The complexity of acyclic subhypergraph problems

David Duris and Yann Strozecki

Équipe de Logique Mathématique (FRE 3233) - Université Paris Diderot-Paris 7
{duris,strozecki}@logique.jussieu.fr

Abstract. We investigate the computational complexity of two decision problems concerning the existence of certain acyclic subhypergraphs of a given hypergraph, namely the Spanning Acyclic Subhypergraph problem and the Maximal Acyclic Subhypergraph problem. The former is about the existence of an acyclic subhypergraph such that each vertex of the input hypergraph is contained in at least one hyperedge of the subhypergraph. The latter is about the existence of an acyclic subhypergraph with k hyperedges where k is part of the input. For each of these problems, we consider different notions of acyclicity of hypergraphs: Berge-acyclicity, γ-acyclicity, β-acyclicity and α-acyclicity. We are also concerned with the size of the hyperedges of the input hypergraph. Depending on these two parameters (notion of acyclicity and size of the hyperedges), we try to determine which instances of the two problems are in $\mathsf{P} \cap \mathsf{RNC}$ and which are NP-complete.

1 Introduction

A spanning tree of a graph G is a connected and acyclic subgraph H of G such that every vertex of G belongs to at least one edge of H. When G is connected, it always has a spanning tree. Various efficient algorithms have been devised to find a spanning tree, since it is an important object both for theoritical reasons (base in the cycle matroid) and practical applications (spanning tree protocol in networks). For instance, by means of a depth-first search, one can build a spanning tree in time linear in the size of the graph. One may also find it in probabilistic logarithmic parallel time (cf. [9]). If we change the spanning condition into a size condition, that is to say deciding if a graph has an acyclic subgraph of a given size, the problem is trivial.

The generalization of these two problems to hypergraphs make them much more interesting. First because there exist various notions of acyclicity for hypergraphs, and because the complexity of these problems depends also on the sizes of the hyperedges. The notions of acyclicity we consider here are (in increasing order of generality): Berge-acyclicity (cf. [2]), γ-acyclicty (cf. [5]), β-acyclicity (cf. [3]) and α-acyclicity (cf. [1]). They arise from hypergraph theory and database theory but they are studied in various areas such as combinatorics (cf. [16]) or logic (cf. [4]).

There are a few results on the complexity of the two problems Spanning Acyclic Subhypergraph and Maximal Acyclic Subhypergraph. Lovász has proved that, for Berge-acyclicity and 3-uniform hypergraphs, Maximal Acyclic Subhypergraph is computable in polynomial time thanks to an adaptation of its matching algorithm in linear polymatroids (cf. [12]). In [13], they also consider Berge-acyclicity and they show that Spanning Acyclic Subhypergraph on k-uniform hypergraphs is in RP for $k = 3$ and NP-complete if $k \geq 4$. In [10], they consider Spanning Acyclic Subhypergraph and Maximal Acyclic Subhypergraph for α-acyclicity and they show that they are NP-complete. But they have no condition on the size of the hyperedges of the input hypergraph. They even notice that they do not know the complexity of Maximal Acyclic Subhypergraph on hypergraphs where there is no hyperedge contained in another (so, in particular, when the input hypergraph is k-uniform for any k).

In this paper, we first show that Spanning Acyclic Subhypergraph is NP-complete even when $k = 3$ for γ, β and α-acyclicity. We also give an algorithm solving Maximal Acyclic Subhypergraph for 3-uniform hypergraphs and Berge-acyclicity. This algorithm is in probabilistic polylogarithmic parallel time, it uses algebraic techniques and a polynomial from [14]. Then we prove the NP-completeness of Maximal Acyclic Subhypergraph when $k \geq 4$ for Berge acyclicity and when $k \geq 3$ for the other notions of acyclicity. These results show once again that Berge-acyclicity is quite different from γ, β and α-acyclicity.

2 Definitions

A *hypergraph* is a couple $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is a finite set and $\mathcal{E}$ is a set of nonempty subsets of $\mathcal{V}$. This is a generalization of the notion of graph since the hyperedges can have any size instead of size 2. A *subhypergraph* $\mathcal{S} = (\mathcal{V}', \mathcal{E}')$ of $\mathcal{H}$ is a hypergraph such that $\mathcal{V}' \subseteq \mathcal{V}$ and $\mathcal{E}' \subseteq \mathcal{E}$.

