A Representation Theorem for union-difference Families and Application

Binh-Minh Bui-Xuan, Michel Habib

A Representation Theorem for union-difference Families and Application[*]

B.-M. Bui Xuan[1] and M. Habib[2]

[1] CNRS - LIRMM - Université Montpellier II
161 rue Ada, 34392 Montpellier Cedex 05, France. `buixuan@lirmm.fr`
[2] CNRS - LIAFA - Université Paris Diderot
Paris 7 Case 7014, 75205 Paris Cedex 13, France. `habib@liafa.jussieu.fr`

Abstract. We give a quadratic $O(|X|^2)$ space representation based on a canonical tree for any subset family $\mathcal{F} \subseteq 2^X$ holding the closure under union and difference of overlapping members. The cardinal of $\mathcal{F}$ is potentially in $O(2^{|X|})$, and its size higher. As far as we know this is the first representation theorem for such families. As an application of this framework we obtain a uniqueness decomposition theorem on a digraph decomposition that captures and is strictly more powerful than the well-studied modular decomposition. Moreover a polynomial time decomposition algorithm for this case is described.

1 Introduction

Many combinatorial decompositions lead to interesting subset families, such as crossing families for minimum cuts in network flows theory (see e.g. [19]), and partitive families for modular decomposition in graph theory [2, 8, 17]. Cross-free families as defined in [19] using the famous Edmonds-Giles's theorem [7] admit a tree structure and arise in many combinatorial decompositions such as the split decomposition [4–6, 16] and also in phylogeny [20].

For a given set family $\mathcal{F} \subseteq 2^X$, it is worth to study its distance from a tree structure, namely to examine if it can be represented via a tree. Such a representation must allow the enumeration of all members of the family in $O(|\mathcal{F}|)$ time. Let us define the complexity of a family as the size of its minimal tree. At first level one can find simple hierarchies (c.f. laminar in [19]) and cross-free families. Then, (weakly) partitive families which admit a unique tree decomposition with 3 types of nodes (prime, complete and linear) also have complexity $O(|X|)$. For crossing families only a representation tree in $O(|X|^2)$ space is known [11].

This paper deals with union-difference families – families closed under union and difference of overlapping elements – which is a natural generalisation of partitive families. We show the existence of a canonical tree representation in $O(|X|^2)$ space. Furthermore, from this we obtain a uniqueness decomposition theorem on a new polynomial decomposition of directed graphs, generalising modular decomposition. A polynomial time decomposition algorithm for this case is then depicted in the last section.

2   Representation Theorem

Let X be a finite set. Those that belong to X are called *elements*, sets included in X are *subsets*, and sets of subsets of X are *families*. For more clarity, elements of a family are called *members*. Two subsets A and B *overlap*, denoted by $A \textcircled{\circ} B$, if none among $A \cap B$, $A \setminus B$, and $B \setminus A$ is empty. They *cross*, if we have both $A \textcircled{\circ} B$ and $\overline{A} \textcircled{\circ} \overline{B}$, where $\overline{A} = X \setminus A$.

$\mathcal{F} \subseteq 2^X$ is a *union-difference* family if: $\mathcal{F}$ contains the *trivial members* X and $\{x\}$ (for all $x \in X$), and $\mathcal{F}$ is closed under union and difference of overlapping members. If a union-difference family is also closed under symmetric difference of overlapping members, then it is closed under intersection of overlapping members. Union-difference-intersection families are well-studied under the name of *partitive families* [2], and fundamental for modular graph decomposition [8, 17].

Henceforth $\mathcal{F}$ is a union-difference family. Notice that if $|X| \leq 2$ then $\mathcal{F} = 2^X$ and representing $\mathcal{F}$ is trivial. We assume throughout the paper that $|X| \geq 3$. $A \in \mathcal{F}$ is a *strong member of* $\mathcal{F}$ if A does not overlap any $B \in \mathcal{F}$. Likewise, $A \in \mathcal{F}$ is a *semi-strong member of* $\mathcal{F}$ if it does not cross any $B \in \mathcal{F}$. Let $\mathcal{S} \subseteq \mathcal{F}$ be the family of semi-strong members of $\mathcal{F}$. For sake of simplicity, X is *excluded* from $\mathcal{S}$ although it is vacuously semi-strong. By definition, no two members of $\mathcal{S}$ cross, and $\mathcal{S}$ is called *cross-free*.

Let us now recall a cross-free family representation [7] which is widely used in combinatorial optimisation research areas (refer to e.g. [19]). Let $x \in X$, we consider $\mathcal{S}' = \{A \mid A \in \mathcal{S} \land x \notin A\} \cup \{\overline{A} \mid A \in \mathcal{S} \land x \in A\}$. No two members of $\mathcal{S}'$ overlap, and their inclusionwise ordering results in a tree rooted at $X \setminus \{x\}$. We then add x to the children of the root and unroot the tree. The set of leaves is
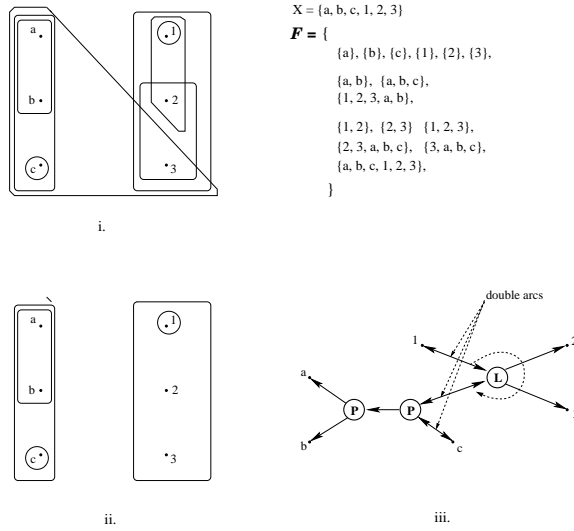

