Parameterizations of Hitting Set of Bundles and Inverse Scope

Peter Damaschke*
Department of Computer Science and Engineering
Chalmers University, 41296 Göteborg, Sweden
ptr@chalmers.se

Abstract

Hitting Set of Bundles generalizes the ordinary Hitting Set problem in the way that prescribed bundles of elements rather than single elements have to be put in a hitting set. The goal is to minimize the total number of distinct elements in the solution. First we prove that Hitting Set of Bundles, with the number of hyperedges and the solution size as parameter, is $W[1]$-complete. This contrasts to the to the corresponding parameterized Hitting Set version which is in FPT. Then we use this result to prove $W[i]$-hardness also for the Inverse Scope problem and some of its variants. This problem asks to identify small sets of chemical reactants being able to produce a given set of target compounds in a network of reactions. The problem has a graph-theoretic formulation as a reachability problem in directed graphs. On the positive side, we give an FPT algorithm where the parameter is the total number of compounds involved in the reactions.

Keywords: hitting set, parameterized complexity, W[i]-hardness, chemical reaction network

1 Introduction

Complex systems of chemical reactions can be described by directed graphs $G = (V, E)$ with two types of vertices: *compound vertices* in C and *reaction vertices* in R, where $V = C \cup R$. Every directed edge goes from V to R, or vice versa. Directed edges that enter a reaction vertex come from the *substrates* of the reaction, and edges that leave a reaction vertex go to the *products* of the reaction. If all substrates of a reaction are present, the reaction can take place and will generate all its products. Sometimes we abuse notation and identify a vertex with the compound or reaction it represents.

The *scope* of a set S of compounds is the set of all compounds that can be produced from S by a chain of reactions. For clarity we formulate this as an inductive definition: S is in the scope of itself, and if all substrates of a reaction are in the scope of S, then the products of the reaction are in the scope of S, too.

The Inverse Scope problem asks to find a minimum set of precursors such that a given set of target compounds can be produced from them. In the version called Inverse Scope with a Forbidden Set, the precursors must come from a prescribed subset $P \subset C$; the rest is "forbidden". (Formal problem specifications are presented later.)

*Tel. 0046-31-772-5405, Fax 0046-31-772-3663

Graphs as specified above are used to model, e.g., metabolite networks. Instead of graphs with two types of vertices one could also use hypergraphs of compound vertices, where reactions are described by directed hyperedges. The formulations are equivalent; here we stick to the graph notation.

A motivation for the Inverse Scope problem is to determine minimum sets of "seed compounds" that are needed to produce a desired set of compounds in a network of possible reactions. These can be, e.g., sets of nutrients needed by a living cell or organism to survive, or sets of raw materials needed to build a variety of chemical products in a factory. The problem whose parameterized complexity was studied by Ben-Zwi et al. (2011) is of similar spirit: A small subset of vertices in a graph shall be determined, that activates a certain number of other vertices. However, that problem deals with a usual undirected graph, and a vertex gets activated if some threshold number of neighbored vertices are activated.

Whereas earlier work (Angel et al. 2009; Nikoloski et al. 2008) is concerned with NP-hardness and approximability, in the present paper we are interested in the parameterized complexity. An algorithm that enumerates all minimal precursor sets of a target is given by Acuna et al. (2012), however without explicit complexity bounds (other than "exponential time").

We assume that the reader is familiar with the notion of *fixed-parameter tractable (FPT)* problems and $W[i]$-completeness, see (Downey and Fellows 1999; Niedermeier 2006). Briefly, a problem with input size n and an input parameter k is in FPT if some algorithm can solve it in $f(k) \cdot p(n)$ time, for some computable function f and some polynomial p. Instead of $f(k) \cdot p(n)$ we write $O^*(f(k))$, emphasizing the parameterized part of the time complexity and omitting the less critical polynomial part. There is a hierarchy of $W[i]$-complete problems ($i = 1, 2, \ldots$) that are unlikely to be in FPT. The well-known problems Clique and Hitting Set (defined below) are $W[1]$-complete and $W[2]$-complete, respectively, and $W[i]$-hardness can be proved by parameter preserving reductions from known $W[i]$-hard problems, in analogy to NP-hardness proofs in traditional complexity theory.