A hypergraph is *Berge-acyclic* if it contains no Berge-cycle. A *Berge-cycle* is a sequence $(E_1, x_1, ..., E_n, x_n)$ with $n \geq 2$ such that :

- the E_i are distinct hyperedges,
- the x_i are distinct vertices and
- for every $i \in [1, n-1]$, x_i belongs to E_i and E_{i+1}.
- x_n belongs to E_n and E_1.

A hypergraph is γ-*acyclic* if it contains no γ-cycle. A γ-*cycle* is a Berge-cycle $(E_1, x_1, ..., E_n, x_n)$ with $n \geq 3$ such that, for every i in $[1, n-1]$, x_i belongs no other E_j than E_i and E_{i+1} (but x_n belongs possibly to other E_j).

Equivalently, a hypergraph is γ-acyclic iff one obtains a hypergraph with no hyperedge after applying successively the following four rules (see [5] for a proof):

1. If a vertex is isolated (i.e. it belongs to precisely one hyperedge), then remove that vertex from $\mathcal{V}$ and from the hyperedge that contains it.

2. If a hyperedge has one or zero element, then remove that hyperedge from $\mathcal{E}$.
3. If two hyperedges contain precisely the same vertices, then remove one of those hyperedges from $\mathcal{E}$.
4. If two vertices belong to precisely the same hyperedges, then remove one of those vertices from $\mathcal{V}$ and from every hyperedge that contains it.

A hypergraph is *β-acyclic* if it contains no β-cycle. A *β-cycle* is a γ-cycle $(E_1, x_1, ..., E_n, x_n)$ such that x_n belongs to no other E_j than E_n and E_1.

A hypergraph is *α-acyclic* if it has a *join tree*, i.e. a tree T whose vertices are the hyperedges of $\mathcal{H}$ and such that, for every vertex v of $\mathcal{H}$, the subgraph of T induced by the vertices of T containing v is connected.

For every hypergraph, we have the following implications: Berge-acyclic $\Rightarrow$ γ-acyclic $\Rightarrow$ β-acyclic $\Rightarrow$ α-acyclic.

We are also interested in the size of the hyperedges. A hypergraph is *k-uniform* if all of its hyperedges have size k. In this paper, we consider the two following decision problems for all notions of acyclicity and we restrict them to k-uniform hypergraphs for $k = 3$ and $k \geq 4$.

Spanning Acyclic Subhypergraph

Input: a hypergraph $\mathcal{H}$

Output: is there a spanning acyclic subhypergraph of $\mathcal{H}$, i.e. an acyclic subhypergraph of $\mathcal{H}$ such that each vertex of $\mathcal{H}$ is contained in at least one hyperedge of the subhypergraph?

Maximal Acyclic Subhypergraph

Input: a hypergraph $\mathcal{H}$ and an integer n

Output: is there an acyclic subhypergraph of $\mathcal{H}$ of size n, i.e. with n hyperegdes?

3 Spanning acyclic subhypergraphs

In [13], the problem of finding a spanning hypertree for Berge-acyclicity on k-uniform hypergraphs is considered. It is a variation of Spanning Acyclic Subhypergraph, where the subhypergraph must be connected. They show that this problem admits an RP algorithm for $k = 3$ and is NP-complete if $k \geq 4$. A probabilistic algorithm using the same idea is given in the next section and solves both Spanning Acyclic Subhypergraph and Maximal Acyclic Subhypergraph for Berge-acyclicity on 3-uniform hypergraphs. We consider here the other ayclicity notions and prove that this problem is NP-complete even when $k = 3$.

3.1 NP-completeness for γ, β and α-acyclicity

Proposition 1. *For the three notions γ, β and α-acyclicity and 3-uniform hypergraphs,* Spanning Acyclic Subhypergraph *is NP-complete.*

Proof. We give a polynomial time reduction from Sat to Spanning Acyclic Subhypergraph. Let f be a propositional formula under conjunctive normal form (instance of the Sat problem). We have $f = \wedge_{i\in[1,n]} C_i$ where, for every i, C_i is a clause $\vee_{j\in[1,n_i]} B_{i,j}$ with $B_{i,j}$ equal to a variable or its negation. We define a 3-uniform hypergraph $\mathcal{H}_f = (V_f, \mathcal{E}_f)$ such that f is satisfiable if and only if $\mathcal{H}_f$ has a θ-acyclic spanning subhypergraph (for $\theta = \gamma$, β and α). For each variable A, V_f contains the vertices $0_{A,a}$, $0_{A,b}$, $1_{A,a}$, $1_{A,b}$, $r_{A,a}$ and $r_{A,b}$. For each clause C, V_f contains the vertex p_C. For each variable A, $\mathcal{E}_f$ contains the hyperedges $\{0_{A,a}, r_{A,a}, 1_{A,a}\}$ and $\{0_{A,b}, r_{A,b}, 1_{A,b}\}$. Moreover, for each clause C and each variable A in C, $\mathcal{E}_f$ contains :

- the hyperedge $\{0_{A,a}, 0_{A,b}, p_C\}$ if $\neg A$ appears in C and
- the hyperegde $\{1_{A,a}, 1_{A,b}, p_C\}$ if A appears positively in C.

