

Explaining the Lack of Locally Envy-Free Allocations

Aurélie Beynier^a, Jean-Guy Mailly^b, Nicolas Maudet^a and Anaëlle Wilczynski^c

^aLIP6, Sorbonne Université, CNRS, {aurelie.beynier,nicolas.maudet}@lip6.fr

^bIRIT, Université Toulouse Capitole, jean-guy.mailly@irit.fr

^cMICS, CentraleSupélec, Université Paris-Saclay, anaelle.wilczynski@centralesupelec.fr

Abstract. In fair division, local envy-freeness is a desirable property which has been thoroughly studied in recent years. In this paper, we study explanations which can be given to explain that no allocation of items can satisfy this criterion, in the house allocation setting where agents receive a single item. While Minimal Unsatisfiable Subsets (MUSes) are key concepts to extract explanations, they cannot be used as such: (i) they highly depend on the initial encoding of the problem; (ii) they are flat structures which fall short of capturing the dynamics of explanations; (iii) they typically come in large number and exhibit great diversity. In this paper we provide two SAT encodings of the problem which allow us to extract MUS when instances are unsatisfiable. We build a dynamic graph structure which allows to follow step-by-step the explanation. Finally, we propose several criteria to select MUSes, some of them being based on the MUS structure, while others rely on this original graphical explanation structure. We give theoretical bounds on these metrics, showing that they can vary significantly for some instances. Experimental results on synthetic data complement these results and illustrate the impact of the encodings and the relevance of our metrics to select among the many MUSes.

1 Introduction

The need to make AI systems accountable has gained momentum in recent years, with a staggering amount of contributions dedicated to explaining ML-based recommendations in particular [10]. In this landscape, explaining collective decisions has been relatively neglected, with a few notable exceptions, especially in voting [7, 9, 23]. This may come as a surprise as this is a context in which explanations are particularly valuable and challenging, as emphasized by Suryanarayana et al. [25]. Indeed, the outcome is likely to make some agents not fully satisfied, and mechanisms for collective decision typically articulate several criteria which require to be justified, even though axioms provide a good normative basis [24].

In this paper we tackle a problem of *fair division* [22] where agents are located on a graph, and the designer's objective is to allocate exactly one item to each agent in such a way that no agent envies a neighbor in this graph [3]. This is arguably one of the most basic settings of a problem studied in many contexts [8, 11, 16]. Our prime objective is to explain the fact that, for a given instance, no allocation can satisfy the desired criterion of *local envy-freeness*. As we will see, this setting is both simple and rich enough to unveil many facets of explanations in this context, providing a concrete example of the still scarce *explainable fair division* domain [15]. Recently, Zahedi et al. [26] proposed an approach allowing to counterfactually contest

an allocation in a context of incomplete information of other agents' preferences, seeing the problem as a sequential bargaining game.

Our approach will rely on the Boolean Satisfiability (SAT) modeling of the problem, and on the use of Minimally Unsatisfiable Subsets (MUSes) of clauses as natural candidates to exhibit concise certificates of unsatisfiability, following a well-established tradition of SAT or constraint-based formal explanations [2, 13, 14, 19]. While many approaches assume that MUSes are viable explanations *per se*, our contribution explores what it takes to turn them into proper explanations. While MUSes are precious intermediary steps towards explanations, they suffer from several issues: (i) they highly depend on the initial encoding of the problem, meaning that the objective of explanation must be integrated early in the modeling process; (ii) they are flat structures which fall short of capturing the dynamics of explanations, meaning that they must be translated to a representation more amenable to an interactive process [6]; (iii) they typically come in large number and exhibit great diversity, meaning that some criteria must be used to filter out the most convincing explanations.

In this paper, we provide two SAT encodings for deciding whether a locally envy-free allocation exists. These encodings allow us to extract MUSes (Section 3). While Answer Set Programming encodings for more general resource allocation problems have been proposed before [20], our encodings are dedicated to this setting and more easily amenable to explanations. In Section 4, we show how to build a dynamic graph structure from a MUS which allows to follow step-by-step the explanation. This is in line with other approaches which seek to provide step-by-step explanations for constraint satisfaction problems [6]. Finally, we propose in Section 5 several criteria to select MUSes, some of them being based on the MUS structure, while others rely on this original graphical explanation structure. We give theoretical bounds on these metrics, showing that they can vary significantly for some instances. Experimental results reported in Section 6 on synthetic data complement these results and illustrate the impact of the encodings and the relevance of our metrics to select among the many MUSes. Due to space restrictions, some proofs and experimental results are deferred to the technical report [4].

2 Preliminaries

For any integer k , let $[k]$ denote $\{1, \dots, k\}$. Let $N = [n]$ be a set of n agents and $O = \{o_1, o_2, \dots, o_n\}$ be a set of n indivisible objects (or items) that must be assigned to the agents. The agents express strict ordinal preferences over the objects, that is the preferences of each agent $i \in N$ are represented by a linear order \succ_i over O . The goal is to find an allocation $\sigma : N \rightarrow O$ such that $\{\sigma(i) \mid i \in N\} = O$

and $\sigma(i) \neq \sigma(j)$ for every pair of agents $i, j \in N$. We aim to fulfill the fairness criterion of *local envy-freeness* [3] which is based on the plausible envy that may occur between agents, because it takes into account the relations between the agents which are given by a social network represented by an undirected graph $G = (N, E)$.

Definition 1 (Local Envy-Freeness). *An allocation σ is locally envy-free (LEF) if for every pair of agents $i, j \in N$ such that $\{i, j\} \in E$, we have $\sigma(i) \succ_i \sigma(j)$.*

Note that a locally envy-free allocation does not always exist and deciding about its existence is NP-complete [3]. An instance of fair house allocation is given by $I = \langle N, O, (\succ_i)_{i \in N}, G = (N, E) \rangle$, and the set of all possible instances is denoted by \mathcal{I} . Let $\bar{\mathcal{I}}$ denote the set of all instances where no LEF allocations exist.

Approaches based on Boolean Satisfiability (SAT) propose to represent a set of constraints to satisfy as a Conjunctive Normal Form (CNF) formula, i.e., a set of clauses, and check whether this set admits at least one model (i.e., an interpretation which makes the formula true) [5]. A CNF is *unsatisfiable* when it does not admit any model. Given a CNF ϕ , a MUS is an unsatisfiable subset $\psi \subseteq \phi$ of clauses such that removing any clause from ψ makes it satisfiable. Computing a MUS is difficult in general, as it requires (several) calls to a SAT solver [21]. However, algorithms which are efficient in practice are available, like the algorithm implemented in the library that is used later in the experimental part of this paper [17]. In this article, we will use a SAT-based approach to explain the lack of LEF allocations.

3 SAT Formulation of Local Envy-Freeness

We first propose two variants of a SAT encoding such that, for a given instance I , the corresponding Boolean formula is satisfiable iff I is LEF, i.e., there exists an LEF allocation. Moreover the models of the formula correspond to LEF allocations. As far as we know, this is the first attempt at encoding LEF into SAT.

