to find a flux for this situation is presented in the Appendix. □

In the following, we say that the whole network and the closure of the empty set are **trivial organisations**.

Observe that the closure of the empty set may not always produce the whole network. An example is given in Figure 1 since the closure of the empty set for that network is $\{a\}$.

Theorem 5. *If the network is flux-consistent and acyclic, i.e. the underlying directed graph of the hypergraph is acyclic, then the whole network is the only organisation.*

Proof. The smallest organisation is given by the closure of the empty set, which can be obtained by applying the forward propagation algorithm to the empty set. As the network is flux-consistent and acyclic, from the inputs any metabolite can be reached, *i.e.*, produced. Hence, the closure of the empty set is the entire network. From the flux-consistency of the network and from Theorem 4, it follows that the smallest organisation is the whole network. □

The next result shows that the problem of finding a non trivial organisation in a flux-consistent network is NP-hard. The proof (which is given in the Appendix) is based on a reduction from the 3-SAT problem, which is an appropriate modification of the original reduction given in [2], that showed that finding a reactive organisation in a general reaction system is NP-hard.

Theorem 6. *Deciding if a flux-consistent network contains a non trivial organisation is NP-hard.*

4 Enumerating Chemical Organisations

Theorem 6 immediately implies that it is not possible to enumerate all organisations in a flux-consistent network in polynomial-time-delay in the size of the network.

We now observe that Theorem 5 indicates that for flux-consistent networks, the difficulty of finding non trivial organisations comes from the presence of cycles in the network. Indeed, as shown in Figure 3(b), cycles may interrupt the forward propagation if there exists a reaction that can produce a new metabolite

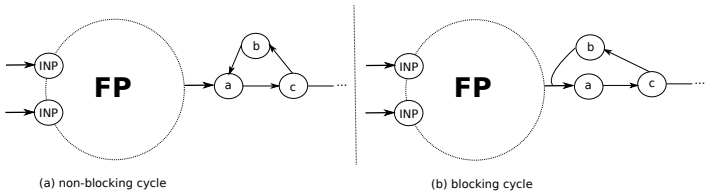

Fig. 3. (a) Non blocking cycle that will be traversed by the forward propagation procedure. (b) Forward propagation blocked by a unreached metabolite b.

a but needs for this a metabolite b which is not available and therefore blocks the reaction.

In order to find the reactive connected organisations, we need to process cycles every time the forward propagation procedure stops. This simple observation gives an upper bound of 2^k on the number of reactive connected organisations in flux-consistent networks, where k is the number of cycles in the network. In order to proceed we first define cycles formally. By a **cycle** in the metabolic network we mean a simple *directed* cycle in the underlying graph. Self-loops are also considered as cycles.

Definition 7. *A **hitting set** of a set of cycles is a set of metabolites such that each cycle contains at least one element of the hitting set.*

Theorem 8. *Let H be a hitting set of all the cycles of a directed hypergraph. The set of all reactive connected organisations, denoted as $\mathcal{O}$, is such that*

$$\mathcal{O} \subseteq \bigcup_{C \subseteq H} \{Cl_C\}$$

Proof. It is sufficient to show that if A is a reactive connected organisation then $A = Cl_C$, where $C = A \cap H$.

First observe that, since A is closed and C is a subset of its metabolites, it follows that $Cl_C \subseteq A$.

Let us suppose that A contains vertices which are not in Cl_C. We colour these vertices white and the vertices of Cl_C black. Consider any white metabolite a_1. Since A is an organisation, a_1 cannot be vanishing. Moreover, it is not an input of the network as otherwise it would be black. Therefore, there exists a reaction r fired by A that has a_1 as product and has a white substrate a_2, $a_2 \neq a_1$, otherwise a_1 would again be black by closure.