Figure 1 shows an example of this construction.

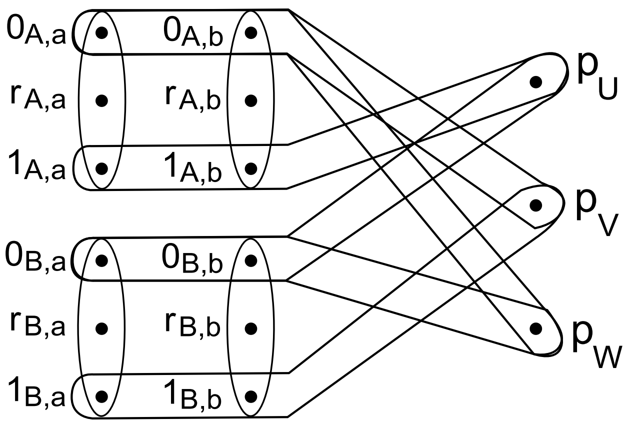

Fig. 1. The hypergraph $\mathcal{H}_f$ associated to the propositional formula $f := U \wedge V \wedge W$ with $U := A \vee \neg B$, $V := \neg A \vee B$ and $W := \neg A \vee \neg B$.

Let $\theta \in \{\gamma, \beta, \alpha\}$. For every variable A, a spanning subhypergraph contains necessarily the hyperedges $\{0_{A,a}, r_{A,a}, 1_{A,a}\}$ and $\{0_{A,b}, r_{A,b}, 1_{A,b}\}$ (so that they cover the vertices $r_{A,a}$ and $r_{A,b}$). In order to be θ-acyclic, a spanning subhypergraph can only contain for every A, hyperedges of the form $\{0_{A,a}, 0_{A,b}, p_C\}$ or $\{1_{A,a}, 1_{A,b}, p_C\}$. In other terms, we have to choose an assignment 0 or 1 for each variable A. In order to be spanning, the subhypergraph must also contain at least one hyperedge $\{\varepsilon_{A,a}, \varepsilon_{A,b}, p_C\}$ (with $\varepsilon \in \{0, 1\}$) for each clause C, which means that the assignment must satisfy every clause. Moreover, in order not to contain a cycle, we can keep, for each clause C, only one hyperedge containing p_C. Thus, there exists a θ-acyclic spanning hypergraph if and only if f is satisfiable.

We easily check that this reasoning works for every θ in $\{\gamma, \beta, \alpha\}$. However, we notice that, in most cases, a spanning subhypergraph corresponding to an assignment f making f true is Berge-cyclic. For instance, if it contains the hyperedges $\{0_{A,a}, 0_{A,b}, p_C\}$ and $\{0_{A,a}, 0_{A,b}, p_D\}$, then it contains the Berge-cycle $(\{0_{A,a}, 0_{A,b}, p_C\}, 0_{A,a}, \{0_{A,a}, 0_{A,b}, p_D\}, 0_{A,b})$. $\square$

4 Maximal acyclic subhypergraphs

The next two subsections show in particular that the Maximal Acyclic Subhypergraph problem is in RP when we consider Berge-acyclicity and when the input hypergraph is 3-uniform. In Subsection 4.3, we show that this problem is NP-complete for every acyclicity notion on 4-uniform hypergraphs and for γ, β and α-acyclicity on 3-uniform hypergraphs.

4.1 A method to determine the degree of polynomial relatively to a subset of its variables

From now on, we consider polynomials with n variables, denoted by $X_1, \ldots, X_n$, and rational coefficients. A sequence of n positive integers $\boldsymbol{e} = (e_1, \ldots, e_n)$ characterizes the term $\boldsymbol{X}^{\boldsymbol{e}} = X_1^{e_1} X_2^{e_2} \ldots X_n^{e_n}$. The total degree of a monomial is the sum of the degrees of its variables and the total degree of a polynomial is the maximum of the total degrees of its monomials.