Fig. 1. i. A union-difference family, circles represent their complement. ii. The semi-strong subfamily excluding X. iii. Decomposition tree.

now in bijection with X: by abusiveness we confound the two sets. In this tree, deleting any edge gives rise to two connected components. If each component is regarded as the set of its leaves, then at least one of them is a member of $\mathcal{S}$. Then, edge orientation can denote which ones belong to $\mathcal{S}$ (see Fig. 1). On the other hand, each member of $\mathcal{S}$ corresponds to one edge of the tree.

Definition 1 (Decomposition tree). *We define the* decomposition tree *of a union-difference family $\mathcal{F} \subseteq 2^X$ as the Edmonds-Giles's tree representation [7] of its semi-strong members, X is excluded. Such a tree has no degree 2 node.*

We shall label this tree to obtain an enumerating object of all members of $\mathcal{F}$. In the tree, the deletion of an internal node n gives rise to $k = d(n)$ connected components, which can also be seen as a $k-$partition of X. Let $\{X_1, \ldots, X_k\}$ denote this partition. For instance, the node labelled "L" in Fig. 1 yields $\{\{1\}, \{2\}, \{3\}, \{a, b, c\}\}$. For later use, notice that we always have $k \geq 3$. (This can also be seen as a quick proof that the unlabelled decomposition tree is of linear $O(|X|)$ size.) Let us consider $Y = \{X_1, \ldots, X_k\}$ as a set, and define the *quotient of $\mathcal{F}$ w.r.t. node n* as the family $\mathcal{Q}(n) \subseteq 2^Y$ such that

$$\begin{cases} \{X_i\} \text{ belongs to } \mathcal{Q}(n) \text{ for all } 1 \leq i \leq k, \\ Q = \{X_i \mid i \in I\} \text{ with } |Q| \neq 1 \text{ belongs to } \mathcal{Q}(n) \iff \cup_{i \in I} X_i \text{ belongs to } \mathcal{F}. \end{cases}$$

The membership of X_i in $\mathcal{F}$ (resp. exclusion of X_i from $\mathcal{F}$) is *already* stored by the edge orientation of the decomposition tree. Roughly, each member Q of the quotient $\mathcal{Q}(n)$ corresponds to one and only one member of $\mathcal{F}$, except for the singletons $\{X_i\}$. Moreover, it is not obvious but folklore that the converse holds:

Proposition 1. *For all member $A \in \mathcal{F}$ of a subset family $\mathcal{F} \subseteq 2^X$, there exists a node n in the semi-strong tree of $\mathcal{F}$ such that A corresponds to a member of the quotient $\mathcal{Q}(n)$ of $\mathcal{F}$ w.r.t. n. This node is unique.*

Consequently, if there is a way to describe $\mathcal{Q}(n)$ for every node n of the decomposition tree, then one can rebuild the initial family $\mathcal{F}$ in an exact manner. As a step towards this aim, we say that a member A of $\mathcal{F}$ is *quasi-trivial* if $|A| = |X| - 1$, and notice a second non obvious but folklore fact:

Proposition 2. *Trivial and quasi-trivial members are vacuously semi-strong. On the other hand, any semi-strong members of a quotient $\mathcal{Q}(n) \subseteq 2^Y$ is either trivial or quasi-trivial.*

Remark 1. Both Propositions 1 and 2 hold for arbitrary subset families.

Definition 2 (Quotient property). *We say that a subset family satisfies the* quotient property *if all its semi-strong members are either trivial or quasi-trivial.*

Obviously the quotient of a union-difference family is also a union-difference family. We thus focus on families satisfying both union-difference and quotient properties, which form a super-class of the quotient nodes of a union-difference decomposition tree. We shall prove that there are at most 5 types of them. Moreover, each type will be proved to be of "small enough" size, that is

Main Representation Theorem. *There is a node-labelling of the decomposition tree of a union-difference family $\mathcal{F} \subseteq 2^X$ such that every member of $\mathcal{F}$ can be retrieved from the tree and its labels. Moreover, the size of the tree and its labels is in $O(|X|^2)$.*

Proof. We shall consider two main categories. *Simply-linked* quotients (see further in Definition 3) will be characterised by Theorem 1 into 4 types. Section 2.2 addresses the remaining ones. Theorem 2 proves the quadratic global size.   □

Before continuing, let us highlight a useful tool from previous works on partitive families. A subset family $\mathcal{F} \subseteq 2^X$ can also be seen as an undirected hypergraph with vertex set X. Let us define the $2-graph$ of $\mathcal{F}$ as its restriction to size 2 hyperedges: $G_\mathcal{F} = (X, E)$ with $E = \{A \in \mathcal{F} \text{ and } |A| = 2\}$. Though the following property was discovered for partitive families, its proof only requires the union and difference closures. The proof given in [8] is recalled in Appendix.