Contributions:

In Section 2 we prove that some generalization of Hitting Set called Hitting Set of Bundles is $W[1]$-complete even if the bundles are just pairs, and the number of hyperedges is the parameter. Thus, the (folklore) FPT result for Hitting Set with the same parameter is very unlikely to apply to this generalized problem. This fact should be of independent interest. In Section 3 we use it to prove $W[1]$-hardness of Inverse Scope with the target size as the parameter, even in acyclic directed graphs with certain degree limitations. Some related hardness results are shown as well. We remark that another Hitting Set generalization with double hits is in FPT, however some sophisticated dynamic programming method is needed for that; see (Damaschke 2009) for details about the problem and the algorithm.

After the hardness results we also provide a positive result: Section 4 describes an FPT algorithm taking the total number of compounds (including intermediate products) as the parameter. It works for reaction networks where every compound can be produced by only a limited number of alternative reactions. Section 5 lists some open problems.

2 Hitting Set of Bundles Parameterized

The following is a well-known fundamental problem. We formulate it in a non-standard way, such that the subsequent generalization is easier to understand.

HITTING SET: Given a hypergraph, i.e., a set of vertices equipped with a collection of c subsets called hyperedges. Choose one vertex from every hyperedge so as to minimize the number of distinct vertices among the c selected vertices.

A generalization of this problem was defined by Angel, Bampis, and Gourvès (2009): In the following, a *bundle* is just a subset of vertices. In the HITTING SET OF BUNDLES problem, hyperedges are sets (families) of bundles rather than sets of vertices.

HITTING SET OF BUNDLES: Given is a set of vertices, and a collection of hyperedges B_i, $i = 1, \ldots, c$, each being a set of bundles. Choose one bundle from every B_i so as to minimize the cardinality of the union of the c selected bundles.

Sometimes it is convenient to assign a *color* to every hyperedge B_i, and to say that the bundles in B_i have the assigned color. Note that a bundle may occur in several B_i, thus it may carry several colors.

Earlier we had already introduced the same problem under the name BOUNDED UNION (Damaschke 2006). The case when all bundles have size 1 is the ordinary HITTING SET problem, where the colors correspond to the hyperedges of the input hypergraph. Also recall the CLIQUE problem:

CLIQUE: Given an undirected graph and a number k, find a clique (complete subgraph) of k vertices.

Several parameters for HITTING SET OF BUNDLES come to mind very naturally:

c: size of the collection (number of colors, i.e., sets of bundles),
b: maximum number of bundles in a set,
a: maximum size of a bundle,
k: solution size, i.e., allowed size of the union of the selected bundles.

HITTING SET OF BUNDLES parameterized by (c, k) belongs to $W[1]$. This can be shown in the "Turing way", see (Cesati 2003) for the method and further references. A single-tape nondeterministic Turing machine can guess one bundle of each color, write the selected bundles on the tape, and then verify in $f(c, k)$ time, for some computable function f, that they contain at most k different vertices.

HITTING SET (the case $a = 1$) is well-known to be $W[2]$-complete in parameter k, however, HITTING SET parameterized in (c, k) allows for an FPT algorithm working with dynamic programming on subsets. See, e.g., (Niedermeier 2006) for an introduction. (We remark that it is sufficient to adopt only c as the parameter, since trivially $k \leq c$ can always be achieved.) One might hope that this algorithm can be generalized to HITTING SET OF BUNDLES, but this is unlikely due to our first theorem below. Going from $a = 1$ to $a = 2$ is a jump in complexity. First we establish an auxiliary problem that might also be useful for proving W[1]-hardness of other problems.

RAINBOW EDGE SET: Given an undirected graph $G = (V, E)$ where every edge is colored with one of c colors, select one edge of each color, so as to minimize the total number k of vertices incident to these c edges.

In the following we parameterize by the combined parameter (c, k); of course, we could also take either one of them, as they bound each other polynomially.