Definition 1 (Degree of a polynomial with regard to a set of indices). *Let n be an integer and $S \subseteq [1, n]$. The degree of the term $\boldsymbol{X}^{\boldsymbol{e}}$ with regard to S is the sum of the e_i such that $i \in S$. The degree of an n variable polynomials with regard to S is the maximum of the total degrees of its monomials with regard to S, we denote it by $d_S(P)$.*

Let $P(\boldsymbol{X})$ be an n variables polynomial over $\mathbb{Q}$, and (S_1, S_2) a partition of $[1, n]$. We can see P as a polynomial with variables $(X_i)_{i \in S_1}$ over the ring $\mathbb{Q}[(X_i)_{i \in S_2}]$. In fact, the total degree of P as a polynomial over this ring is equal to $d_{S_1}(P)$. We introduce a new variable X_{n+1}, the polynomial $\tilde{P}$ is the polynomial P where $X_{n+1}X_i$ is substituted to X_i when $i \in S_1$. We have the equality $d_{\{n+1\}}(\tilde{P}) = d_{S_1}(P)$.

We now propose an algorithm which, given a polynomial P and a set of indices S, finds $d_S(P)$. The polynomial is given as a black box, meaning that we can evaluate it on any input in unit time. We use the following lemma which states that we can decide with good probability if a polynomial is non zero by evaluating it on big enough points.

Lemma 1 (Schwartz-Zippel (cf. [15])). *Let P be a non zero polynomial with n variables of total degree D, if we chose randomly $x_1, \ldots, x_n$ in a set of integers S of size $\frac{D}{\epsilon}$ then the probability that $P(x_1, \ldots, x_n) = 0$ is bounded by ϵ.*

Proposition 2. *Let $P(\boldsymbol{X})$ be a polynomial with n variables, a total degree D and let S be a subset of $[n]$. There is an algorithm which finds $d_S(P)$ with probability greater than $\frac{1}{2}$ in time polynomial in n, D and the size of the coefficients of P.*

Proof. From P and S we define the polynomial with $n+1$ variables $\tilde{P}$ as explained previously. It is equal to

$$\sum_{i=0}^{d} (X_{n+1})^i Q_i(\boldsymbol{X})$$

where Q_d is a non zero polynomial. Here d is both $d_S(P)$ and the degree of $\tilde{P}$ seen as an univariate polynomial over $\mathbb{Q}[X_1, \ldots, X_n]$. Now choose randomly a value x_i in $[1, 2D]$ for each X_i. The polynomial $\tilde{P}(x_1, \ldots, x_n, X_{n+1})$ is a univariate polynomial, and the coefficient of $(X_{n+1})^d$ is $Q_d(\boldsymbol{x})$. By Lemma 1, the probability that $Q_d(\boldsymbol{x})$ is zero is bounded by $\frac{1}{2}$.

We can interpolate the polynomial $\tilde{P}(x_1, \ldots, x_n, X_{n+1})$ if we have its value on the integers $0, \ldots, D$, because it is of degree less or equal to $d_S(P)$, which is less than the total degree D. The value of $\tilde{P}(x_1, \ldots, x_n, x_{n+1})$ is equal to $P(x'_1, \ldots, x'_n)$ where $x'_i = x_{n+1}x_i$ if $i \in S$ and x_i otherwise. The time to interpolate $\tilde{P}(x_1, \ldots, x_n, X_{n+1})$ with s a bound on the size of $\tilde{P}(x_1, \ldots, x_n, x_{n+1})$ for $0 \leq x_{n+1} \leq D$ is $O(D^2 \log(s))$. Note that s is polynomial in the size of the coefficients of P and in D.

Finally, the interpolation of $\tilde{P}(x_1, \ldots, x_n, X_{n+1})$ gives its degree, which is equal to $d_S(P)$ with probability greater than $\frac{1}{2}$. $\square$

4.2 Application of the method

Thanks to the method given in the previous section, we are going to find the size of the largest Berge-acyclic subhypergraph of a given 3-uniform hypergraph. To this purpose we introduce a family of polynomials $Z_{\mathcal{H}}$, where each $Z_{\mathcal{H}}$ is associated to the hypergraph $\mathcal{H}$. The monomials of $Z_{\mathcal{H}}$ are in bijection with the spanning hypertrees – that is the connected Berge acyclic subhypergraphs– of $\mathcal{H}$, whose set is denoted by $\mathcal{T}(\mathcal{H})$.

Definition 2.

$$Z_{\mathcal{H}} = \sum_{T \in \mathcal{T}(\mathcal{H})} \epsilon(T) \prod_{e \in E(T)} w_e$$

where $\epsilon(T) \in \{-1, 1\}$.