Lemma 1. *(c.f. [2, 8] with partitive families) Let $\mathcal{F}$ be a union-difference family. If its $2-graph$ $G_\mathcal{F}$ is connected then $G_\mathcal{F}$ is either a clique, a path, or a cycle.*

2.1   Simply-linked quotients

We first focus on a case of "easy" decomposition, fully exploiting Lemma 1. Though vacuously semi-strong, a quasi-trivial member is not necessarily strong. Moreover, we say that

Definition 3 (Simply-linked Property). *A subset family is* simply-linked *if none of its quasi-trivial members is strong.*

For simply-linked quotients, the following nice theorem holds. A family is *prime* if it has only trivial and quasi-trivial members.

Theorem 1. *If a union-difference family $\mathcal{F}$ satisfies both quotient and simply-linked properties, then one and only one of the following holds:*

 – *$G_\mathcal{F}$ is a clique (we say that $\mathcal{F}$ is complete),*
 – *$G_\mathcal{F}$ is a path (we say that $\mathcal{F}$ is linear),*
 – *$G_\mathcal{F}$ is a cycle (we say that $\mathcal{F}$ is circular),*
 – *$\mathcal{F}$ is prime.*

Proof. First we have to prove the two lemmas 2 and 3 (below). Then, notice by Lemma 1 that if $G_\mathcal{F}$ is connected, it is either a clique, a path, or a cycle.   □

By union closure, $G_\mathcal{F}$ is a clique if and only if $\mathcal{F} = 2^X$, and we say that $\mathcal{F}$ is *complete*. Likewise, $G_\mathcal{F}$ is a path (resp. cycle), if and only if there is a linear (resp. circular) ordering of X such that $\mathcal{F}$ is exactly the family of all intervals (resp. circular intervals) of this ordering. $\mathcal{F}$ is then *linear* (resp. *circular*).

Corollary (Representing simply-linked quotients). Let $X_1, \ldots, X_k$ denote the resulting connected components of a decomposition tree when deleting a quotient node. Representing a *complete* quotient node is easily done with $O(1)$

label, stating the quotient is the family of all union of some X_i's. For a *linear* or *circular* node, we also need to code an ordering on the incident edges. Then, an $O(1)$ label can state the quotient is the family of all union of some consecutive X_i's (Fig. 1 gives an illustration on the node labelled "L"). Except for the special case of X, members of a *prime* quotient node are *already* stored in the edge orientation of the decomposition tree (they are semi-strong, and belong to $\mathcal{S}$). Accordingly, we only need an $O(1)$ label for all prime nodes, stating there are no members bound to the node other than those given by the edge orientation.

Let us head back to the proof of Theorem 1. A *chain* of length k of $\mathcal{F}$ is a sequence $(A_1, \ldots, A_k)$ of members of $\mathcal{F}$ such that $A_i \oslash A_{i+1}$ for all i, and $A_i \cap A_j = \emptyset$ for all $|i - j| > 1$. The chain is *covering* if $A_1 \cup \cdots \cup A_k = X$, and *irreducible* if $|A_i| = 2$ for all $1 \leq i \leq k$. An irreducible and covering chain of $\mathcal{F}$ can also be seen as a Hamiltonian path in the $2-$graph $G_{\mathcal{F}}$, which would imply its connectivity, and enable the use of Lemma 1.

Lemma 2. *If a union-difference family $\mathcal{F}$ satisfies both quotient and simply-linked properties, then $\mathcal{F}$ is either prime, or has a length 3 covering chain.*

Proof. Suppose that $\mathcal{F}$ is not prime, and let $A \in \mathcal{F}$ be neither trivial nor quasi-trivial. We take A maximal by inclusion. The quotient property provides us with $B \in \mathcal{F}$ such that A and B cross. The closure under union implies $A \cup B \in \mathcal{F}$. Moreover, A is maximal. Hence $A \cup B$ is either trivial or quasi-trivial. However, $A \cup B$ cannot be trivial since A and B cross. Then, the simply-linked property implies that $A \cup B$ is not strong. Hence it overlaps some member $C \in \mathcal{F}$. Here, all cases lead to either $D = C \cup B \setminus A$ or $E = C \cup A \setminus B$ is a member of $\mathcal{F}$. Then, either (A, B, D) or (B, A, E) is a covering chain of length 3. $\square$

Lemma 3. *If a union-difference family $\mathcal{F}$ satisfies both quotient and simply-linked properties, and has a covering chain of length at least 3, then $\mathcal{F}$ has an irreducible covering chain (then $G_{\mathcal{F}}$ is connected).*

Proof. By hypothesis $\mathcal{F}$ has a covering chain $\mathcal{A} = (A_1, \ldots, A_k)$ with $k \geq 3$. We take k maximum. Assume for some $1 < i < k$ that $A_i \setminus (A_{i-1} \cup A_{i+1}) \neq \emptyset$. In this case $B = A_i \setminus A_{i+1}$ and $C = A_i \setminus A_{i-1}$ are overlapping members of $\mathcal{F}$ (see Figure 2(a)). Then, replacing $\mathcal{A}$ with $(A_1, \ldots, A_{i-1}, B, C, A_{i+1}, \ldots, A_k)$ would improve k. Hence, $A_i \setminus (A_{i-1} \cup A_{i+1}) = \emptyset$ for all $1 < i < k$.

We now assume that $|A_i| > 2$ for some $1 < i < k$. Then at least one among $B = A_i \setminus A_{i+1}$ and $C = A_i \setminus A_{i-1}$ is neither trivial nor quasi-trivial (hence not semi-strong by quotient property). By symmetry we suppose it was B. Let $D \in \mathcal{F}$ cross B. We show in all cases a contradiction as follows (see also Figure 2(b)).