For our SAT encodings, we use the following Boolean allocation variables: $\forall i \in N, \forall o \in O, \sigma_{i,o} = \top$ iff o is allocated to i . The constraints that encode the specific structure of the allocation and the local envy-freeness requirement are defined as follows.

$$\phi_{alloc} = \bigwedge_{i \in N} \left(\phi_{alloc}^{\geq 1, N}(i) \wedge \bigwedge_{o \in O} \left(\bigwedge_{j \in N: j > i} \phi_{alloc}^{\leq 1, O}(o, i, j) \right) \right)$$

$$\text{with: } \phi_{alloc}^{\geq 1, N}(i) = \bigvee_{o \in O} \sigma_{i,o}, \quad \phi_{alloc}^{\leq 1, O}(o, i, j) = (\neg \sigma_{i,o} \vee \neg \sigma_{j,o})$$

$$\phi_{lef} = \bigwedge_{i \in N} \bigwedge_{j \in N: \{i, j\} \in E} \bigwedge_{o \in O} \phi_{lef}(i, j, o), \text{ where}$$

$$\phi_{lef}(i, j, o) = \neg \sigma_{i,o} \vee \bigvee_{o' \in O: o' \succ_j o \wedge o \succ_i o'} \sigma_{j,o'}$$

The $\phi_{lef}(\cdot, \cdot, \cdot)$ clauses are called *lef-clauses* while structural clauses $\phi_{alloc}^{\geq 1, N}(\cdot)$ and $\phi_{alloc}^{\leq 1, O}(\cdot, \cdot, \cdot)$ are called *at-least-one-per-agent* and *at-most-one-per-object* allocation clauses, respectively. In total, we have n at-least-one-per-agent clauses, $\frac{n^2(n-1)}{2}$ at-most-one-per-object clauses, and $2n|E|$ lef-clauses. Intuitively, a lef-clause $\phi_{lef}(i, j, o)$ means that, if agent i receives the object o , then her neighbor j must receive an object o' such that j prefers o' , and i prefers o .

Note that, in case $\{o' \in O : o' \succ_j o \wedge o \succ_i o'\} = \emptyset$ for given two agents i and j and object o , we have the associated lef-clause which is reduced to a single negative literal: $\phi_{lef}(i, j, o) = (\neg \sigma_{i,o})$.

Proposition 1. *There exists a locally envy-free allocation iff formula $\phi = \phi_{alloc} \wedge \phi_{lef}$ is satisfiable.*

Note that the structural allocation clauses in ϕ_{alloc} are sufficient to impose that exactly one object is assigned to every agent, because there are exactly as many objects as agents. However, one may add redundant clauses specifying that each object must be allocated at least once (*at-least-one-per-object* clauses) and no agent can be assigned more than one object (*at-most-one-per-agent* clauses), without hurting the encoding of LEF allocations.

$$\phi_{alloc}^+ = \phi_{alloc} \wedge \bigwedge_{o_j \in O} \phi_{alloc}^{\geq 1, O}(o_j) \wedge \bigwedge_{i \in N} \bigwedge_{k > j} \phi_{alloc}^{\leq 1, N}(i, o_j, o_k)$$

$$\text{with: } \phi_{alloc}^{\geq 1, O}(o) = \bigvee_{i \in N} \sigma_{i,o}, \quad \phi_{alloc}^{\leq 1, N}(i, o, o') = (\neg \sigma_{i,o} \vee \neg \sigma_{i,o'})$$

Corollary 2. *There exists a locally envy-free allocation iff formula $\phi^+ = \phi_{alloc}^+ \wedge \phi_{lef}$ is satisfiable.*

In general, clauses $\phi_{alloc}^{\geq 1, \cdot}$ are called *at-least* clauses and clauses $\phi_{alloc}^{\leq 1, \cdot}$ are called *at-most* clauses. The encodings based on formulas ϕ and ϕ^+ are called the *basic* and *redundant* encodings, respectively.

When an LEF allocation does not exist, we will particularly focus on subsets of clauses of ϕ or ϕ^+ that make the instance negative. In this respect, the redundant encoding ϕ^+ can sometimes be useful to derive more direct explanations. We first provide a basic observation on key clauses for unsatisfiability.

Observation 3. *Each unsatisfiable subset of clauses must contain at least one at-least clause and at least one lef-clause.*

4 Satisfiability Implication Graph

Our goal is to provide explanations for the non-existence of LEF allocations. For this purpose, we will construct a graph representation of unsatisfiable subsets of clauses.

4.1 Construction of the graph

Let us first analyze the structure of the clauses of formula ϕ^+ . Observe that the lef-clauses and at-least clauses are dual-Horn clauses, i.e., they contain at most one negative literal, thus they can be written as implications between a positive literal (or “true” for an at-least clause) and a disjunction of positive literals. The at-most clauses are goal clauses, i.e., they contain no positive literals, and thus they can be written as an implication between a conjunction of positive literals and “false”. It follows that we can generally rewrite all clauses of the LEF formula ϕ^+ as an implication between a (possibly empty) conjunction of positive literals and a (possibly empty) disjunction of positive literals. More precisely, the clauses of ϕ^+ can be rewritten as simple implications as follows:

$$\phi_{alloc}^{\geq 1, N}(i) = \top \rightarrow \bigvee_{o \in O} \sigma_{i,o}; \quad \phi_{alloc}^{\geq 1, O}(o) = \top \rightarrow \bigvee_{i \in N} \sigma_{i,o}$$

$$\phi_{alloc}^{\leq 1, O}(o, i, j) = (\sigma_{i,o} \wedge \sigma_{j,o}) \rightarrow \perp$$

$$\phi_{alloc}^{\leq 1, N}(i, o, o') = (\sigma_{i,o} \wedge \sigma_{i,o'}) \rightarrow \perp \tag{1}$$

$$\phi_{lef}(i, j, o) = \sigma_{i,o} \rightarrow \bigvee_{o' \in O: o' \succ_j o \wedge o \succ_i o'} \sigma_{j,o'}$$

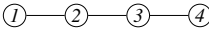
For a given subset of clauses ψ , we construct its *satisfiability implication graph* $H^\psi = (V^\psi, F^\psi)$, which is a directed bipartite graph where the nodes are partitioned into variable-nodes and clause-nodes, i.e., $V^\psi = Var^\psi \cup Cl^\psi$ with $Var^\psi := \{\top, \perp\} \cup \{\sigma_{i,o} :$

$\sigma_{i,o} \in \psi$ or $\neg\sigma_{i,o} \in \psi$ and $Cl^\psi := \psi$, and F^ψ denotes the set of arcs between variable-nodes and clause-nodes. The set of arcs is such that there is an arc in F^ψ from a variable-node x to a clause-node y iff x corresponds to a variable which is part of the implicant conjunction of the clause (as formulated in (1)) related to y , and an arc from a clause-node y to a variable-node z iff z corresponds to a variable which is part of the implied disjunction of the clause related to y . More precisely, we have the following arcs in F^ψ :