This polynomial has exactly one variable w_e for each hyperedge e of $\mathcal{H}$. We also write $w_{\{i,j,k\}}$ for the variable associated to the hyperedge which contains the vertices i, j and k.

Definition 3. *Let* $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ *be a* 3*-uniform hypergraph,* $\Lambda(\mathcal{H})$ *is the Laplacian matrix defined by*

$$\Lambda(\mathcal{H})_{i,j} = \sum_{k \neq i,j} \epsilon_{i,j,k} w_{\{i,j,k\}},$$

where $w_{\{i,j,k\}}$ *is* 0 *when* $\{i, j, k\} \notin \mathcal{E}$ *and* $\epsilon_{i,j,k} \in \{-1, 1\}$.

In both cases, ϵ has a precise definition, see [14], but it is not needed for the present article. We may relate to $Z_{\mathcal{H}}$, the Pfaffian of the Laplacian matrix which is of interest since it is computable in polynomial time. The following theorem is inspired by a similar theorem for graphs called the Matrix-Tree Theorem.

Theorem 1 (Pfaffian-Hypertree (cf. [14])). *Let $\Lambda(i)$ be a minor of $\Lambda(\mathcal{H})$ where we have removed a column and a line of index i.*

$$Z_{\mathcal{H}} = (-1)^{i-1} Pf(\Lambda(i)).$$

For $\mathcal{H}$ a hypergraph with n vertices and m hyperedges, $Z_{\mathcal{H}}$ is a multilinear polynomial with m variables, its monomials are of total degree $2n-1$ and the size of its coefficient is one. Since we are interested in the acyclic subhypergraphs of a hypergraph, we now give a connection between them and the spanning hypertrees of a related hypergraph.

Proposition 3. *Let $\mathcal{H}_n$ be the complete 3-uniform hypergraph over n elements and $\mathcal{H}$ one of its acyclic subhypergraphs. We can extend $\mathcal{H}$ into a spanning subhypertree of $\mathcal{H}_n$, $\mathcal{H}_{n+1}$ or $\mathcal{H}_{n+2}$.*

Proof. Let $\mathcal{H}$ be an acyclic hypergraph with n vertices, let $C_0, \ldots, C_t$ be its connected components and $v_0, \ldots, v_t$ be vertices such that $v_i \in C_i$. The hypergraph $\mathcal{H}'$ is the union of the hyperedges of $\mathcal{H}$ and of $\{v_{2i}, v_{2i+1}, v_{2i+2}\}$ for $0 \leq i \leq \lfloor \frac{t}{2} \rfloor$. If v_{t+1} appears, it is a new vertex, thus $\mathcal{H}'$ is a subhypergraph of $\mathcal{H}_n$ or $\mathcal{H}_{n+1}$. Since we have connected by a path all connected components of $\mathcal{H}$ which is acyclic, $\mathcal{H}'$ is both acyclic and connected: it is a hypertree.

Let now $\mathcal{H}$ be a subhypertree on the vertices $1, \ldots, k$ of the hypergraph $\mathcal{H}_n$. The hypergraph $\mathcal{H}''$ is the union of the hyperedges of $\mathcal{H}$ and the hyperedges $\{2i+k, 2i+k+1, 2i+k+2\}$ for $0 \leq i \leq \lfloor \frac{n-k}{2} \rfloor$. Again we may introduce a new vertex labeled $n+1$, which makes $\mathcal{H}''$ a subhypergraph of $\mathcal{H}_{n+1}$ instead of $\mathcal{H}_n$. The hyperedges added to $\mathcal{H}$ form a path which covers all points not in $\mathcal{H}$, therefore $\mathcal{H}''$ spans either $\mathcal{H}_n$ or $\mathcal{H}_{n+1}$. Since this path has only one point in common with $\mathcal{H}$ which is acyclic, $\mathcal{H}''$ is also acyclic.

Combining the two constructions proves the result. □

The class NC is the set of decision problems decidable in polylogarithmic time on a parallel computer with a polynomial number of processors. It does not depend on the model of parallel computer and can be alternatively defined to be the decision problems decidable by a uniform Boolean circuit with polylogarithmic depth and a polynomial number of gates. A problem is in randomized NC, denoted by RNC, if it is decided with probability $\frac{3}{4}$ by a family of randomized Boolean circuits with polylogarithmic depth and a polynomial number of gates. For more details on parallel computation see [8].