- *Case 1: $D \subseteq A_i$.* Among other, D and A_{i-1} overlap. Let $E = A_{i-1} \setminus D$, we can improve k by replacing $\mathcal{A}$ with $(A_1, \ldots, A_{i-2}, E, B, D, A_{i+1}, \ldots, A_k)$.
- *Case 2: $D \setminus A_i \neq \emptyset$ and $C \setminus D \neq \emptyset$.* Then, we are conducted to *Case 1* by replacing D with $D' = A_i \setminus D$.
- *Case 3: $D \setminus A_i \neq \emptyset$ and $C \subseteq D$.* We define the left and right as $L = A_1 \cup \cdots \cup A_{i-2}$ and $R = A_{i+1} \cup \cdots \cup A_k$. Notice that $L \cup R = \overline{\overline{B}}$. Since D
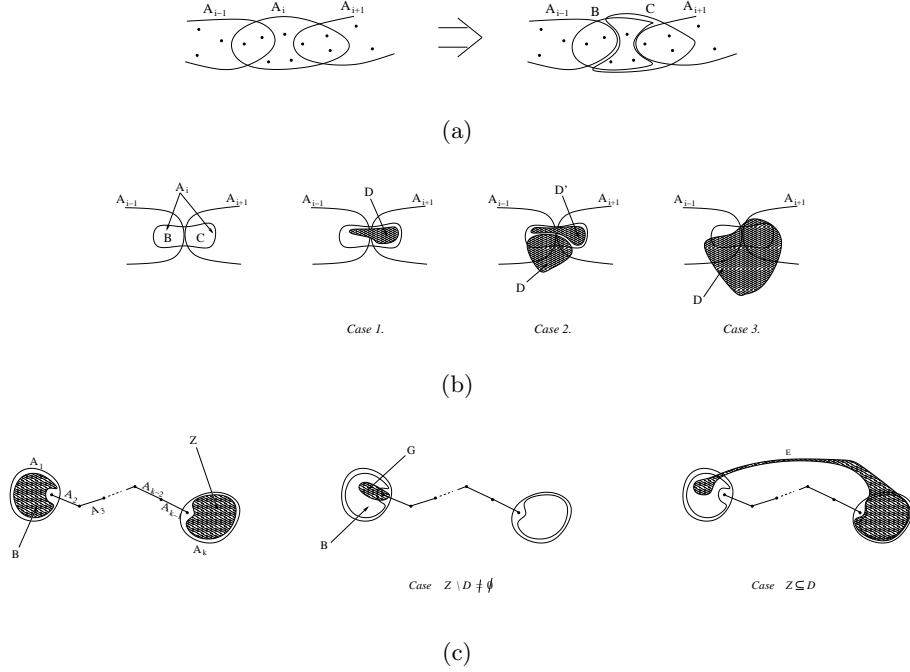
(a)

Case 1.   Case 2.   Case 3.

(b)

Case $Z \setminus D \neq \emptyset$ Case $Z \subseteq D$

(c)

Fig. 2. Illustration for the proof of Lemma 3.

and B cross, there is some element in either L or R that does not belong to D. If it was L, replacing D with $A_i \setminus (A_{i-1} \setminus (D \setminus L))$ leads back to *Case 1*. If it was R, the same can be done with $A_i \setminus (D \setminus R)$.

Hence, $|A_i| = 2$ for all $1 < i < k$. Now, assume that $|A_1| > 2$, and let $D \in \mathcal{F}$ cross $B = A_1 \setminus A_2$. Let $Z = A_k \setminus A_{k-1}$. We shall examine whether $Z \setminus D \neq \emptyset$ or $Z \subseteq D$ (see Figure 2(c)). In the first case, let $E = A_3 \cup \cdots \cup A_k$ and $F = D \cup A_2 \cup \cdots \cup A_{k-1}$: they overlap. Then, $G = F \setminus E$ is a member of $\mathcal{F}$, and replacing $\mathcal{A}$ with $(B, G, A_2, \ldots, A_k)$ would improve k. In the second case, since D and B cross, there is some element in $A_2 \cup \cdots \cup A_{k-1}$ that does not belong to D. In other words, $A_2 \cup \cdots \cup A_{k-1}$ and D overlap. Then, $E = D \setminus (A_2 \cup \cdots \cup A_{k-1})$ is a member of $\mathcal{F}$ which contains Z. That $E \in \mathcal{F}$ implies $(E, A_1, \ldots, A_{k-1})$ is a chain of $\mathcal{F}$. That E contains Z implies the chain is covering. Moreover, it is of length k, i.e. of maximum length. However, from the last paragraph, this chain cannot have A_1 with more than two elements. Therefore, $|A_1| = 2$. Then, by symmetry we obtain $|A_k| = 2$, and $\mathcal{A}$ is an irreducible covering chain.   □

2.2   Other quotients

We now address a family $\mathcal{F} \subseteq 2^X$ that is not simply-linked. By definition it has a quasi-trivial member that is strong. We note $Y = X \setminus \{x\}$ with $x \in X$ that member. Since Y is strong, except for X and $\{x\}$, $\mathcal{F}$ has no other member

containing x. Let us consider the sub-family $\mathcal{G} = \mathcal{F} \setminus \{X, \{x\}\}$, which holds $\mathcal{G} \subseteq 2^Y$. Obviously, if $\mathcal{F}$ is a union-difference family, so is $\mathcal{G}$. Fainthearted, we represent $\mathcal{F}$ with the member $\{x\}$ and the union-difference decomposition tree of $\mathcal{G}$. We process the same way with all quotient nodes of a union-difference decomposition tree that are not simply-linked. Therefore, such a tree may have recursive levels. Fortunately enough, its total size still is polynomial:

Theorem 2. *The global size of the labelled decomposition tree of a given union-difference family $\mathcal{F} \subseteq 2^X$ is in $O(|X|^2)$.*

Proof. By induction on $n = |X|$. Let $f(n)$ be the maximum size of all decomposition trees of n leaves. Obviously, $f(1)$ and $f(2)$ are non null constants. Let $f(k) \leq \alpha \times k^2$ hold for all $k < n$. We suppose without loss of generality that α is greater than any other constant in this proof. Let us consider a decomposition tree of n leaves and let N be the set of its internal nodes. For each $i \in N$, let n_i be its degree. The label of i is either of constant size (c.f. prime and complete nodes), of linear size on n_i (c.f. linear and circular nodes), or of size bounded by $f(n_i - 1) + \beta$ (c.f. nodes that are not simply-linked). In all cases, it is bounded by $\alpha \times (n_i - 1)^2 + \alpha$ since $n_i \geq 3$ and $\alpha \geq \beta$. The total size of leaves, edges, and orientations is linear on n, hence bounded by $\alpha \times n$. We deduce that

$$f(n) \leq \alpha \times \left(\sum_{i \in N} ((n_i - 1)^2 + 1) + n \right) \leq \alpha \times \left(\sum_{i \in N} (n_i - 1)^2 + n' + n \right),$$

where $n' = |N|$. Notice that $\sum_{i \in N} n_i = n + 2 \times (n' - 1)$ (the n pendant edges are counted once while other edges are counted twice). In other words, $S = \sum_{i \in N} (n_i - 1) = n + n' - 2$. Then, the greatest value that $\sum_{i \in N} (n_i - 1)^2$ can reach happens when one among the n_i gets the greatest value possible. Since $n_i - 1 \geq 2$, we have $\sum_{i \in N} (n_i - 1)^2 \leq (n' - 1) \times 2^2 + (S - (n' - 1) \times 2)^2$. Then, $f(n) \leq \alpha \times (n^2 + n'^2 + 5n' + n(1 - 2n') - 4)$. Besides, that there are no degree 2 nodes in the tree provides us with $n \geq n' + 2$. Finally, combining the previous facts and $1 - 2n' \leq 0$ allows to conclude. $\square$

3 Application to Graphs: Sesquimodular Decomposition

In graph theory modular decomposition is now a well-studied notion $[2, 8, 13, 17]$, as well as some of its generalisations $[4–6, 15, 16, 18]$. As having been rediscovered in other fields, the notion also appears under various names, including intervals, externally related sets, autonomous sets, partitive sets, and clans. Direct applications of modular decomposition include tractable constraint satisfaction problems $[3]$, computational biology $[12]$, graph clustering for network analysis, and graph drawing. This rich research field lays heavily on the nice combinatorial properties of modules. Among most important ones, that modules form a partitive family allows representing them compactly with a tree $[2, 8, 17]$.

Besides, in the area of social networks, several vertex partitioning have been introduced in order to catch the idea of putting in the same part all vertices
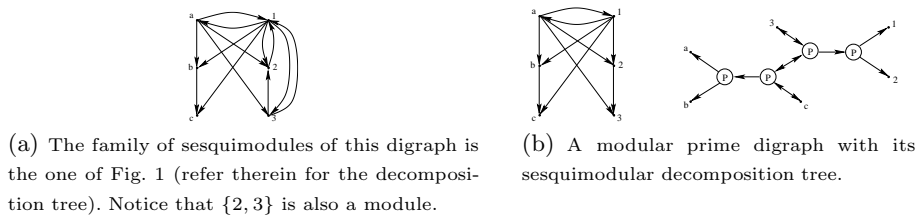
(a) The family of sesquimodules of this digraph is the one of Fig. 1 (refer therein for the decomposition tree). Notice that $\{2, 3\}$ is also a module.

(b) A modular prime digraph with its sesquimodular decomposition tree.

Fig. 3. Modules v.s. sesquimodules.

acknowledging similar behaviour, in other words finding regularities [21]. Modular decomposition provides such a partitioning, yet seemingly too restrictive for real life applications. The concept of a role [9] on the other hand seems promising, however its computation unfortunately is $NP-$hard [10]. As a natural consequence, there is need for the search of *relaxed*, but *tractable*, variations of the modular decomposition scheme. We here investigate the case of directed graphs, and propose a weakened definition of module in order to further decompose. Fortunately enough, we still obtain a well-structured variation, thanks to union-difference families.

Digraphs here refer to loopless simple directed graphs where $2-$cycles are allowed. Let $G = (V, A)$ be a digraph, $M \subseteq V$ is a *sesquimodule* if:

– $\forall x, y \in M$, $N^-(x) \setminus M = N^-(y) \setminus M$.
– $\forall x, y \in M$, either $N^+(x) \setminus M = N^+(y) \setminus M$ or $N^+(x) \setminus M = \overline{N^+(y)} \setminus M$.

In an undirected graph, there is only one requirement to be a module, which is $\forall x, y \in M, N(x) \setminus M = N(y) \setminus M$. The classical generalisation to directed graphs requires *two full conditions*, one on in-neighbours and one on out-neighbours: $\forall x, y \in M, N^-(x) \setminus M = N^-(y) \setminus M$ and $N^+(x) \setminus M = N^+(y) \setminus M$. In the new definition, there is a full condition on in-neighbours, and a relaxed one on out-neighbours: the exterior still has to be partitioned into out-/non-out-neighbour vertices, however, their order is irrelevant. This is the reason for the terminology. Fig. 3(a) exemplifies an instance where the sesquimodules form the family given in Fig. 1, while Fig. 3(b) shows that the generalisation of modules to sesquimodules is proper. A more complex example of sesquimodular decomposition tree is given in Appendix. We have the following theorem.