- for every at-least clause $\phi_{alloc}^{\geq 1,Y}(x) \in \psi$ where $Y \in \{N, O\}$, we have the arcs $(\top, \phi_{alloc}^{\geq 1,Y}(x))$ and:
 - $(\phi_{alloc}^{\geq 1,Y}(x), \sigma_{x,o})$ for every $o \in O$ if $Y = N$, or
 - $(\phi_{alloc}^{\geq 1,Y}(x), \sigma_{i,x})$ for every $i \in N$ if $Y = O$;
- for every at-most clause $\phi_{alloc}^{\leq 1,Y}(x, y, z) \in \psi$, for $Y \in \{N, O\}$, we have the arcs $(\phi_{alloc}^{\leq 1,Y}(x, y, z), \perp)$ and:
 - $(\sigma_{y,x}, \phi_{alloc}^{\leq 1,Y}(x, y, z))$ and $(\sigma_{z,x}, \phi_{alloc}^{\leq 1,Y}(x, y, z))$ if $Y = O$,
 - $(\sigma_{x,y}, \phi_{alloc}^{\leq 1,Y}(x, y, z))$ and $(\sigma_{x,z}, \phi_{alloc}^{\leq 1,Y}(x, y, z))$ if $Y = N$;
- for every left-clause $\phi_{lef}(i, j, o) \in \psi$, we have the arcs $(\sigma_{i,o}, \phi_{lef}(i, j, o))$ and $(\phi_{lef}(i, j, o), \sigma_{j,o'})$ for all $o' \in O$ s.t. $o' \succ_j o$ and $o \succ_i o'$, or $(\phi_{lef}(i, j, o), \perp)$ if there is no such o' .

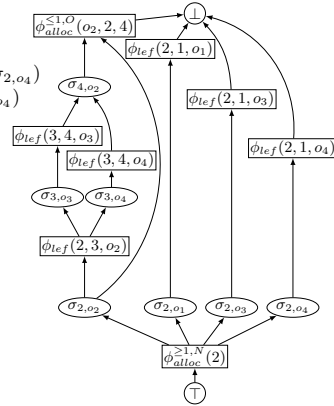
The construction of the satisfiability implication graph is illustrated in the next example.

Example 1. Let us consider the following instance with four agents.

1:	o_1	\succ	o_4	\succ	o_3	\succ	o_2	
2:	o_1	\succ	o_2	\succ	o_4	\succ	o_3	
3:	o_3	\succ	o_4	\succ	o_1	\succ	o_2	
4:	o_2	\succ	o_3	\succ	o_4	\succ	o_1	

We consider the subset of clauses ψ given below on the left. The associated satisfiability implication graph H^ψ can be constructed as shown below on the right, where variable-nodes are represented with circles and clause-nodes with rectangles.

$$\begin{aligned}
 \phi_{alloc}^{\geq 1,N}(2) &= (\sigma_{2,o_1} \vee \sigma_{2,o_2} \vee \sigma_{2,o_3} \vee \sigma_{2,o_4}) \\
 \phi_{lef}(2, 3, o_2) &= (\neg\sigma_{2,o_2} \vee \sigma_{3,o_3} \vee \sigma_{3,o_4}) \\
 \phi_{lef}(3, 4, o_3) &= (\neg\sigma_{3,o_3} \vee \sigma_{4,o_2}) \\
 \phi_{lef}(3, 4, o_4) &= (\neg\sigma_{3,o_4} \vee \sigma_{4,o_2}) \\
 \phi_{alloc}^{\leq 1,O}(o_2, 2, 4) &= (\neg\sigma_{2,o_2} \vee \neg\sigma_{4,o_2}) \\
 \phi_{lef}(2, 1, o_1) &= (\neg\sigma_{2,o_1}) \\
 \phi_{lef}(2, 1, o_3) &= (\neg\sigma_{2,o_3}) \\
 \phi_{lef}(2, 1, o_4) &= (\neg\sigma_{2,o_4})
 \end{aligned}$$



4.2 Dynamic activation process

A satisfiability implication graph H^ψ is associated with an *activation function* $v : V^\psi \rightarrow \{0, 1\}$. A node x is said to be activated iff $v(x) = 1$. The interpretation of the activation differs depending on the types of the node:

- an activated variable-node means that we set the corresponding variable to true, and
- an activated clause-node means that the implicant part of the clause (as formulated in (1)) is true and thus the implied part must be true too.

We represent a dynamic activation process on the satisfiability implication graph by considering an initial activation state v^0 where only the variable-node \top is activated, and then recursively defining new successor activation states as described below. An activation state v' is a successor of activation state v iff $v' \neq v$ and:

- if $v(x) = 1$, then $v'(x) = 1$, for every node $x \in V^\psi$,
- if x is a clause-node and all predecessors of x are activated in v , then x becomes activated in v' ,
- if x is a clause-node and becomes activated in v , then one of its successor variable-nodes y is chosen to be activated in v' .

The set of all possible successor activation states of an activation state v is denoted by $succ^\psi(v)$, and all activated nodes in a given activation state v is denoted by $activ^\psi(v)$, i.e., $activ^\psi(v) := \{x \in V^\psi : v(x) = 1\}$. By definition, between one activation state v and one of its successor states $v' \in succ^\psi(v)$, we have $activ^\psi(v) \subsetneq activ^\psi(v')$. We create new successor activation states as long as it is possible, but then, by strict monotony, there necessarily exist activation states v which are final, i.e., $succ^\psi(v) = \emptyset$. Let S^ψ denote the set of all activation states.¹ Let an *activation path* define a path $\langle v^0, v^1, \dots, v^T \rangle$, where for each $t \in [T]$, $v^t \in succ^\psi(v^{t-1})$ and $succ^\psi(v^T) = \emptyset$, i.e., v^T is a final activation state. Let us denote by \mathcal{P}^ψ the set of all possible activation paths. The size $|\mathbf{v}|$ of an activation path $\mathbf{v} = \langle v^0, v^1, \dots, v^T \rangle \in \mathcal{P}^\psi$ is equal to T .

The dynamic activation process on the satisfiability implication graph H^ψ associated with a subset of clauses ψ enables to characterize the satisfiability of ψ , as stated below.

Theorem 4. A subset of clauses ψ is unsatisfiable iff every activation path $\langle v^0, v^1, \dots, v^T \rangle \in \mathcal{P}^\psi$ eventually activates the node \perp , i.e., $\perp \in activ^\psi(v^T)$.

Sketch of proof. Suppose that there exists an activation path $\mathbf{v} = \langle v^0, v^1, \dots, v^T \rangle$ in \mathcal{P}^ψ such that $\perp \notin activ^\psi(v^T)$. We construct the corresponding truth assignment φ of the variables of ψ , i.e., φ sets to true all variables associated with variable-nodes which are activated in v^T and sets to false the remaining variables of ψ . One can prove that φ satisfies all the clauses of ψ and thus ψ is satisfiable.

Suppose now that the set of clauses ψ is satisfiable, i.e., there exists a truth assignment φ of the variables of ψ such that all clauses of ψ are satisfied. Let us construct a specific activation path $\mathbf{v} = \langle v^0, v^1, \dots, v^T \rangle$ where only node \top is activated in v^0 and, for every $t \in [T]$, we choose $v^t \in succ^\psi(v^{t-1})$ in such a way that only variable-nodes associated with true positive literals in φ are activated (following this definition, \perp cannot be activated). One can prove that the constructed path \mathbf{v} is indeed a valid activation path, in the sense that v^T is a final activation state, i.e., $succ^\psi(v^T) = \emptyset$. \square

4.3 Textual explanation from the graph activations

From the satisfiability implication graph and its activation states, we can thus deduce a formal proof to derive an explanation for the non-existence of an LEF allocation. This explanation can be constructed by performing a depth-first search over activation states starting from v^0 and each time printing the meaning of each new activated node, i.e., by calling $Expl_{\mathcal{P}^\psi}(v^0)$, where $Expl_{\mathcal{P}^\psi}(v)$ is defined as described in Algorithm 1, and $semantics(x)$ refers to the textual meaning of each node x as defined in Table 1.