By iterating the above reasoning, it follows that the subgraph of the underlying directed graph induced by white vertices has minimum in-degree at least 1 and contains a directed cycle. This contradicts the fact that H hits all the cycles. The set of white vertices is therefore empty and $A = Cl_C$. □

The bound of the previous theorem is tight. Indeed, an example where the number of organisations reaches $2^{|H|}$ is given in Figure 4 where from the input, k metabolites are produced by k independent reactions and all of them are blocked

by cycles. Any combination of these independent paths can be de-blocked and produce a new organisation, and therefore we have 2^k organisations.

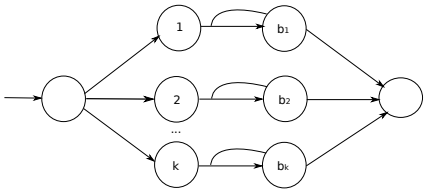

Fig. 4. Example with k parallel blocking cycles and 2^k organisations.

However, some cycles never interrupt the forward propagation procedure as illustrated in Figure 3(a). Other cycles, on the other hand, exhibit structural properties that may lead to a blocking situation. Such cycles are called **potentially blocking cycles**. A basic solution to find all organisations is to know how to unblock all cycles of the network independently of whether they are potentially blocking cycles or not. Therefore, instead of finding a hitting set for all cycles of the network, it is enough to break all potentially blocking cycles to compute all reactive connected organisations. In order to prove this, we first introduce a more formal definition of potentially blocking cycle.

Definition 9. *A **potentially blocking cycle** is a cycle such that there exists a reaction $r = (\{s_1, \ldots, s_h\}, \{p_1, \ldots, p_k\})$ in the network satisfying the following two conditions: (1) there exist i and j such that (s_i, p_j) is an edge of the cycle, and (2) there exists ℓ such that s_ℓ is not in the cycle.*

A potentially blocking cycle may or may not interrupt the forward propagation depending on the metabolites that were produced by the procedure once the cycle is reached. Figure 5(a) shows an example in which the cycle will be traversed, while Figure 5(b) shows an example in which it will block the forward propagation algorithm.

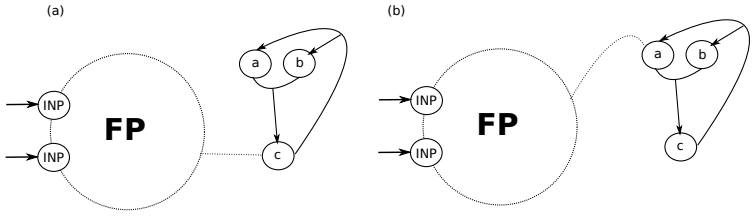

Fig. 5. Example of a potentially blocking cycle formed by the vertices a and c and reactions $a + b \to c$ and $c \to a + b$. This cycle will be traversed (a) or will block the forward propagation procedure (b) depending on how it is reached by procedure itself.

Theorem 10. *Let H be a hitting set of all the potentially blocking cycles of a directed hypergraph. The set of all reactive connected organisations, denoted as $\mathcal{O}$, is such that*

$$\mathcal{O} \subseteq \bigcup_{C \subseteq H} \{Cl_C\}$$

Proof. As in the proof of Theorem 8, it is sufficient to show that, given a reactive connected organisation A, $A = Cl_C$, where $C = A \cap H$ (in the following, all paths and cycles are meant directed in the underlying directed graph). Once again, it is easy to see that $Cl_C \subseteq A$. Let us then suppose that A contains vertices which are not in Cl_C and let us colour them white and those of Cl_C black.

Let a_i $(i = 1, 2, \ldots, k)$ be the set of white vertices such that there exists a reaction r_i having a_i as product and at least one black substrate. Then r_i has also at least one white substrate, which we denote by w_i, as otherwise a_i would be black by closure. Note that a white vertex does not have to belong to the set of the a_i's, but that this set is not empty as the organisation is connected and the set of white vertices is not empty.

If $w_i = a_i$, the selfloop induced by r_i is a potentially blocking cycle that contains no vertex of H, leading to a contradiction. Thus we may assume that w_i is distinct from a_i.