Proposition 4. *For Berge-acyclicity and 3-uniform hypergraphs,* Maximal Acyclic Subhypergraph *is in RNC.*

Proof. Let $\mathcal{H}$ be a hypergraph on n vertices and k an integer, we want to decide if there is an acyclic subhypergraph of size k in $\mathcal{H}$. Consider the polynomial $Z_{\mathcal{H}_n}$, its monomials are in bijection with the spanning hypertrees of the complete hypergraph $\mathcal{H}_n$. We denote by S the set of indices of the variables of $Z_{\mathcal{H}_n}$ in bijection with the hyperedges of $\mathcal{H}$.

By Proposition 3, an acyclic subhypergraph $\mathcal{H}'$ of $\mathcal{H}$ can be extended into a spanning hypertree, say w.l.o.g. of $\mathcal{H}_n$, which is hence represented by a monomial of $Z_{\mathcal{H}_n}$. This monomial has a degree with regard to S equal to the size of $\mathcal{H}'$. Conversely, since the restriction of a spanning hypertree of $\mathcal{H}_n$ to $\mathcal{H}$ is acyclic, the degree of the corresponding monomial in $Z_{\mathcal{H}_n}$ with regard to S is the size of the restriction. Therefore the maximum of $d_S(Z_{\mathcal{H}_n})$, $d_S(Z_{\mathcal{H}_{n+1}})$ and $d_S(Z_{\mathcal{H}_{n+2}})$ is the maximal size of an acyclic subhypergraph of $\mathcal{H}$.

In order to find $d_S(Z_{\mathcal{H}_n})$, $d_S(Z_{\mathcal{H}_{n+1}})$ and $d_S(Z_{\mathcal{H}_{n+2}})$ we use the algorithm of Proposition 2. These polynomials have $O(n^3)$ variables, a total degree $O(n)$ and coefficient bounded in size by 1, hence the algorithm is in time polynomial in n and it does $O(n)$ evaluations on points of size less than $O(\log n)$. Thanks to Theorem 1, the evaluations can be done in time polynomial in n, thus we find the degrees in polynomial time with probability $\frac{1}{2}$. If the maximum of the degrees is more or equal to k, we are sure that there is an acyclic subhypergraph of size k, and if not there is none with probability $\frac{1}{2}$, which proves that the problem is in RP.

The evaluation of $Z_{\mathcal{H}_n}$ is the evaluation of a Pfaffian on values obtains by sums of the values of the variables of $Z_{\mathcal{H}_n}$. Since the Pfaffian is the square root of a determinant, one can compute it and then the polynomial $Z_{\mathcal{H}_n}$ with a circuit of logarithmic depth and polynomial size. The algorithm of Proposition 2 does one interpolation of a univariate polynomial obtained from $Z_{\mathcal{H}_n}$ and a random choice of values. The interpolation is done by solving a linear system built from the results of the evaluation of the polynomial. It is well known that it can be done in NC, therefore the whole algorithm is in RNC. $\square$

One may consider only the "3-Pfaffian" hypergraphs (cf. [7]), a property similar to the "Pfaffian" orientation, which is used to count perfect matchings in planar graphs [11]. In this case, the polynomial $Z_{\mathcal{H}}$ has only positive coefficients and the previous algorithm can be trivially made deterministic, and is thus in NC.

4.3 NP-complete cases

Proposition 5. *For the four notions Berge, γ, β and α-acyclicity and 4-uniform hypergraphs,* Maximal Acyclic Subhypergraph *is NP-complete.*

Proof. We give a polynomial time reduction from Hamiltonian Path to Maximal Acyclic Subhypergraph. More precisely, we consider the Hamiltonian Path problem for cubic graphs (i.e. graphs whose vertices have degree 3) and we look for Hamiltonian paths that are not Hamiltonian circuits. This instance of Hamiltonian Path is still NP-complete (cf. [6], p. 199).

Let $G = (V, E)$ be a cubic graph. We can assume that G is connected because else it can not have a Hamiltonian path. We define a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ as follows. We have

$$\mathcal{V} = \{x_y \mid \{x, y\} \in E\}$$

and

$$\mathcal{E} = \{\{x_a \mid a \neq y\} \cup \{y_b \mid b \neq x\} \mid \{x, y\} \in E\}.$$

Since G is cubic, each element of $\mathcal{E}$ has size 4. We show that G has a Hamiltonian path if and only if $\mathcal{H}$ has an acyclic subgraph $\mathcal{S}$ with $|V| - 1$ hyperedges. For each vertex x of G, there are three vertices x_a, x_b and x_c in $\mathcal{H}$ and three hyperedges

$$\{x_b, x_c, a_p, a_q\}, \{x_a, x_c, b_r, b_s\} \text{ and } \{x_a, x_b, c_u, c_t\}.$$

A hypergraph which contains those three hyperedges is not θ-acyclic for $\theta \in \{\text{Berge}, \gamma, \beta, \alpha\}$. Since $\mathcal{S}$ is acyclic, it does not contain them and the subgraph S of G with edges

$$\{\{x, y\} \mid \{x_a \mid a \neq y\} \cup \{y_b \mid b \neq x\} \text{ is a hyperedge of } \mathcal{S}\}$$

has maximal degree 2. S also have $|V|-1$ edges because $\mathcal{S}$ has $|V|-1$ hyperedges, hence it is a Hamiltonian path of G.