Let us compute a textual explanation for our running example.

¹ By definition, S^ψ only contains the initial activation state v^0 and activation states which are successors from previously created activation states.

Table 1. Description of the semantics of the nodes of graph H^ψ

Node x	$semantics(x)$
$\sigma_{i,o}$	“Suppose that object o is assigned to agent i .”
\perp	“Contradiction.”
$\phi_{alloc}^{\geq 1,N}(i)$	“Agent i must get at least one object.”
$\phi_{alloc}^{\geq 1,O}(o)$	“Object o must be assigned to at least one agent.”
$\phi_{alloc}^{\leq 1,O}(o, i, j)$	“Object o cannot be assigned to both agents i and j .”
$\phi_{alloc}^{\leq 1,N}(i, o, o')$	“Agent i cannot get both objects o and o' .”
$\phi_{lef}(i, j, o)$	“If agent i gets object o then, to avoid local envy, her neighbor j must be assigned to an object that agent j prefers to o and that agent i likes less than o , i.e., one object among:” $\{o' \in O : o' \succ_j o \wedge o \succ_i o'\}$.

Algorithm 1: $Expl_{\mathcal{P}\psi}(v)$

Input: Activation state v

1 **if** $\perp \in activ^\psi(v)$ **then return;**

2 **foreach** $v' \in succ^\psi(v)$ **do**

3 **foreach** $x \in activ^\psi(v') \setminus activ^\psi(v)$ **do**

4 print($semantics(x)$);

5 $Expl_{\mathcal{P}\psi}(v')$;

Example 1 (continued). Let us run Algorithm 1 with the call $Expl_{\mathcal{P}\psi}(v^0)$ on the instance given in Example 1. We detail in Figure 1 the output of the algorithm by mentioning the specific calls with the activation states indexed w.r.t. the depth-first search traversal.

5 Minimal Explanation based on MUS

In this section, we will try to derive minimal explanations for negative instances (when no LEF allocations exist), based on MUSes. We denote by $\mathcal{M}(I)$ (resp., $\mathcal{M}^+(I)$) the set of all MUSes of ϕ (resp., ϕ^+) for instance $I \in \mathcal{I}$. Basically, $\mathcal{M}(I) \subseteq \mathcal{M}^+(I)$, and $\mathcal{M}(I) \neq \emptyset$ iff $\mathcal{M}^+(I) \neq \emptyset$ iff $I \in \bar{\mathcal{I}}$. We first observe that the satisfiability implication graph associated with a MUS exhibits a particular structure.

Proposition 5. *If a set of clauses ψ is a MUS, then the satisfiability implication graph H^ψ is connected and contains exactly one source, namely the node \top , and exactly one sink, namely the node \perp .*

This proposition holds thanks to the property of minimality by inclusion of the MUS. Indeed, the graph associated with an arbitrary unsatisfiable set of clauses may not satisfy any of the two properties.

5.1 Complexity of an LEF explanation

Even if a MUS provides a *minimal* unsatisfiable subset of clauses, it does not necessarily give the shortest or the most understandable explanation. We will analyze several metrics to measure the complexity of an explanation based on a MUS. For a negative instance $I \in \bar{\mathcal{I}}$, a metric m applied on I is a function $m_I : \mathcal{M}^+(I) \rightarrow \mathbb{R}$ to minimize.

Let us first provide below examples of canonical explanations based on specific MUSes for two particular instances.

Example 2. Consider an instance where two neighbors i and j share the same preferences. A possible MUS contains the at-least-one-per-agent clause $\phi_{alloc}^{\geq 1,N}(i)$ and all n lef-clauses $\phi_{lef}(i, j, o) = (\neg\sigma_{i,o})$ for all $o \in O$.

Example 3. Consider an instance where the same item o is ranked last by all agents, and no agent is isolated in the social network. A possible MUS contains the at-least-one-per-object clause $\phi_{alloc}^{\geq 1,O}(o)$ and all n lef-clauses $\phi_{lef}(i, j, o) = (\neg\sigma_{i,o})$ for all $i \in N$, and j some neighbor of i .

We introduce the notion of metric gap to measure how much the complexity of an explanation can increase if we choose the “worst” MUS from which to derive an explanation.

Definition 2 (Metric Gap). *For a given metric m , the metric gap of m is defined as the worst ratio over all negative instances between the worst value of m on a MUS and the best one, i.e.:*

$$MG(m) := \max_{I \in \bar{\mathcal{I}}} \max_{\psi, \psi^* \in \mathcal{M}^+(I)} \frac{m_I(\psi)}{m_I(\psi^*)}$$

The following example will be useful to derive metric gaps because it exhibits MUSes of very different complexity.

Example 4. Consider an instance with an even number n of agents where the social network $G = (N, E)$ is a matching with one additional edge, i.e., $E = \{\{i, i+1\} : i \in \{1, 3, \dots, n-1\}\} \cup \{1, 3\}$. Each agent $i \in N$ has the following preferences if i is odd: $o_1 \succ_i o_2 \succ_i \dots \succ_i o_{n-1} \succ_i o_n$, and the following preferences if i is even: $o_{n-1} \succ_i \dots \succ_i o_2 \succ_i o_1 \succ_i o_n$. The two connected agents 1 and 3 have the same preferences, therefore there is a MUS ψ^1 similar to the one given in Example 2. Moreover, since all agents have the same last object o_n , there is also a MUS ψ^2 similar to the one given in Example 3. Consider now the MUS ψ^3 composed of the following clauses: the at-least-one-per-agent clauses $\phi_{alloc}^{\geq 1,N}(i)$ for every agent $i \in N$, the lef clauses $\phi_{lef}(i, i+1, o_n) = (\neg\sigma_{i,o_n})$ for every $i \in \{1, 3, \dots, n-1\}$, the lef clauses $\phi_{lef}(i, i-1, o_n)$ for every $i \in \{2, 4, \dots, n\}$, and finally all at-most-one-per-object clauses $\phi_{alloc}^{\leq 1,O}(o_k, i, j)$ for every $k \in [n-1]$ and $i, j \in N$.

We will propose several natural metrics in the next two subsections and prove that their metric gap is unbounded, showing that we need to carefully choose the MUSes to derive explanations. For most of the metrics, the gap holds even if we restrict to the basic encoding.

5.1.1 Metrics based on the SAT formula

Some basic metrics can for instance be based on the number of objects or agents involved in an explanation.

Proposition 6. *The tight lower and upper bounds for the minimal number of agents (resp., items) involved in a MUS are 2 and n (resp., 1 and n), respectively. The metric gap for the number of agents (resp., items) is $\Theta(n)$.*

When restricting to the basic encoding, any MUS always involves all the n objects. Therefore, we need to consider the redundant encoding if the number of involved objects is a concern.

Since our explanation is based on a CNF formula, metrics counting the number of variables or clauses in ϕ^+ turn out to be very natural.