Lemma 1 *There is a parameterized reduction from* RAINBOW EDGE SET *to* HITTING SET OF BUNDLES *with* $a = 2$*, that preserves* (c, k)*.*

Proof. From an edge-colored graph $G = (V, E)$ we construct an instance of HITTING SET OF BUNDLES with vertex set V, where the bundles are the edges from E (thus $a = 2$), and the c hyperedges are simply the sets of edges of every single color. Equivalence of the instances is obvious from the problem definitions. $\diamond$

Lemma 2 *There is a parameterized reduction from* CLIQUE *to* RAINBOW EDGE SET *with* $k < c$*.*

Proof. Given a graph G (without loops) and an integer r, we construct a graph G' by replacing every vertex v of G with r copies $v_1, \ldots, v_r$. For every edge uv in G we create in G' all edges $u_i v_j$ where $i \neq j$. Let $k := r$ and $c := \binom{r}{2}$. We introduce a color for every unordered pair of indices $\{i, j\}$, and we paint every edge $u_i v_j$ in G' with the corresponding color $\{i, j\}$. Note that c is the number of colors.

Let C be any clique of size r in G. For every $v \in C$ we take some vertex v_i, thereby using every index i exactly once. Then all c edges between the selected vertices of G' exist, and they have c different colors.

Conversely, let C be any set of k vertices in G' such that the subgraph induced by C contains all c edge colors. Since $c := \binom{k}{2}$, this subgraph is a clique in G'. Since the copies of any one vertex of G are pairwise not adjacent, the vertices of C come from $k = r$ distinct vertices of G. Moreover, since edges in G' are always copies of edges in G, these r vertices of G form a clique.

Thus we have shown: G has a clique of size r if and only if G' has a subgraph with k vertices and c edges, one of each color. $\diamond$

Since CLIQUE is $W[1]$-complete, so is RAINBOW EDGE SET, due to Lemma 2. (A side remark is that the somewhat similar MULTICOLORED CLIQUE problem, where vertices rather than edges are colored, is known to be $W[1]$-complete, see (Fellows et al. 2010).) Together with the membership in $W[1]$ and Lemma 1 this implies:

Theorem 3 HITTING SET OF BUNDLES *with the combined parameter* (c, k) *is* $W[1]$*-complete even if* $a = 2$ *is constant, and* $k < c$*.* $\diamond$

The problem is more well-behaved in the other parameter b: HITTING SET OF BUNDLES with the combined parameter (b, k) is in FPT. As already observed in (Damaschke 2006), a trivial search tree algorithm solves it in $O^*(b^k)$ time, and if a is also part of the parameter, we have yet another option: In that case we can easily reduce the problem in $O^*(a^b)$ time to HITTING SET and subsequently run any parameterized HITTING SET algorithm for hypergraphs of rank b. The advantage is that base b in $O^*(b^k)$ can be improved to some base between $b - 1$ and b. (The *rank* of a hypergraph is the maximum size of its hyperedges.) We refer to (Niedermeier and Rossmanith 2003; Wahlström 2007; Fernau 2010a; Fernau 2010b) for these matters.

3 Inverse Scope Parameterized

This section is devoted to our central problem:

Inverse Scope: Given a directed graph $G = (V, E)$, $V = C \cup R$, and a target set $T \subseteq C$, find a minimum set $S \subseteq C$ such that T is in the scope of S.

Inverse Scope comprises several natural parameters: $t := |T|$ and $s := |S|$, and maximum indegrees and outdegrees of reaction and compound vertices, respectively. The latter parameters have obvious meanings: number of substrates and products of a reaction, number of alternative reactions that yield the same product, number of different reactions a substrate can be involved in.

Inverse Scope is NP-complete even in acyclic directed graphs (Nikoloski et al. 2008), hence the question arises which of these parameters and their combinations lead to FPT problems. Notice that $s < t$ can always be assumed, otherwise the problem is trivial since T is in the scope of $S := T$. We use some special terminology:

Definition 4 *A reaction with only one substrate (indegree 1) is a* fission. *A reaction with only one product (outdegree 1) is a* fusion. *A reaction with indegree and outdegree 1 is a* transformation.

In the following we introduce some possible modifications of input graphs that we summarize under the name *tree insertions*. They serve as gadgets, with the purpose to reduce the degree of networks in some parameterized complexity results. However we will have to apply them carefully, depending on the problem.

Tree insertions:

Consider a reaction vertex v with outdegree larger than 2. We insert two compound vertices representing dummy products of v which in turn generate the actual products by two fission reactions. Doing this recursively we can replace the product side of v with a tree of fission reactions, all with outdegree 2. Similarly, for a reaction vertex v with indegree larger than 2, we can replace the substrate side of v with a tree of fusion reactions, all with indegree 2.