For $1 \leq i \leq k$, define T_i as follows: a vertex w is in T_i if either $w = w_i$ or there exists a white path starting from w and ending in w_i. Up to a reordering, we may assume that $|T_1| \leq |T_i|$ for $2 \leq i \leq k$. As T_1 is not empty, it has to contain at least one vertex that is a product of a reaction having a black substrate. If that vertex is a_1, there exists a path from a_1 to w_1, which yields a white cycle with the edge (w_1, a_1). That cycle is a potentially blocking cycle (because of the reaction r_1) which contains no vertex in H, leading to a contradiction.

In other words, T_1 contains a vertex among $(a_2, \ldots, a_k)$, say a_2. This implies that every path ending in w_2 can be extended to a path ending in w_1 and thus $T_2 \subseteq T_1$. Therefore, by minimality of T_1, $T_1 = T_2$.

This implies that $w_1 \in T_2$: hence, there exists a white path from w_1 to w_2. As $a_2 \in T_1$, there also exists a white path from a_2 to w_1. Thus, we can construct a white path from a_2 to w_2. Considering the edge (w_2, a_2), we again obtain a white cycle, which is potentially blocking (because of the reaction r_2) and contains no vertex in H, leading to a contradiction.

Thus, the set of white vertices is empty and $A \subseteq Cl_C$. □

Even in the case of the previous theorem, the bound is tight. Indeed, an example where the number of organisations is $2^{|H|}$ is, once again, given in Figure 4, since all cycles presented in the example are potentially blocking.

5 Hitting Set Approach to Enumerate Organisations

Two exact algorithms were proposed in [2] to enumerate organisations. The first one consisted in enumerating all closed sets and then checking for their self-maintenance, while the second one consisted in enumerating all self-maintaining

sets and then checking linear combinations of them in order to obtain closed sets. A third approach was also proposed that was based on the second one but avoided enumerating all self-maintaining sets. This algorithm however was a heuristic not guaranteed to find all self-maintaining sets (and thus organisations). Finally, a variation of the first algorithm was proposed in order to enumerate only reactive connected organisations. This algorithm comes closer to the one we describe later and works as follows. First, the forward propagation of the empty set is computed. Once the procedure is blocked, all possible combinations of metabolites that are connected to the produced set X are considered for addition, in order to obtain further closed sets that include X. At this point, the algorithm recursively continues.

Note that in the above procedures, no concept of blocking cycles has been formally identified and used. Now that we know that the hardness comes from such cycles, two different approaches can be applied in flux-consistent networks. One is to find a **global hitting set** for all cycles of the network and then, following Theorem 8, to apply the forward propagation procedure on each subset of the hitting set to produce closed sets which together form all candidate organisations and, finally, to check through LP if the candidates are self-maintaining. However, the problem of finding a minimum hitting set for all cycles of a directed graph is NP-hard as indeed it corresponds to the **feedback vertex sex (FVS)** problem [7]. Nevertheless approximation algorithms such as the one described in [13] can be used in order to perform this step.

A second possibility is to find a **local hitting set**. According to Theorem 10 only potentially blocking cycles need to be considered. This is a superset of the blocking cycles that can be identified when the forward propagation procedure stops because it is at this moment that we know we are dealing with actually blocking cycles. A more efficient algorithm to enumerate reactive connected organisations is thus the following one: apply the forward propagation algorithm and once blocked, identify the set B of metabolites that are blocking the closure and find a hitting set that unblocks only the cycles which directly or indirectly involve these blocking metabolites.

In [5], the authors presented an approximation algorithm to a generalisation of the FVS problem, called SUBSET-FVS, in which only a subset of the directed cycles in the graph is considered interesting, more specifically the ones that intersects a set of special vertices. In our case, the set of special vertices would be the blocking metabolites locally identified as described in the previous paragraph. The authors in [5] gave two approximation algorithms for the SUBSET-FVS problem. The first algorithm achieves an approximation factor of $O\left(\log^2 |B|\right)$. The second achieves an approximation factor of $O\left(\min\{\log T \log\log T, \log n \log\log n\}\right)$, where T is the value of the optimum fractional solution of the problem at hand, and n is the number of vertices in the graph.