Proposition 7. *The tight lower and upper bounds for the number of variables involved in a MUS are n and n^2 , respectively. The metric gap for the number of variables is $\Theta(n)$.*

Proof. By Observation 3, each MUS contains at least one at-least clause associated with an agent $i \in N$ (resp., an object $o \in O$), which involves the n variables $\sigma_{i,o}$ for all n objects $o \in O$ (resp., all

$[\mathbf{Expl}(v^0)]$ Agent 2 must get at least one object:

- $[\mathbf{Expl}(v^1)]$ Suppose that object o_2 is assigned to agent 2. $[\mathbf{Expl}(v^2)]$ If agent 2 gets object o_2 then, to avoid local envy, her neighbor 3 must be assigned to an object that agent 3 prefers to o_2 and that agent 2 likes less than o_2 , i.e., one object among: $\{o_3, o_4\}$
 - $[\mathbf{Expl}(v^3)]$ Suppose that object o_3 is assigned to agent 3. $[\mathbf{Expl}(v^4)]$ If agent 3 gets object o_3 then, to avoid local envy, her neighbor 4 must be assigned to an object that agent 4 prefers to o_3 and that agent 3 likes less than o_3 , i.e., one object among: $\{o_2\}$. $[\mathbf{Expl}(v^5)]$ Suppose that object o_2 is assigned to agent 4. $[\mathbf{Expl}(v^6)]$ Object o_2 cannot be assigned to both agents 2 and 4. $[\mathbf{Expl}(v^7)]$ Contradiction.
 - $[\mathbf{Expl}(v^3)]$ Suppose that object o_4 is assigned to agent 3. $[\mathbf{Expl}(v^8)]$ If agent 3 gets object o_4 then, to avoid local envy, her neighbor 4 must be assigned to an object that agent 4 prefers to o_4 and that agent 3 likes less than o_4 , i.e., one object among: $\{o_2\}$. $[\mathbf{Expl}(v^9)]$ Suppose that object o_2 is assigned to agent 4. $[\mathbf{Expl}(v^{10})]$ Object o_2 cannot be assigned to both agents 2 and 4. $[\mathbf{Expl}(v^{11})]$ Contradiction.
- $[\mathbf{Expl}(v^1)]$ Suppose that object o_1 is assigned to agent 2. $[\mathbf{Expl}(v^{12})]$ If agent 2 gets object o_1 then, to avoid local envy, her neighbor 1 must be assigned to an object that agent 1 prefers to o_1 and that agent 2 likes less than o_1 , i.e., one object among: \emptyset . $[\mathbf{Expl}(v^{13})]$ Contradiction.
- $[\mathbf{Expl}(v^1)]$ Suppose that object o_3 is assigned to agent 2. $[\mathbf{Expl}(v^{14})]$ If agent 2 gets object o_3 then, to avoid local envy, her neighbor 1 must be assigned to an object that agent 1 prefers to o_3 and that agent 2 likes less than o_3 , i.e., one object among: \emptyset . $[\mathbf{Expl}(v^{15})]$ Contradiction.
- $[\mathbf{Expl}(v^1)]$ Suppose that object o_4 is assigned to agent 2. $[\mathbf{Expl}(v^{16})]$ If agent 2 gets object o_4 then, to avoid local envy, her neighbor 1 must be assigned to an object that agent 1 prefers to o_4 and that agent 2 likes less than o_4 , i.e., one object among: \emptyset . $[\mathbf{Expl}(v^{17})]$ Contradiction.

Figure 1. Textual explanation derived from Example 1.

n agents $i \in N$). The total number of variables n^2 is a trivial upper bound for the number of variables involved in a MUS. Both bounds are tight by the instance given in Example 4 where the MUS ψ^1 is a canonical one with n variables while the MUS ψ^3 contains all n^2 variables. We can thus derive the metric gap, which is equal to n . \square

Proposition 8. *The tight lower bound for the number of clauses in a MUS is $n + 1$. The metric gap for the number of clauses is $\Omega(n^2)$.*

Sketch of proof. One can prove that any MUS contains at least $n + 1$ clauses. To derive the metric gap, consider the instance given in Example 4 where the MUS ψ^1 is a canonical one with $n + 1$ clauses, while the MUS ψ^3 contains n at-least clauses, n left-clauses, and $\frac{n(n-1)}{2}(n-1)$ at-most clauses, for a total of $\Theta(n^3)$ clauses. Therefore, we can deduce that the metric gap is $\Omega(n^2)$. \square

5.1.2 Metrics based on the satisfiability implication graph

We will now focus on metrics based on the satisfiability implication graph associated with a MUS. In fact, an explanation for the non-existence of an LEF allocation in an instance $I \in \bar{\mathcal{I}}$ performs a depth-first search over all possible activation states associated with the satisfiability graph H^ψ for a given MUS $\psi \in \mathcal{M}^+(I)$. Therefore, some basic metrics can be defined according to the structure of explanation paths in \mathcal{P}^ψ .

Definition 3 (Explanation length). *For a given MUS ψ , the length L^ψ of the explanation is its total duration, i.e., the running time of Algorithm 1, which is given by the total number of activation states, i.e., $L^\psi = |\mathcal{S}^\psi|$.*

In our context, an explanation is a proof where some disjunction cases are necessary, see, e.g., the at-least clauses. These disjunction cases are made explicit by all possible different successors of a given activation state. For a given case, it can be important for the understandability of the explanation to conclude quickly to a contradiction. The depth of the explanation captures this idea by considering the longest case to develop within the proof to reach a contradiction, which corresponds to the longest activation path.

Definition 4 (Explanation depth). *For a given MUS ψ , the depth d^ψ of the explanation is the maximum duration of a possible activation path, i.e., $d^\psi = \max_{\mathbf{v} \in \mathcal{P}^\psi} |\mathbf{v}|$.²*

In a similar vein, the proof can be more difficult to follow if there are many disjunction cases to develop. The breadth of the explanation counts the number of disjunction cases to develop within the proof.

Definition 5 (Explanation breadth). *For a given MUS ψ , the breadth B^ψ of the explanation is the total number of possible activation paths, i.e., $B^\psi = |\mathcal{P}^\psi|$.*

The different metrics are illustrated in the next example.

Example 5. *Let us consider the following instance with three agents.*

1:	o_1	\succ	o_2	\succ	o_3	
2:	o_1	\succ	o_2	\succ	o_3	
3:	o_2	\succ	o_3	\succ	o_1	

The four following subsets of clauses, ψ^1 , ψ^2 , ψ^3 and ψ^4 , are MUSes of ϕ^+ , where only the first three ones are MUSes of ϕ .