Consider a compound vertex v with outdegree larger than 2. Similarly as above, instead of connecting v directly to many reactions with substrate v, we can insert a tree of dummy fission reactions with outdegree 2, that just produce compounds that are identical to the compound v but have different names in the graph (like "brands"). Finally, for a compound vertex v with indegree larger than 2, we can replace the substrate side of v with a tree of compound vertices of indegree 2 where every edge is subdivided by a transformation vertex.

First of all, there is a simple reduction from Hitting Set to Inverse Scope that shows hardness in the parameter s.

Theorem 5 Inverse Scope *with parameter s is $W[2]$-hard even in acyclic directed graphs.*

Proof. Consider any Hitting Set instance, that is, a hypergraph. We represent every hyperedge by a compound vertex in T, and every hypergraph vertex v by a compound vertex which is the only substrate of a reaction that produces the compounds of all hyperedges containing v.

Obviously, any hitting set of size s yields a way to produce the entire T from s compounds. Conversely, if T can be obtained from a set S of s compounds, we can always assume that S and T are disjoint: Since we have fission reactions only, every compound in T could also be produced from some compound not being in T. Thus, our hypergraph has a hitting set of size s. $\diamond$

Recall that $s < t$ can always be assumed in Inverse Scope. Thus Theorem 5 does not yet exclude the possibility of an FPT algorithm with parameter t. In fact, t in the above reduction corresponds to the number of hyperedges, and Hitting Set is an FPT problem in this parameter. However, Inverse Scope is also expressive enough to encode bundles, and hence Theorem 3 dashes our hopes:

Theorem 6 Inverse Scope *with parameter t is $W[1]$-hard even in acyclic directed graphs, where $s < \sqrt{t}$, all vertices have indegree at most 2, all compound vertices have outdegree at most 2, and all reaction vertices have outdegree at most t.*

Proof. We establish a parameterized reduction from Hitting Set of Bundles with parameter (c, k), where $k < c$, and $a = 2$ (using Theorem 3).

Given an instance H of Hitting Set of Bundles with c colors, we construct a directed graph as follows. Let $t = c^2$. The target set T is divided into c groups of c vertices, one group for each color, thus we call them *color groups*. Further compound vertices represent the elements of H. Every bundle in H is the set of substrates of a reaction that produces the color group of the color of the bundle. We set $s := k$.

An important point in this construction is that having vertices from T in the solution S is never beneficial: Since $s = k < c$, no color group is entirely in S. Hence in each color group some compound must be produced by reactions from other compounds in S which are not in that color group. But since all vertices in a color group are undistinguishable in the graph, it follows that we can produce the whole color group from the same compounds, thus no vertex from the color group is needed in S.

With that observation in mind, the equivalence of the problems is easy to establish: In order to produce T from $s < c$ substances we must first produce all color groups, thus we need in S a bundle of each color. Conversely, if such bundles are contained in S, clearly we can produce T. Also, the graph is acyclic. It remains to limit the degrees.

Since an element can belong to arbitrarily many bundles, the outdegrees of compound vertices that represent elements are arbitrary, but we can get rid of large outdegrees by tree insertion. We only have to make sure that having dummy vertices from the inserted trees in S is not beneficial. However this is obvious, since any such dummy compound can be produced from the real compound, and no two real compounds can produce the same dummy compounds. Similarly, a compound in a color group can be produced by one of arbitrarily many "bundle" reactions, but we can limit the indegree to 2 by tree insertion. Using the same arguments as above, dummy vertices in S cannot yield a smaller solution: Recall that all c vertices of a color group must be produced, and all of them have distinct dummy vertices as predecessors. Now it is easy to check that all claimed degree bounds hold. In particular, the indegree 2 of reaction vertices comes from $a = 2$. $\diamond$

We remark that a similar hardness result can be obtained for the range $s < t$, however with weaker degree bounds: Just do the previous reduction and insert a final reaction vertex having all c^2 color group vertices as substrates, and a new target set T of size c as its set of products. Next we consider:

Inverse Scope with a Forbidden Set: Given a directed graph $G = (V, E)$, $V = C \cup R$, a set $P \subset C$ of possible precursors, and a target set $T \subseteq C$, find a minimum set $S \subseteq P$ such that T is in the scope of S.