Conversely, it is clear that, if S is a Hamiltonian path of G then the hypergraph with hyperedges

$$\{\{x_a \mid a \neq y\} \cup \{y_b \mid b \neq x\} \mid \{x, y\} \text{ is an edge of } S\}$$

is a θ-acyclic subhypergraph of $\mathcal{H}$ with $|V| - 1$ hyperedges. □

Proposition 6. *For the three notions γ, β and α-acyclicity and 3-uniform hypergraphs,* Maximal Acyclic Subhypergraph *is NP-complete.*

Proof. We make a polynomial time reduction to Maximal Acyclic Subhypergraph from the same instance of Hamiltonian Path as in the preceding proof. As previously, given a cubic graph $G = (V, E)$, we define a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ such that, for every $v \in V$ and the three vertices x, y and z such that $\{v, x\}$, $\{v, y\}$ and $\{v, z\}$ are edges of G, we have in $\mathcal{V}$ three vertices v_x, v_y and v_z. This time, instead of a hyperedge in the form $\{s_a, s_b, t_e, t_f\}$ for every $\{s, t\} \in E$, we have in $\mathcal{E}$ two hyperedges $\{s_a, s_b, t'\}$ and $\{s', t_e, t_f\}$ where t' is any vertex among t_e and t_f and s' is any vertex among s_a and s_b. Now, G has a Hamiltonian path if and only if $\mathcal{H}$ has a θ-acyclic subhypergraph $\mathcal{H}' = (\mathcal{V}', \mathcal{E}')$ of size $2(|V| - 1)$ for $\theta \in \{\gamma, \beta, \alpha\}$. Indeed, if $\mathcal{H}$ has a θ-acyclic subhypergraph $\mathcal{H}'$ of size $2(|V| - 1)$ then, for every $v \in V$, $\mathcal{E}'$ can not contain the three hyperedges in the form $\{v_x, v_y, z_g\}$, $\{v_x, v_z, y_d\}$ and $\{v_y, v_z, x_a\}$ because it θ-acyclic. Also, if it contains a hyperedge $\{v_x, v_y, z_g\}$ then it also contains the hyperedge in the form $\{z_g, z_h, v'\}$ where $v' \in \{v_x, v_y\}$ because it has $2(|V| - 1)$ hyperedges and does not contain a γ-cycle (this is for the same reasons that a connected graph with at least as much edges than vertices necessarily contains a cycle). The same reasoning as in the preceding proof completes the proof. We just have to check that if S is a Hamiltonian path of G then the subhypergraph $\mathcal{S}$ of $\mathcal{H}$ with hyperedges $\{s_a, s_b, t'\}$ and $\{s', t_e, t_f\}$ for every $\{s, t\}$ edge of S is γ-acyclic. For this, we use the four rules deciding γ-acyclicity described in Section 2. Since

S is a path, it has a vertex u of degree 1. Let v be the vertex such that $\{u, v\}$ is an edge of S. We can apply the rules so that we remove from $\mathcal{S}$ the vertices u_v and u_b by Rule 1, u_a by Rule 4 (because v_d belongs to the same hyperedges as u_a), the hyperedge $\{v_d\}$ by Rule 2, v_d by Rule 1 and the hyperedge $\{v_e\}$ by Rule 2 (see Figure 2). By applying repeatedly the rules with the remaining elements of $\mathcal{S}$ we will be able to obtain the empty hypergraph.