ψ^1		ψ^2	
$\phi_{alloc}^{\geq 1, N}(1) = (\sigma_{1,o_1} \vee \sigma_{1,o_2} \vee \sigma_{1,o_3})$		$\phi_{alloc}^{\geq 1, N}(3) = (\sigma_{3,o_1} \vee \sigma_{3,o_2} \vee \sigma_{3,o_3})$	
$\phi_{alloc}^{\leq 1, N}(2) = (\sigma_{2,o_1} \vee \sigma_{2,o_2} \vee \sigma_{2,o_3})$		$\phi_{lef}(3, 1, o_1) = (\neg\sigma_{3,o_1})$	
$\phi_{lef}(1, 3, o_2) = (\neg\sigma_{1,o_2})$		$\phi_{lef}(3, 1, o_2) = (\neg\sigma_{3,o_2} \vee \sigma_{1,o_1})$	
$\phi_{lef}(1, 3, o_3) = (\neg\sigma_{1,o_3})$		$\phi_{lef}(3, 2, o_2) = (\neg\sigma_{3,o_2} \vee \sigma_{2,o_1})$	
$\phi_{lef}(2, 3, o_2) = (\neg\sigma_{2,o_2})$		$\phi_{lef}(3, 1, o_3) = (\neg\sigma_{3,o_3} \vee \sigma_{1,o_1})$	
$\phi_{lef}(2, 3, o_3) = (\neg\sigma_{2,o_3})$		$\phi_{lef}(3, 2, o_3) = (\neg\sigma_{3,o_3} \vee \sigma_{2,o_1})$	
$\phi_{alloc}^{\leq 1, O}(o_1, 1, 2) = (\neg\sigma_{1,o_1} \vee \neg\sigma_{2,o_1})$		$\phi_{alloc}^{\leq 1, O}(o_1, 1, 2) = (\neg\sigma_{1,o_1} \vee \neg\sigma_{2,o_1})$	
ψ^3		ψ^4	
$\phi_{alloc}^{\geq 1, N}(1) = (\sigma_{1,o_1} \vee \sigma_{1,o_2} \vee \sigma_{1,o_3})$		$\phi_{alloc}^{\geq 1, O}(o_3) = (\sigma_{1,o_3} \vee \sigma_{2,o_3} \vee \sigma_{3,o_3})$	
$\phi_{lef}(1, 3, o_2) = (\neg\sigma_{1,o_2})$		$\phi_{lef}(1, 3, o_3) = (\neg\sigma_{1,o_3})$	
$\phi_{lef}(1, 3, o_3) = (\neg\sigma_{1,o_3})$		$\phi_{lef}(2, 3, o_3) = (\neg\sigma_{2,o_3})$	
$\phi_{lef}(1, 3, o_1) = (\neg\sigma_{1,o_1} \vee \sigma_{3,o_2} \vee \sigma_{3,o_3})$		$\phi_{lef}(3, 1, o_3) = (\neg\sigma_{3,o_3} \vee \sigma_{1,o_1})$	
$\phi_{lef}(3, 2, o_2) = (\neg\sigma_{3,o_2} \vee \sigma_{2,o_1})$		$\phi_{lef}(3, 2, o_3) = (\neg\sigma_{3,o_3} \vee \sigma_{2,o_1})$	
$\phi_{lef}(3, 2, o_3) = (\neg\sigma_{3,o_3} \vee \sigma_{2,o_1})$		$\phi_{alloc}^{\leq 1, O}(o_1, 1, 2) = (\neg\sigma_{1,o_1} \vee \neg\sigma_{2,o_1})$	
$\phi_{alloc}^{\leq 1, O}(o_1, 1, 2) = (\neg\sigma_{1,o_1} \vee \neg\sigma_{2,o_1})$			

The satisfiability implication graphs for each of the MUSes are given in Figure 2, in the order of presentation of the MUSes.

The following table collects the evaluation of the four MUSes on the different proposed metrics.

² The depth is also the size of the longest path without repetition from \top to \perp in H^ψ . However, since directed cycles can occur in satisfiability implication graphs, the definition based on activation paths is more appropriate.

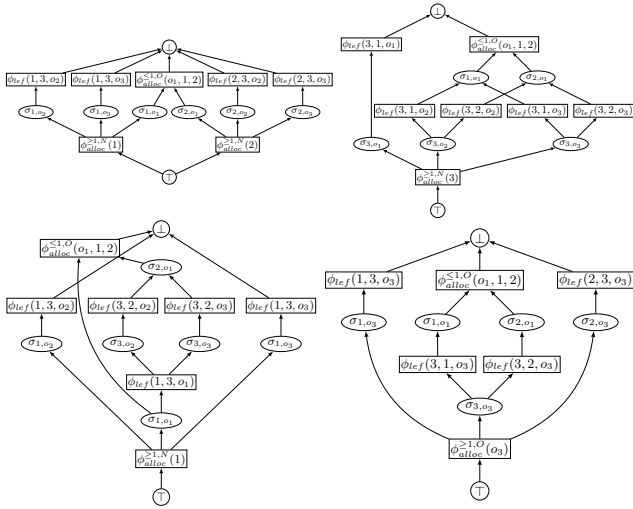


Figure 2. Satisfiability implication graphs for the MUSes of Example 5

	# clauses	# variables	# agents	length	depth	breadth
ψ^1	7	6	3	28	4	9
ψ^2	7	5	3	14	6	3
ψ^3	7	6	3	19	8	4
ψ^4	6	5	3	12	6	3

We can observe that ψ^2 provides a smaller (or as small) explanation than ψ^3 on all our metrics, and ψ^4 provides a smaller (or as small) explanation than ψ^2 on all our metrics, showing the interest of the redundant encoding ϕ^+ for possibly smaller explanations.

Proposition 9. The tight lower bounds for the length, the depth, and the breadth of a MUS are $3n + 1$, 4, and n , respectively.

Proposition 10. The metric gap for the depth is $\Omega(n)$.

Proof. Consider an instance with n agents where the social network $G = (N, E)$ is a circle around all agents, i.e., $E = \{\{i, i + 1\} : i \in [n - 1]\} \cup \{1, n\}$. Each agent $i \in [n - 1]$ has the following preferences: $o_i \succ_i o_{i-1} \succ_i o_1 \succ_i o_2 \succ_i \dots \succ_i o_n$, while agent n has the following preferences: $o_1 \succ_n o_{n-1} \succ_n o_2 \succ_n \dots \succ_n o_n$. Consider the MUS ψ^1 composed of the following clauses: $\phi_{alloc}^{>=1,N}(1)$, $\phi_{def}(1, n, o_1) = (\neg\sigma_{1,o_1})$, and the $n - 1$ clauses $\phi_{def}(1, 2, o_i) = (\neg\sigma_{1,o_i})$ for $i \in \{2, \dots, n\}$. This minimum MUS has a depth equal to $d^{\psi^1} = 4$. However, there is another MUS ψ^2 which is the same as ψ^1 except that clause $\phi_{def}(1, n, o_1)$ is replaced by the following subset of clauses: $\phi_{def}(i, i + 1, o_i) = (\neg\sigma_{i,o_i} \vee \sigma_{i+1,o_{i+1}})$, for every $i \in [n - 2]$, $\phi_{def}(n - 1, n, o_{n-1}) = (\neg\sigma_{n-1,o_{n-1}} \vee \sigma_{n,o_1})$, and $\phi_{alloc}^{<=1,O}(o_1, 1, n) = (\neg\sigma_{1,o_1} \vee \neg\sigma_{n,o_1})$. This replacement induces an explanation path whose size is $2n + 2$. Therefore, the depth of ψ^2 is $d^{\psi^2} = 2n + 2$. By considering the ratio between the depth of these two MUSes, we thus get that the metric gap is $\Omega(n)$. \square

Proposition 11. The metric gap for the breadth and the length is $\Omega(n^{n-1})$.