Before proving that this idea can be correctly used to exactly solve our problem, we need to define the concept of a blocking cycle in relation to a given set C of metabolites.

Definition 11. *Let C be a set of metabolites. A C-**blocking cycle** is a cycle of vertices which are not in C such that there exists a reaction r in the cycle whose set of substrates contains at least one metabolite in C.*

C-blocking cycles correspond to those which actually stop the forward propagation procedure.

Theorem 12. *Let C be a closed set and H a hitting set of the C-blocking cycles of a metabolic network. Let A be a reactive connected organisation whose metabolites contain C. Then either $C = A$ or there exists a non empty subset B of H such that the closure of $C \cup B$ is still a subset of A.*

Proof. Let A be a reactive connected organisation containing C as a subset of its metabolites and let $B = A \cap H$. Let us suppose that $C \neq A$. To prove the theorem, it is then sufficient to prove that $B = A \cap H$ is not empty.

We colour the vertices of $A \setminus C$ in white and those of C in black. Since A is a reactive connected organisation, there exist edges between the white and the black metabolites and some of them go from a black to a white vertex, as otherwise, white vertices would be vanishing. Let $(a_1, \ldots, a_k)$ be the set of white vertices reached by at least one edge coming from a black vertex.

The same argument of the proof of Theorem 10 can now be applied, showing that, as the set of white vertices is not empty, it contains a white C-blocking cycle. Therefore, B is not empty. □

Corollary 13. *Every reactive connected organisation is included in the set $\mathcal{CO}$ returned by the procedure given in Algorithm CCO.*

Proof. Let A be a reactive connected organisation. It has to contain C_0 as every organisation contains the closure of the empty set. Let C be maximum among the elements of $\mathcal{CO}$ which are subsets of A. Then Theorem 12 implies, by maximality of C, that $A = C$. □

Algorithm CCO(G)

Require: a metabolic network represented as a hypergraph $G = (M, R)$;
Ensure: the set $\mathcal{CO}$ of all candidates for being organisations.

$\mathcal{CO} \leftarrow \{C_0\}$ where C_0 is the closure of the empty set ($Cl_{\{\}}$)
for all elements C in $\mathcal{CO}$ which have not been treated before **do**
  Compute a hitting set H of the C-blocking cycles
  for every $B \subset H$ **do**
    Compute $Cl_{C \cup B}$ and add it to $\mathcal{CO}$ if it was not present already
return $\mathcal{CO}$

Notice that the size of the hitting set computed by the algorithm is never greater than the number of blocking metabolites. Thus we can guarantee that our algorithm is theoretically better than existing algorithms which consider all blocking metabolites and then test all subsets. In the Appendix, we provide an example in which finding the hitting set gives a better result.

6 Conclusion

All the results presented in this paper correspond to enumerating closed sets as potential organisations. The problem of enumerating self-maintaining sets is still open. Such an approach could enable us to design a method that enumerates stoichiometrically valid precursor sets, which would be an important follow-up of the work on minimal precursors sets presented in [3]. Finally, the algorithms introduced in this paper do not take any specific advantage of the fact that the network should be mass-consistent and exploiting this might lead to better algorithms for enumerating self-maintaining sets.