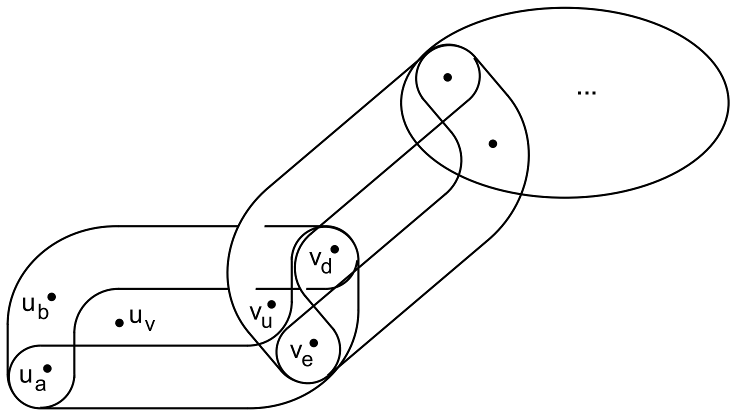

Fig. 2. The part of $\mathcal{S}$ associated with the edge $\{u, v\}$ of S.

□

5 Conclusion

Figures 3 and 4 give a recap of the results in this paper.

$\theta =$	Berge	γ, β and α
$k = 3$	P (cf. [12]) $\cap$ RNC (cf. [13])	NP-complete
$k \geq 4$	NP-complete (cf. [13])	NP-complete

Fig. 3. Complexity of Spanning θ-Acyclic Subhypergraph for k-uniform hypergraphs.

There are two possible roads to further study the problem Maximal θ-Acyclic Subhypergraph. The first is to find either an approximation algo-

$\theta =$	Berge	γ, β and α
$k = 3$	P (cf. [12]) $\cap$ RNC	NP-complete
$k \geq 4$	NP-complete	NP-complete

Fig. 4. Complexity of Maximal θ-Acyclic Subhypergraph for k-uniform hypergraphs.

rithm or to prove unapproximability results for the NP-complete cases of Maximal θ-Acyclic Subhypergraph.

The second is to see it as a problem parametrized by the size of the acyclic subhypergraph. A hypergraph is Berge-acyclic if and only if there is an order on the edges such that for all edges E, the union of all edges greater than E has an intersection with E of size less than one. One may express the fact that a hypergraph represented by an incidence structure, equipped with an order, satisfies this condition by means of a Π_1 formula. This means that Maximal *Berge*-Acyclic Subhypergraph is in the class W[1]. Now an open question is to know if it is W[1]-complete or FPT. The same kind of expressiveness in Π_1 can be done for some other acyclicity notions, so of course the same question holds.

References

1. C. Beeri, R. Fagin, D. Maier, A. Mendelzon, J. Ullman, and M. Yannakakis. Properties of acyclic database schemes. In *Proceedings of the 13th annual ACM symposium on Theory of computing*, pages 355–362, 1981.
2. C. Berge. *Graphs and hypergraphs.* North Holland, 1976.
3. A. E. Brouwer and A. W. J. Kolen. A super-balanced hypergraph has a nest point. Technical report, Stichting Mathematisch Centrum, 1980.
4. D. Duris. Hypergraph acyclicity and extension preservation theorems. In *Proceedings of the 23rd Annual IEEE Symposium on Logic in Computer Science*, pages 418–427, 2008.
5. R. Fagin. Degrees of acyclicity for hypergraphs and relational database schemes. *Journal of the ACM*, 30(3):514–550, July 1983.
6. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., 1990.
7. A. Goodall and A. de Mier. Spanning trees of 3-uniform hypergraphs. Preprint available as arXiv:1002.3331v1, February 2010.
8. R. Greenlaw, H.J. Hoover, and W.L. Ruzzo. *Limits to parallel computation: P-completeness theory.* Oxford University Press, USA, 1995.
9. S. Halperin and U. Zwick. Optimal randomized EREW PRAM algorithms for finding spanning forests and for other basic graph connectivity problems. In *Proceedings of the seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 438–447. Society for Industrial and Applied Mathematics, 1996.

10. K. Hirata, M. Kuwabara, and M. Harao. On finding acyclic subhypergraphs. *Lecture Notes in Computer Science*, 3623/2005:491–503, 2005.
11. P.W. Kasteleyn. The statistics of dimers on a lattice. *Physica*, 27(12):1209–1225, 1961.
12. L. Lovász. Matroid matching and some applications. *Journal of Combinatorial Theory, Series B*, 28(2):208 – 236, 1980.
13. G. Masbaum, S. Caracciolo, A. Sokal, and A. Sportiello. A randomized polynomial-time algorithm for the spanning hypertree problem on 3-uniform hypergraphs. Preprint available as arXiv:0812.3593.
14. G. Masbaum and A. Vaintrob. A new matrix-tree theorem. *International Mathematics Research Notices*, 2002(27):1397, 2002.
15. J. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27:701–717, October 1980.
16. J. Wang and H. Li. Counting acyclic hypergraphs. *Science in China Series A: Mathematics*, 44(2):220–224, 2001.