Proof. Consider the instance given in Example 4. The MUS ψ^1 has a breadth and a length equal to n and $3n + 1$, respectively. In contrast, the MUS ψ^3 contains n at-least clauses from which n^n activation paths will be derived. Therefore, ψ^3 has a breadth equal to n^n and a length equal to $3n^n + 1$ (each activation path has size 4). Hence, the metric gap for both metrics is $\Omega(n^{n-1})$. \square

Note that, except for the number of items metric gap, the computation of all other metric gaps involves MUSes which only use clauses from the basic encoding ϕ . Therefore, restricting to MUSes of the basic encoding would not help decreasing the provided bounds.

Many proofs are based on the instance given in Example 4. However, if we remove edge $\{1, 3\}$ from the social network, then ψ^1 is not a valid MUS anymore. Then, the only smallest MUS, in terms of the number of variables, the number of clauses, the breadth, and the length, is ψ^2 which is not a MUS of the basic encoding. This again highlights the interest of the redundant encoding to get small MUSes.

6 Experimental Evaluation

In this section, we will empirically compare our encodings and metrics, with the aim of finding an appropriate MUS in order to construct its associated satisfiability implication graph and derive an explanation from it, thanks to a depth-first search along its activation states.

For this purpose, we generate synthetic data where agents' preferences are drawn from *impartial culture*, i.e., given a number of agents n , each linear order \succ_i is drawn with uniform probability among all possible linear orders, for $i \in [n]$, and social network graphs are generated from two well-known models for random network generation: *Erdős-Rényi's model (ER)* [12], and *Barabási-Albert's model (BA)* [1]. In ER random graphs, each edge is added with independent probability $p \in [0, 1]$, producing a graph whose density tends to p . For our simulations, we use $p \in \{0.25, 0.375, 0.5, 0.625, 0.75\}$. Alternatively, in BA random graphs, the idea is to iteratively construct the network by adding a new node to connect to m existing nodes which are chosen according to a preferential attachment mechanism, i.e., it is more likely to be connected to higher degree nodes. For our simulations, we use $m \in \{\lfloor 0.25n \rfloor, \lfloor 0.5n \rfloor, \lfloor 0.75n \rfloor\}$. We use NetworkX implementations of ER and BA random graphs.

We solve the problem of LEF existence thanks to SAT solvers. In our simulations, we use the PySAT [18] module with in particular the OptUx solver to extract and enumerate smallest size MUSes, on the basis of a weighted CNF formula, and possibly all the MUSes. However, enumerating all MUSes was a highly demanding task, computationally speaking, even for small instances. Therefore, we have decided to only focus on MUSes of minimum size. For this purpose, we use OptUx on weighted CNF formulas with a weight 1 on each clause. Even generating all minimum MUSes was computationally challenging in practice, therefore our experiments consider a relatively small number of agents, i.e., $n \in \{3, \dots, 8\}$. Since we aim to explain the lack of LEF allocations, we focus on 100 random instances where no LEF allocations exist for each experimental setting.

By definition, the redundant encoding contains more clauses than the basic one. However, using the redundant encoding can help to get smaller MUSes. We report in Figure 3 the size in average (over 100 instances) of the minimum MUSes for the basic or redundant encoding. It turns out that, on average, minimum MUSes under the redundant encoding always have a smaller size than under the basic one. The gap in sizes seems to increase with the number of agents.

Concerning the number of minimum MUSes on average, which are presented in Figure 4, we remark an interesting behavior between the two encodings. While the number of minimum MUSes is larger under the redundant encoding for a small number of agents, this situation is reversed for a sufficiently large number of agents.

Even if we focus on minimum MUSes for computational reasons (thus minimizing the number of clauses metric), we still have an important number of MUSes. Therefore, one could desire to discriminate even more among the minimum MUSes by using the other met-

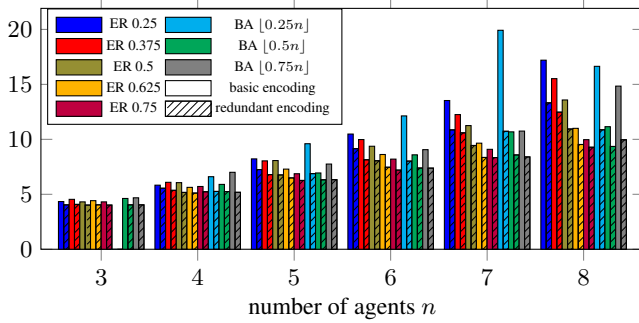


Figure 3. Average size of minimum MUSes

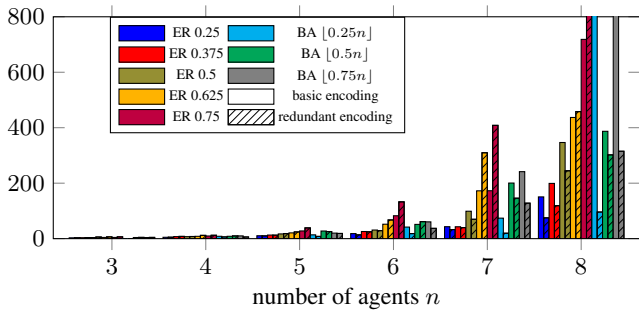


Figure 4. Average number of minimum MUSes

rics defined in Section 5. To get a clearer idea on which metrics better filter the MUSes, we perform simulations where we count the average number of minimum MUSes which also minimize each of the other proposed metrics. The results are presented in Figure 5 for the basic encoding and ER graphs. It is rather clear that choosing to minimize the number of concerned agents in MUSes enables to get fewer MUSes. Another metric which discriminates a lot among the minimum MUSes turns out to be the length, especially on sparser graphs.

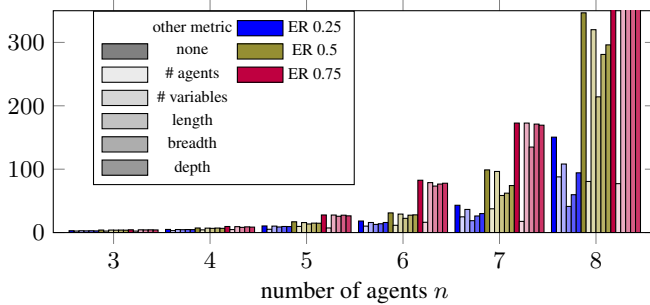


Figure 5. Average number of minimum MUSes which also minimize other metrics for Erdős-Rényi generated graphs under the basic encoding

In order to derive our explanation, we thus decide to choose, randomly, a minimum MUS which also minimizes the number of concerned agents. Nevertheless, to ensure that we do not lose much in the other metrics, we run more experiments to compute the average value of such chosen MUSes on the other metrics compared to the minimum, average, and maximum value of the metric over all minimum MUSes. We present in Figure 6 the results for the length metric in Erdős-Rényi graphs (the results for the other metrics and graphs show a similar behavior). It turns out that the average value of the metric over the minimum MUSes minimizing the number of concerned agents is around the average value of the metric over all min-

imum MUSes. Moreover, as the number of agents grows, it seems that the maximum value of the metric increases the gap with its average value, while our chosen MUSes stick to the average. Therefore, we rarely obtain the worst configurations for the other metrics by minimizing the number of agents.