Theorem 3 (Uniqueness Decomposition Theorem). *There is a unique unrooted tree associated to a digraph $G = (V, A)$ such that: the leaves of the tree are in one-to-one correspondence with the vertices of G; the edges of the tree are oriented; the internal nodes of the tree are marked with at most 4 types of labels; and all sesquimodules of G can be generated from this tree without the knowledge of the graph. The size of this tree and its labels is in $O(|V|^2)$.*

This theorem lays on the simple fact that

Proposition 3. *The sesquimodules of a digraph form a union-difference family. Furthermore there are no circular nodes in its decomposition tree.*

Proof. Let $G = (V, A)$ be a digraph. Vacuously, the trivial vertex subsets are sesquimodules of G. Let X and Y be two overlapping sesquimodules of G. It follows straight from definition that $X \cup Y$ is a sesquimodule. We only need to prove that $Z = X \setminus Y$ is also a sesquimodule.

First suppose that there exist an exterior vertex $s \notin Z$ and two vertices $x, y \in Z$ s.t. $(s, x) \in A$ and $(s, y) \notin A$. We shall denote arc $(x, y) \in A$ by xy, and non-arc $(x, y) \notin A$ by $\overline{xy}$. Since X is a sesquimodule s belongs to $X \cap Y$. Moreover, that X and Y overlap implies there is a vertex t belonging to $Y \setminus X$. Notice that $s, t \in Y$ and $x, y \notin Y$. Additionally, we have sx and $\overline{sy}$. Since Y is a sesquimodule, we have either $tx \wedge \overline{ty}$ or $\overline{tx} \wedge ty$. But then X no more is a sesquimodule as $t \notin X$ and $x, y \in X$. Hence, for all $x, y \in Z$, $N^-(x) \setminus Z = N^-(y) \setminus Z$.

Now let $x, y \in Z$ and $s, t \notin Z$. For convenience, we refer to the fact that either $xs \wedge xt$ or $\overline{xs} \wedge \overline{xt}$ by "x *is not a splitter of* $\{s, t\}$", denoted by $x|st$. We need to prove that $x|st \Leftrightarrow y|st$. If none of s and t belong to X, that X is a sesquimodule allows to conclude. If both s and t belong to Y, that Y is a sesquimodule allows to conclude. By symmetry, the only remaining case is when $s \in X \cap Y$ and $t \notin X \cup Y$. In this case, let $u \in Y \setminus X$. Since X is a sesquimodule, we already have $x|tu \Leftrightarrow y|tu$, but we would like the same property with vertex u replaced by vertex s. For this, notice that $x \notin Y$, but $s, u \in Y$, and Y is a sesquimodule. Therefore, $x|su$. Likewise, $y|su$. Then, combining the two latter facts and $x|tu \Leftrightarrow y|tu$ leads to the desired property.

Finally, a circular sesquimodule quotient node would be a complete one. $\square$

4 Polynomial Time Algorithm for Sesquimodular Decomposition

This section describes a brute-force algorithm to compute in polynomial time the sesquimodular decomposition tree of a given digraph $G = (V, A)$. We divide the computation into two main steps, generalising the two-step scheme introduced by [1] for modular graph decomposition.

Definition 4 (Factoring Permutation). *[1] A* factoring permutation *of a decomposition tree is the visit order of the leaves of the underlying decomposition tree by some depth-first graph search.*

For sesquimodular decomposition tree, which is unrooted, we define the factoring permutation as a circular permutation. This notion dues its name to the fact that every node of the tree is a (circular) interval of the (circular) permutation. In the following, results of Section 4.1 can not be used unless we meet a certain notion of *splitter*. However, all the remaining is a scheme that can be used to compute the semi-strong tree of any arbitrary subset family. Indeed, the union-difference property will only be used for eventually typing the nodes.

4.1 Computing a Factoring Permutation

Like modular decomposition, we use a partition refinement technique (see e.g. [14]) based on the notion of a *splitter*. There are two kinds of sesquimodule splitters:

- If there are $s \notin M$ and $x, y \in M$ with $(s, x) \in A$ (denoted by sx) and $(s, y) \notin A$ (denoted by $\overline{sy}$) then M is not a sesquimodule. We say that vertex s *splits* x and y.
- If there are $x, y \in M$ such that there are $s, t \notin M$ with $x|st$ and $\overline{y|st}$ then M is not a sesquimodule, where $x|st$ denotes $(xs \wedge xt) \vee (\overline{xs} \wedge \overline{xt})$ and $\overline{x|st}$ denotes its negation. We say that vertex pair (x, y) is a *self-splitter*.

Let us consider first category slitters. We begin with picking a vertex $x \in V$ and considering the *ordered* partition $\mathcal{P} = \{\overline{N^+(x)}, \{x\}, N^+(x)\}$, which will be seen as an ordered circular partition. We then perform a *round*, which consists of performing the refinement operation (see right below) for all vertex $y \neq x$ until this modifies the partition $\mathcal{P}$. The round ends and we restart a new one whenever $\mathcal{P}$ is modified. If, through some whole round, the partition $\mathcal{P}$ remains unchanged for all $y \neq x$, then the process stops.