References

1. V. Acuña, F. Chierichetti, V. Lacroix, A. Marchetti-Spaccamela, M.-F. Sagot, and L. Stougie. Modes and cuts in metabolic networks: Complexity and algorithms. *Biosystems*, 95(1):51–60, 2009.
2. F. Centler, C. Kaleta, P. S. di Fenizio, and P. Dittrich. Computing chemical organizations in biological networks. *Bioinformatics*, 24(14):1611–1618, 2008.
3. L. Cottret, P. V. Milreu, V. Acuña, A. Marchetti-Spaccamela, F. V. Martinez, M. F. Sagot, and L. Stougie. Enumerating Precursor Sets of Target Metabolites in a Metabolic Network. In *Workshop on Algorithms in Bioinformatics (WABI'2008)*, volume 5251 of *Lecture Notes in BioInformatics*, pages 233–244. Springer-Verlag Berlin, 2008.
4. P. Dittrich and P. S. di Fenizio. Chemical organisation theory. *Bull. Math. Biol.*, 69(4):1199–1231, 2007.
5. G. Even, J. Naor, B. Schieber, and M. Sudan. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica*, 20:151–174, 1998.
6. W. Fontana and L. Buss. "The Arrival of the fittest: towards a theory of biological organization. *Bull. Math. Biol.*, 56:1–64, 1994.
7. R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.
8. V. Lacroix, L. Cottret, P. Thbault, and M. F. Sagot. An Introduction to Metabolic Networks and Their Structural Analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(4):594–617, 2008.
9. J. A. Papin, J. Stelling, N. D. Price, S. Klamt, S. Schuster, and B. O. Palsson. Comparison of network-based pathway analysis methods. *Trends Biotechnol*, 22(8):400–405, 2004.
10. P. Romero and P. D. Karp. Nutrient-related analysis of pathway/genome databases. In *Proceedings of 6th Pacific Symposium on Biocomputing (PSB 2001)*, pages 470–482, 2001.
11. S. Schuster, T. Dandekar, and D. A. Fell. Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering. *Trends in Biotechnology*, 17(2):53–60, 1999.
12. S. Schuster and C. Hilgetag. On elementary flux modes in biochemical reaction systems at steady state. *J. Biol. Syst.*, (2):165–182, 1994.
13. P. D. Seymour. Packing directed circuits fractionally. *Combinatorica*, 15(2):281–288, 1995.

Appendix

Forward Propagation Algorithm

The forward propagation algorithm computes the closure of a given initial set of metabolites.

Algorithm FP(G, C)
Require: a metabolic network represented as a hypergraph $G = (M, R)$, a set $C \subset M$ of available metabolites;
Ensure: a set X of metabolites that can be reached from C.

$X \leftarrow C$
newMetabolite $\leftarrow$ TRUE
while newMetabolite = TRUE **do**
  for all $r \in \mathcal{R}_X$, $N \leftarrow \bigcup prod(r)$
  if $N \subseteq X$ **then**
    newMetabolite $\leftarrow$ FALSE
  else
    $X \leftarrow X \cup N$
return X

Proof of Theorem 4

Theorem 4. *If a network is flux-consistent then the whole network and the closure of the empty set are organisations.*

Proof. In order to proceed, we define $Sv[x]$ as the total amount of molecules x that are produced ($Sv[x] \geq 0$) or consumed (($Sv[x] < 0$) for the flux distribution v. We now present an algorithm to compute the flux vector that fulfills self-maintenance for the closure of the empty set.

Algorithm FluxForTrivialOrganisation(G)
Require: a metabolic network represented as an hypergraph $G = (M, R)$
Ensure: a flux vector v that fulfills the self-maintenance condition for the closure of the empty set.

$X \leftarrow FP(\{\})$
Apply DFS in the underlying graph and order the vertices by finishing timestamp
{ ft[v] is the finishing timestamp for a vertex v}
$v[r] \leftarrow 1$, for all $r \in \mathcal{R}_X$
while exists some molecule x with $Sv[x] < 0$ **do**
  $x \leftarrow$ molecule with $Sv[x] < 0$ and minimum finishing timestamp
  $RX \leftarrow$ reactions r in $\mathcal{R}_X$ with $x \in prod(r)$ and all substrates $s \in subs(r)$ has $ft[s] > ft[x]$
  $v[r] \leftarrow v[r] + (|Sv[x]|/|RX|)$ for every reaction $r \in RX$
return v