In this problem version, also instances with $s \geq t$ can be nontrivial, as opposed to Inverse Scope. Even worse, setting parameter t to the constant $t = 1$ does not make the problem easier: Consider any instance with a target set T. Now insert a further reaction vertex with T as the set of substrates, and just one new target product. The target product is not added to the set P of permitted precursors. Then, clearly, the target product is in the scope of a solution $S \subseteq P$ if and only if T is in the scope.

Due to the extra demand $S \subseteq P$ we obtain a hardness result for more restricted instances:

Theorem 7 Inverse Scope with a Forbidden Set *with parameter s is $W[2]$-hard even in acyclic directed graphs, where $t = 1$ is constant, and all indegrees and outdegrees are at most* 2.

Proof. It suffices to put things together: We start with the reduction from Hitting Set as in Theorem 5, and we define P to be the set of compound vertices representing the elements of the given hypergraph. We also add a final reaction vertex as discussed above, that produces only one target compound. Since the problem requires $S \subseteq P$, no precautions are needed to avoid unwanted vertices in S, therefore we can also reduce all degrees to at most 2 by tree insertions. $\diamond$

4 Parameterization by the Total Number of Compounds

Due to the hardness of Inverse Scope with a Forbidden Set (even for a single target) in the number of precursors used in the reactions, it is sensible to look for alternative parameterizations that enable FPT algorithms. The total number of compounds needed to produce a target set is certainly an interesting parameter, because a compound whose production depends on too many intermediate steps is harder to reach, and the process is more prone to disturbances.

Therefore we consider as a parameter the total number q of compounds that are needed to produce a target set T from precursors. We have to precisely define what is counted in q: When T is produced from precursors by using some subset Q of reactions in R, we define q as the number of compounds that are not in T but appear as substrates in the reactions in Q. Note that we do not count *side products*, i.e., products of reactions in Q which are not needed as substrates of other reactions in Q. (Alternatively we may also choose to count side products, but an FPT result is stronger with a more restrictive parameter.) Clearly, q can be much smaller than the number of all existing compounds in the network.

The reduction from Hitting Set used earlier also shows that Inverse Scope with a Forbidden Set remains $W[2]$-hard in the combined parameter (t, q), since q takes on the role of s. The indegrees of target vertices in the reduction graph are arbitrary. Therefore it is nice to notice that just limiting the indegrees of compound vertices by some integer b makes the problem FPT. (We stress that no other degree limitations are needed.) This is not too artificial an assumption, since in real reaction networks there are not so many alternative reactions that yield the same compound.

At first glance, the announced result might appear to be a straightforward generalization of the branching algorithm for Hitting Set on hypergraphs of rank b: As long as some

compound remains to be produced, we have to choose some of the b possible reactions, and this adds to the necessary compounds, i.e., reduces the parameter, or the reaction is available anyhow. While this is, in fact, the basic idea, things are no longer that simple for our problem, and we have to argue very carefully. One new feature is that the reaction graph may have directed cycles.

Theorem 8 Inverse Scope with a Forbidden Set *is solvable in* $O^*(b^{2q+t})$ *time when all compound vertices have indegree at most* b*. Moreover, we can enumerate, within that time, all solutions that require at most* q *non-target compounds as substrates in their reactions.*

Proof. Our algorithm follows the bounded search tree paradigm. We presume that the reader is familiar with the notion of branching rule. We simply say *rule* for brevity.

Every node of the search tree represents a set X of compound vertices. A vertex $v \in X$ gets *marked* if the compound has been produced (either v is in the precursor set P or v has been generated from precursors). At the root of the search tree we put $X := T$, where all vertices are initially unmarked.

Some special terminology will be convenient. A reaction is said to *produce* a compound v, if v is among the products of this reaction. With respect to the set X we call a reaction *internal* if all its substrates are in X, and *external* otherwise.

Algorithm description:

At any node of the search tree we perform the following rules (0)–(4) exhaustively in the given order, that is, we always apply the first possible rule.

(0) If all vertices of X are marked, then we return X as a possible solution.

(1) If some internal reaction has only marked substrates and produces a vertex $v \in X$ that is yet unmarked, then we mark v.

(2) If some unmarked $v \in X$ is produced by external reactions only, then we branch as follows. In every branch we decide on one such reaction r, and we add the missing substrates of r to X. Every new compound in X that comes from P is marked.