The refinement operation w.r.t. a vertex $y \neq x$ consists of splitting each part P of the partition $\mathcal{P}$ into two parts: $P^+ = P \cap N^+(y)$ and $\overline{P^+} = P \setminus N^+(y)$. The reorganisation of the split parts is as follows. Though obviously the part $Q = \{x\}$ of the partition $\mathcal{P}$ is not split, we have to consider whether $x \in N^+(y)$. Let P and R be the neighbour parts of Q in $\mathcal{P}$: $\mathcal{P} = (\dots, P, Q, R, \dots)$. If $x \in N^+(y)$, replace P with $(\overline{P^+}, P^+)$ and replace R with $(R^+, \overline{R^+})$, else, replace P with $(P^+, \overline{P^+})$ and R with $(\overline{R^+}, R^+)$. If there are some emptyset, we act as if they were present, but skip storing them to the partition list. Thus, the elements that y "sees" the same way as how y "sees" x are locally stick together. We do the same processing for the parts before P and those after R in $\mathcal{P}$ (elements of same vision by y are locally stick together). That there is in the initial partition an odd number of parts – actually 3 – guarantees no conflict when closing the circle.

At this point, we obtain a partition $\mathcal{P}$ such that for all vertex v, and for all part P of the partition $\mathcal{P}$, v is not a first category splitter of P. Since each refinement can be done in $O(|N^+(y)|)$ time (see [14]), a round takes $O(n^2)$ time, where $n = |V|$. Since each round decreases the partition $\mathcal{P}$ to a thinner partition, there are at most n rounds. The total time is in $O(n^3)$.

We now consider second category slitters, with the computed partition $\mathcal{P}$. While there is in $\mathcal{P}$ some part P containing a *self-splitter* $(x, y) \subseteq P$, replace P with (P_x, P_y), which is defined as follows. First, push x in P_x and y in P_y. Let $s, t \notin P$ such that $x|st$ and $\overline{y|st}$. Then, for every other vertex z of P, either $z|st$ or $\overline{z|st}$, and we push z in P_x or P_y accordingly. Testing for *self-splitters* can be done by just testing all vertex quadruplets. This would globally cost $O(n^5)$ time.

At the end, we obtain a circularly ordered partition $\mathcal{P} = (P_1, \dots, P_k)$ of unordered parts P_i's. Then, ordering arbitrary the P_i's results in a circular permutation of V, which is a factoring permutation.

4.2   Computing the Decomposition Tree

We here constantly need to test if a subset is member of the initial family. Let τ denotes the time for such a test. For digraphs, given a vertex subset, we can test in $\tau = O(n^4)$ time if the subset is member of the sesquimodule family by checking

every vertex quadruplets. Then, the shape of the sesquimodular decomposition tree can easily be constructed in a brute-force manner as follows. Compute a factoring permutation. For each interval of the factoring permutation, test if it is a member of the initial family. For each pair of the latter members, if they cross, then remove both. Represent the remaining members in a cross-free tree-representation as explained at the beginning of Section 2. Since there are at most n^2 intervals in a circular permutation, these operations take $O(n^5 + n^2\tau)$ time.

The only remaining thing is to type the nodes. The main difficulty is how to test for nodes that are not simply-linked. Actually, we avoid this test by elimination of cases. For each internal node i of the decomposition tree:

Compute the $2-$graph of the quotient w.r.t. node i (quadratic number of tests for membership). If this is a clique, a path, or a cycle, conclude accordingly, and stop. Compute all quasi-trivial members of the quotient. If there are more than one or none of such, report a prime node, and stop. Else either the node is prime or it is not simply-linked with that unique quasi-trivial member which is strong. Let C be the complement of the unique quasi-trivial member. Assume node i is not simply-linked and recursively compute the decomposition tree of the quotient excluding C (refer to Section 2.2 for details). If the latter tree is anything except a single prime node then node i effectively was not simply-linked, we conclude and stop. The latter tree is a single prime node. If there is some quasi-trivial member therein then node i effectively was not simply-linked, we conclude and stop. Otherwise node i was simply-linked. We report a prime node.

Without recursive calls the process is in $O(n^3\tau) = O(n^7)$ time. Then, an inductive argument similar to the proof of Theorem 2 gives an $O(n^8)$ time bound.

Theorem 4. *The sesquimodular decomposition tree of a given digraph $G = (V, A)$ can be computed in $O(|V|^8)$ time.*

5   Conclusion and Perspectives

We have shown that union-difference families can be represented via a unique tree, and applied this result to a new directed graph decomposition. Of course the polynomial decomposition algorithm proposed here for this variation of modular decomposition has to be improved for a practical use. Another interesting investigation could be on the properties of the family of complements of members of a union-difference family. Such a family owns a quadratic representation straight from the result of union-difference families. However, their intrinsic properties are unclear, as the closure under difference does not behave symmetrically via complementary. Besides, representing families satisfying a number of closure operations remains an interesting question, and we are convinced that some other combinatorial decompositions can be expressed in this framework.