In a self-maintaining set nothing can vanish and all reactions that can be fired must be fired. The algorithm finds the closure of the empty set and names it X. We then initialise a flux vector v that fires all reactions in $\mathcal{R}_X$. Some of the compounds probably will be vanishing with this flux distribution. The loop

from lines 5-8 detects the molecules that are vanishing and increases the fluxes of some reactions that produce them in order to have a self-maintaining set. This is repeated until no molecule is vanishing anymore. The algorithm works because we choose the vanishing molecule to fix based on a previous DFS labelling. This allows us to pick a molecule y knowing that every other molecule that can be reached from y is not vanishing (has a finishing timestamp greater than the one of y). To produce y, we only consume molecules that have finishing timestamp greater than y. At each step, the new vanishing molecule must have a greater finishing time than the previous one and this guarantees that the algorithm will finish and produce a flux vector v in which no molecule vanishes. □

Proof of Theorem 6

Theorem 6. *Deciding if a flux-consistent network contains a non trivial organisation is NP-hard.*

Proof. We reduce our problem from 3-SAT. Given a boolean formula F in 3-CNF with n boolean variables and ℓ clauses, we construct a flux- and mass-consistent reaction network for which the existence of a non-trivial reactive (connected) organisation implies a positive answer to the 3-SAT and vice-versa.

In the reaction network we define the $2n$ literal metabolites $x_1, x_2, ..., x_n$, $\neg x_1, \neg x_2, ..., \neg x_n$ ($\neg x$ is the negation of x), clause metabolites $C_1, C_2, ..., C_\ell$ and additional metabolites $key, a, input$. The hyperarcs (reactions) are defined as:

- **Influx**: A reaction $\emptyset \to input$
- **Clause reactions**: For each clause $C_h = (A_1 \vee A_2 \vee A_3)$, we introduce 3 reactions: $key + A_j \to C_h + A_j,\ j = 1, 2, 3$.
- **Key reaction**: $C_1 + C_2 + ... + C_l + input \to (l + 1)\ key$.
- **Key outflux**: A reaction $key \to \emptyset$
- **Variable reactions of two types**: For each variable x_j, we introduce a type 1 reaction $x_j + \neg x_j \to a$, and a type 2 reaction $a + input \to x_j + \neg x_j + key$, $j = 1, \ldots, n$.

An illustration of this transformation is given in Figure 6. The transformation can obviously be done in polynomial time.

Let us first prove that the network is mass- and flux-consistent. To prove the latter, we construct a flux vector $v > 0$ such that $Sv = 0$ [1].

1. Setting the flux for each of the $2n$ variable reactions to 1, all literal metabolites and a are balanced, n of metabolite *input* are consumed and n of *key* are produced.
2. Next we set the flux through each of the 3ℓ *clause reactions* to $1/3$, producing 1 of each clause metabolite, and consuming ℓ of *key*.
3. These clause metabolites are balanced by setting the flux of the *key reaction* to 1, while consuming 1 of *input* and producing $\ell + 1$ units of *key*.
4. The last step is to balance the *input* and *key* metabolites, by setting the flux of the *influx* and *outflux* reactions both to $n + 1$.