(3) If some unmarked $v \in X$ is produced by both external and internal reactions, then we branch as follows. We create a branch for each external reaction producing v, as in rule (2). Furthermore we create one branch where we decide not to apply any of the external reactions that produce v. We disable these reactions. (That is, disabled reactions cannot be inserted anymore in the solution later on. However, we do not decide at this point which of the internal reactions will produce v, if there are several.)

(4) If all unmarked $v \in X$ are produced by internal reactions only, then we stop processing X, that is, X is a dead end.

We remark that a special case of rule (2) appears when v is not produced by any reaction. In that case, no r exists. Thus, branching means that we stop processing X altogether, since obviously no solution may include X.

We abort a path in the search tree as soon as $|X| > q + t$ or the path length exceeds $2q + t$ (as will be motivated below). In the end we take a solution of minimum cardinality or collect all solutions.

Correctness:

Since we mark only vertices from P and vertices produced by internal reactions from already marked substrates, an obvious inductive argument shows that all marked compounds can be really produced by using only compounds in X. Hence rule (0) returns only valid solutions.

Rule (1) is correct since it only produces those compounds that can be obtained without extending X.

We show correctness of the branching rules (2)–(3); note that correctness means that any solution has to choose one of the branches.

When (0)–(1) are not applicable, clearly there exist unmarked vertices in X. Since X never gets smaller, these unmarked compounds must be produced at some time. A trivial fact is that, for each unmarked $v \in X$, some of the reactions producing v must be applied. This immediately implies correctness of rule (2) and also rule (3).

Finally, suppose that we arrive at rule (4). It remains to show that X cannot be extended to any solution.

Consider any unmarked $v_0 \in X$. As said above, we must apply some reaction r_0 producing v_0. Since r_0 is internal, all substrates of r_0 are in X. Some substrate v_1 of r_0 is unmarked, since otherwise we could apply rule (1). Since $v_1 \in X$, we can now iterate the argument. This yields an arbitrarily long sequence $v_0, r_0, v_1, r_1, v_2, r_2, \ldots$ of compounds and reactions such that each v_{i+1} must be marked before v_i. This contradicts the finiteness of $|X|$.

Time analysis:

Note that all branching rules have branching number at most b. We show that all solutions with at most q non-target compounds are found at depth at most $2q + t$ in the tree, and that the calculations in each node of the search tree need polynomial time. Together this implies the claimed running time.

To this end, we *charge* a vertex v for adding v to X in rule (2) or (3), and for disabling the external reactions that produce v in rule (3) (but only in the branch where disabling is really done). Since X only grows, every $v \in X$ is charged only once for entering X, and only once for disabling: As soon as all reactions producing v are internal, they will remain internal. This shows that path length $2q + t$ is sufficient.

It is evident that rules (0) and (2)–(4) can be implemented to run in polynomial time. In rule (1) we can repeatedly go through all internal reactions until no new vertices get marked. Then we can leave (1), and since some new marks appeared in every round, the time is polynomial. $\diamond$

Theorem 8 yields that the problem is in FPT in the combined parameter (q, t). It remains open if parameter q would be enough. At least we can get such a result in the acyclic case.

Theorem 9 Inverse Scope with a Forbidden Set *is solvable in $O^*(b^q)$ time when the graph is acyclic, and all compound vertices have indegree at most b. Moreover, we can enumerate all solutions within that time.*

Proof. The algorithm is pretty much the same as for general graphs, but with a simplified rule (3).

(3') If some unmarked $v \in X$ is produced by some internal reactions, then we delete v from the products of all external reactions currently producing v.

That is, we commit to produce v using some internal reaction (but the external reactions can still be used later on for other products). Since no branching is done in (3'), and in each application of the only branching rule (2) some vertex is added to X, we do not need an auxiliary parameter as in Theorem 8, hence path length q is sufficient. Also, rule (4) can be canceled.

It remains to show the correctness of (3'). We have to ensure that, if X can be extended to a solution, then all substrates of some *internal* reaction producing v will be available at some time. (This step of the reasoning does not work for arbitrary directed graphs because, informally speaking, there can be cycles of dependencies being unreachable from the precursors.)