References

1. C. Capelle and M. Habib. Graph decompositions and factorizing permutations. In *5th Israel Symposium on Theory of Computing and Systems (ISTCS'97)*, IEEE Computer Society, pages 132–143, 1997.
2. M. Chein, M. Habib, and M.C. Maurer. Partitive hypergraphs. *Discrete Mathematics*, 37(1):35–50, 1981.
3. D. A. Cohen, M. C. Cooper, and P. G. Jeavons. Generalising submodularity and Horn clauses: Tractable optimization problems defined by tournament pair multimorphisms. Technical Report CS-RR-06-06, Oxford University, 2006.
4. C. Crespelle. *Représentations dynamiques de graphes*. PhD thesis, Université Montpellier II, 2007.
5. W. Cunningham. Decomposition of directed graphs. *SIAM Journal on Algebraic and Discrete Methods*, 3:214–228, 1982.
6. W. Cunningham and J. Edmonds. A combinatorial decomposition theory. *Canadian Journal of Mathematics*, 32:734–765, 1980.
7. J. Edmonds and R. Giles. A min-max relation for submodular functions on graphs. *Annals of Discrete Mathematics*, 1:185–204, 1977.
8. A. Ehrenfeucht, T. Harju, and G. Rozenberg. *The Theory of 2-Structures- A Framework for Decomposition and Transformation of Graphs*. World Scientific, 1999.
9. M. G. Everett and S. P. Borgatti. Regular Equivalence: General Theory. *Journal of Mathematical Sociology*, 18:29–52, 1994.
10. J. Fiala and D. Paulusma. A complete complexity classification of the role assignment problem. *Theoretical Computer Science*, 349(1):67–81, 2005.
11. H. Gabow. Centroids, Representations, and Submodular Flows. *Journal of Algorithms*, 18(3):586–628, 1995.
12. J. Gagneur, R. Krause, T. Bouwmeester, and G. Casari. Modular decomposition of protein-protein interaction networks. *Genome Biology*, 5(8), 2004.
13. Tibor Gallai. Transitiv orientierbare Graphen. *Acta Mathematica Academiae Scientiarum Hungaricae*, 18:25–66, 1967.
14. M. Habib, C. Paul, and L. Viennot. Partition refinement techniques: An interesting algorithmic tool kit. *International Journal of Foundations of Computer Science*, 10(2):147–170, 1999.
15. W.-L. Hsu and R. M. McConnell. PC-trees and circular-ones arrangements. *Theoretical Computer Science*, 296:99–116, 2003.
16. J.-M. Lanlignel. *Autour de la décomposition en coupes*. PhD thesis, Université Montpellier II, 2001.
17. R.H. Möhring and F.J. Radermacher. Substitution decomposition for discrete structures and connections with combinatorial optimization. *Annals of Discrete Mathematics*, 19:257–356, 1984.
18. F. de Mongolfier and M. Rao. The bi-join decomposition. In *7th International Colloquium on Graph Theory (ICGT'05)*, 2005.
19. A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Springer-Verlag, 2003.
20. C. Semple and M. Steel. *Phylogenetics*. Oxford Lecture Series in Mathematics and Its Applications, number 24, 2003.
21. D. R. White and K. P. Reitz. Graph and Semigroup Homomorphisms on Networks of Relations. *Social Networks*, 5:193–234, 1983.

Appendix:

Proof of* Lemma 1:*** The proof given in [8] is as follows. Suppose that $G_{\mathcal{F}}$ has a vertex x with degree at least 3, and let y, z, t be three distinct neighbours of x. In other words, $\{x, y\}$ and $\{x, z\}$ are members of $\mathcal{F}$, and so is $\{x, y, z\}$ by union closure. But $\{x, t\}$ is also a member of $\mathcal{F}$. By difference closure we deduce that $\{y, z\}$ is an edge of $G_{\mathcal{F}}$. Likewise, we can deduce that x, y, z, and t form a clique in $G_{\mathcal{F}}$. Now, let v be a vertex that is connected to the previous clique at some point, say t. Then, by a similar argument on the fact that t is of degree at least 3, we can show that v is connected to all other vertices of the clique. Thus the previous clique plus vertex v form a bigger clique, and so on. The connectivity of $G_{\mathcal{F}}$ then can be used to conclude that the whole graph $G_{\mathcal{F}}$ is a clique. Finally, the only connected graphs of degree at most 2 are paths and cycles.   □

Example of sesquimodular decomposition:

(a) A directed graph, bold arcs denote all-to-all arcs.

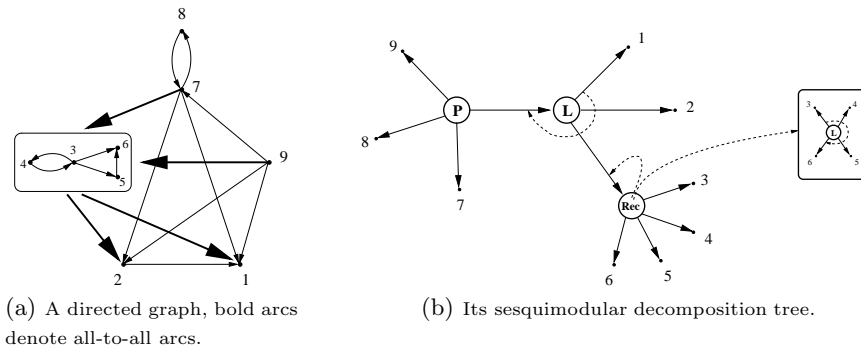(b) Its sesquimodular decomposition tree.

Fig. 4. Sesquimodular decomposition.

The family of non-trivial sesquimodules of the digraph of Fig. 4 is

$$\{\{1, 2\}, \{1, 2, 3, 4, 5, 6\}, \{2, 3, 4, 5, 6\}, \{2, 3, 4, 5, 6, 7, 8, 9\}, \{3, 4, 5, 6, 7, 8, 9\},$$
$$\{3, 4\}, \{3, 4, 5\}, \{3, 4, 5, 6\}, \{4, 5\}, \{4, 5, 6\}, \{5, 6\}\}$$