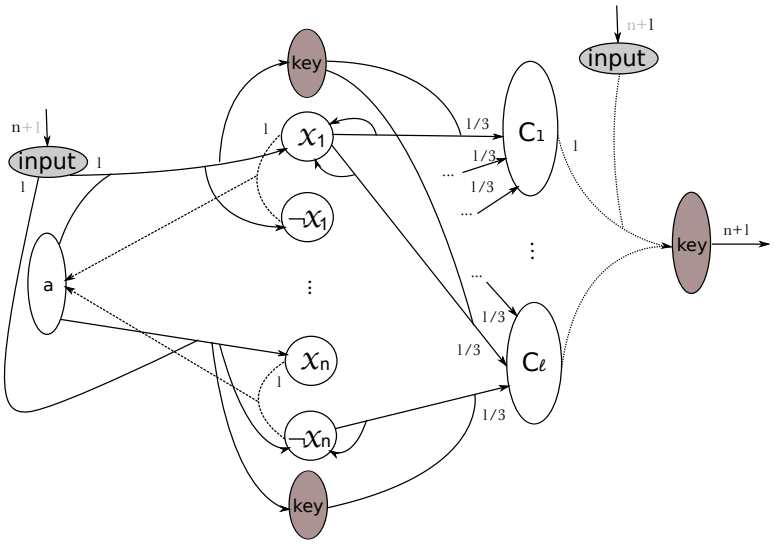

Fig. 6. Finding a non trivial reactive (connected) organisation is NP-hard. The *key* and *input* metabolites are presented more than once in the figure for convenience. Edge labels do not stand for stoichiometries but fluxes for a valid flux distribution that covers the whole network.

The balanced flux distribution v constructed has positive flux on every reaction of the network.

Mass-consistency of the network is easily verified by the mass vector that has mass 1 for all metabolites except for a, for which the mass is 2.

Suppose we have a satisfying truth assignment to F. Let K be the set of literals that are TRUE, then we claim that the set containing all metabolites corresponding to K, all clauses, *input* and *key* is a non-trivial organisation. Closedness is easily verified. We now show self-maintenance. Let k_i be the number of TRUE literals in C_i, $i = 1, \ldots, \ell$. For each TRUE literal in C_i we set a flux of $1/k_i$ on the corresponding *clause reaction*. Together with a flux of at least 1 in the *key reaction*, the *influx* and *outflux* secure self-maintenance.

Reversely, suppose we have a non-trivial reactive (connected) organisation O. First, notice that $\{input\}$ (the closure of $\emptyset$) is a trivial organisation and that it is part of every organisation, hence $input \in O$. As we now explain, this implies that if any organisation contains both x_i and $\neg x_i$ for any i, then it must contain the whole network, and hence be trivial. The presence of x_i and $\neg x_i$ triggers a *variable type 1 reaction*, producing a, which on its turn triggers all *variable type 2 reactions* producing all literal metabolites and *key*. All *clause reactions* are therefore triggered and henceforth the *key reaction* and the *outflux reaction*. Hence, O does not contain both literals corresponding to the same variable, and none of the variable reactions will have positive flux.

Since O is reactive, *input* must be the substrate of some active reaction, hence the *key reaction* must be active, implying that $key \in O$. Because of self-maintenance, all clause metabolites must be produced. To produce a clause metabolite at least one of its three *clause reactions* must be fired. Let K be the set of literal metabolites that are part of our reactive (connected) organisation. By closure, they are able to produce all clause metabolites, hence setting their value to TRUE yields a satisfying truth assignment for F. □

Local Hitting Set Approach

An illustration of the local hitting set approach is given in Figure 7. In this example, the forward propagation procedure produces the set C and is blocked by two metabolites, w_1 and w_2. The goal is to find a hitting set that intersects the blocking cycles for these metabolites. In this case, the hitting set has size 1 which is better than checking the combination of C with each of the blocking metabolites.

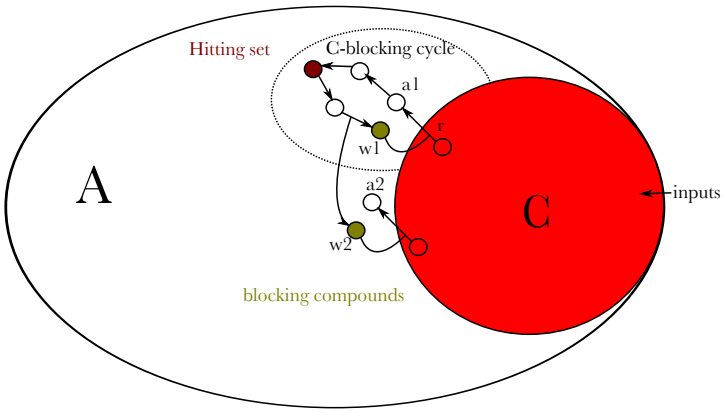

Fig. 7. Local hitting set approach.