Assume for contradiction that some $v_0 \in X$ is yet unmarked upon termination. Then any internal reaction r_0 producing v_0 has an unmarked substrate $v_1 \in X$, otherwise we could apply rule (1). As earlier we inductively obtain a sequence $v_0, r_0, v_1, r_1, v_2, r_2, \ldots$ of internal reactions and unmarked nodes. Since the graph is acyclic and X is finite, this chain must end. Thus, some v_j is unmarked but no internal reactions produce v_j. But then we could apply (2) – a contradiction, or v_j cannot be produced at all (but has been added to X by rule (2) at some time). Hence it is never a mistake to apply the new rule (3'). $\diamond$

5 Conclusions

To our best knowledge this is the first study of the parameterized complexity of Hitting Set of Bundles and Inverse Scope. We mention some of the open problems:

- Get results for other parameterizations and ranges of parameters.
- Where in $W[i]$ hierarchy is the Inverse Scope problem under various parameterizations?
- Could Inverse Scope parameterized by s be $W[i]$-complete for some $i > 2$? Note that only few such combinatorial problems are known.
- Is there an FPT algorithm for Inverse Scope with a Forbidden Set in parameter q only (for any fixed degree b)?
- Can base b in the time bounds in Section 4 be improved (as it has been achieved for Hitting Set in hypergraphs of rank b)?
- What is the parameterized complexity of Inverse Scope with Two Forbidden Sets? In this problem that was also suggested by Nikoloski et al. (2008), a given set of unwanted side products must be avoided by the reactions taking part in a solution.
- Networks of chemical reactions can also be modeled as Petri nets, with the additional feature that also the quantities of substances are represented as tokens on the vertices. This gives rise to a host of "quantitative" problems extending the ones studies here. What are their complexities with respect to various parameterizations?

It may also be interesting to apply the algorithms from Section 4 to real (e.g., biological) reaction networks and see if the results give new insights.

Acknowledgment

This work has been supported by the Swedish Research Council (Vetenskapsrådet), through the project grant 2010-4661, "Generalized and fast search strategies for parameterized problems". The author would like to thank the referees for valuable comments that made the presentation clearer, and for suggesting the Turing way to show $W[1]$ membership.

References

[1] Acuna V, Vieira Milreu P, Cottret L, Marchetti-Spaccamela A, Stougie L, Sagot MF (2012) Algorithms and complexity of enumerating minimal precursor sets in genome-wide metabolic networks. Bioinf 28: 2474–2483

[2] Angel E, Bampis E, Gourvès L (2009) On the minimum hitting set of bundles problem. Theor Comp Sci 410:4534–4542

[3] Ben-Zwi O, Hermelin D, Lokshtanov D, Newman I (2011) Treewidth governs the complexity of target set selection. Discr Optim. 8:87–96

[4] Cesati M (2003) The Turing way to parameterized complexity. J. Comp Syst Sci 67:654–685

[5] Damaschke P (2006) Parameterized enumeration, transversals, and imperfect phylogeny reconstruction. Theor Comp Sci 351:337–350

[6] Damaschke P (2009) Multiple hypernode hitting sets and smallest two-cores with targets. J Comb Optim 18:294–306

[7] Downey RG, Fellows MR (1999) Parameterized complexity. Springer

[8] Fellows MR, Flum J, Hermelin D, Müller M, Rosamond FA (2010) W-hierarchies defined by symmetric gates. Theor Comp Syst 46:311–339

[9] Fernau H (2010a) Parameterized algorithmics for d-hitting set. Int J Comput Math 87:3157–3174

[10] Fernau H (2010b) A top-down approach to search-trees: Improved algorithmics for 3-hitting set. Algorithmica 57:97–118

[11] Niedermeier R (2006) Invitation to fixed-parameter algorithms. Oxford Lecture Series in Math and its Appl, Oxford Univ Press

[12] Niedermeier R, Rossmanith P (2003): An efficient fixed-parameter algorithm for 3-hitting set. J Discr Algor 1:89–102

[13] Nikoloski Z, Grimbs S, Selbig J, Ebenhöh O (2008) Hardness and approximability of the inverse scope problem. In: Crandall KA, Lagergren J (eds) WABI 2008, LNCS 5251, Springer, Heidelberg, pp. 99–112

[14] Wahlström M (2007) Algorithms, measures, and upper bounds for satisfiability and related problems, Dissertation 1079, Linköping Studies in Science and Technology