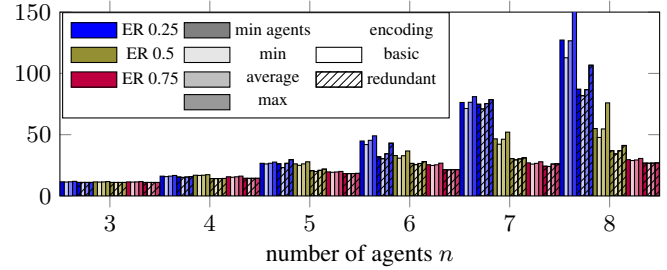


Figure 6. Evaluation of the minimum MUSes minimizing the number of agents on the length metric for Erdős-Rényi generated graphs

Finally, one can efficiently output textual explanations for the lack of LEF allocations, by using the DFS strategy over activation states of Algorithm 1 and the semantics of clauses on these chosen MUSes.

7 Conclusion

In this paper we address the question of explaining why no allocation respecting the property of local envy-freeness can be returned. Our study starts from an original SAT modeling of the problem. While our approach relies on MUSes as basic building blocks for explanations, we share with others [6] the view that a more interactive process is needed to present explanations to users. Thanks to a translation of MUSes to a dynamic activation process, we offer a fully automated way to generate textual explanations. We explore several metrics relevant to capture the simplicity of explanations, show that they can in theory greatly vary even among minimal MUSes, and report on experiments which suggest that minimizing the number of agents involved in explanations is a good filtering heuristic. There are certainly improvements which could simplify further this dynamic process (e.g., factoring some branches of explanations), but we believe this already offers a user-friendly output. As we have seen, this seemingly simple setting already triggers challenging conceptual and computational questions. Indeed, we have seen that even the computation of minimal MUSes can be too demanding. Recently, methods have emerged to compute cost-optimal unsatisfiable subsets [13]. It would be interesting to explore whether they could be adapted in our setting. Finally, while we have focused on explaining allocations without solution, our approach can be seen as a first step towards a more general theory of explainable fair division. First, we note that our approach can be easily adapted to provide local explanations, for instance by focusing on a specific agent (or set of agents) specially concerned by the decision. Furthermore, it can also be used as a basis to provide justifications in case an agent is unsatisfied with an existing allocation – in which case it could challenge counterfactually the outcome (“Why didn’t I get this object?”). If assuming this assignment leads to an unsatisfiable instance, then our approach can readily be used. Of course in practice it may well be that several possible fair solutions exist, including some where the agent indeed gets the item desired. In that case other criteria, like Pareto optimality, may have been used and thus mentioned in the explanation.

Acknowledgements

This work is partially supported by the ANR projects APPLE-PIE (grant ANR-22-CE23-0008-01) and AIDAL (grant ANR-22-CPJ1-0061-01).

References

- [1] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.
- [2] K. Belahcene, Y. Chevaleyre, N. Maudet, C. Labreuche, V. Mousseau, and W. Ouerdane. Accountable approval sorting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI 2018)*, pages 70–76, 2018.
- [3] A. Beynier, Y. Chevaleyre, L. Gourvès, A. Harutyunyan, J. Lesca, N. Maudet, and A. Wilczynski. Local envy-freeness in house allocation problems. *Autonomous Agents and Multi-Agent Systems*, 33:591–627, 2019.
- [4] A. Beynier, J.-G. Mailly, N. Maudet, and A. Wilczynski. Explaining the lack of locally envy-free allocations. Technical report, August 2024. See <https://hal.science/hal-04670468>.
- [5] A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors. *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2021. ISBN 978-1-64368-160-3. doi: 10.3233/FAIA336. URL <https://doi.org/10.3233/FAIA336>.
- [6] B. Bogaerts, E. Gamba, and T. Guns. A framework for step-wise explaining how to solve constraint satisfaction problems. *Artificial Intelligence*, 300:103550, 2021.
- [7] A. Boixel and U. Endriss. Automated justification of collective decisions via constraint solving. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*, pages 168–176, 2020.
- [8] R. Bredereck, A. Kaczmarczyk, and R. Niedermeier. Envy-free allocations respecting social networks. *Artificial Intelligence*, 305:103664, 2022.
- [9] O. Cailloux and U. Endriss. Arguing about voting rules. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016)*, pages 287–295, 2016.
- [10] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. arXiv 1702.08608, 2017.
- [11] E. Eiben, R. Ganian, T. Hamm, and S. Ordyniak. Parameterized complexity of envy-free resource allocation in social networks. *Artificial Intelligence*, 315:103826, 2023.
- [12] P. Erdős and A. Rényi. On random graphs I. *Publicationes Mathematicae (Debrecen)*, 6:290–297, 1959.
- [13] E. Gamba, B. Bogaerts, and T. Guns. Efficiently explaining CSPs with unsatisfiable subset optimization. *Journal of Artificial Intelligence Research*, 78:709–746, 2023.
- [14] C. Geist and D. Peters. Computer-aided methods for social choice theory. In *Trends in Computational Social Choice*, chapter 13, pages 249–267. AI Access, 2017.
- [15] H. Hosseini. The fairness fair: Bringing human perception into collective decision-making. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI 2024)*, volume 38, pages 22624–22631, 2024.
- [16] S. Huang and M. Xiao. Object reachability via swaps along a line. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI 2019)*, volume 33, pages 2037–2044, 2019.
- [17] A. Ignatiev, A. Previti, M. H. Liffiton, and J. Marques-Silva. Smallest MUS extraction with minimal hitting set dualization. In *Proceedings of the 21st International Conference on Principles and Practice of Constraint Programming (CP 2015)*, pages 173–182, 2015.
- [18] A. Ignatiev, A. Morgado, and J. Marques-Silva. PySAT: A Python toolkit for prototyping with SAT oracles. In *Proceedings of the 21st International Conference on Theory and Applications of Satisfiability Testing (SAT 2018)*, pages 428–437, 2018.
- [19] U. Junker. Quickxplain: preferred explanations and relaxations for over-constrained problems. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI 2004)*, pages 167–172, 2004.
- [20] J. Leite, J. Alferes, and B. Mito. Resource allocation with answer-set programming. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, pages 649–656, 2009.
- [21] J. Marques-Silva and A. Previti. On computing preferred muses and mcse. In *Proceedings of the 17th International Conference on Theory and Applications of Satisfiability Testing (SAT 2014)*, pages 58–74, 2014.
- [22] H. Moulin. *Fair division and collective welfare*. MIT Press, 2003.
- [23] D. Peters, A. D. Procaccia, A. Psomas, and Z. Zhou. Explainable voting. In *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, volume 33, pages 1525–1534, 2020.
- [24] A. D. Procaccia. Axioms should explain solutions. In J.-F. Laslier, H. Moulin, M. R. Sanver, and W. S. Zwicker, editors, *The Future of Economic Design: The Continuing Development of a Field as Envisioned by Its Researchers*, pages 195–199. Springer, 2019.
- [25] S. A. Suryanarayana, D. Sarne, and S. Kraus. Justifying social-choice mechanism outcome for improving participant satisfaction. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022)*, pages 1246–1255, 2022.
- [26] Z. Zahedi, S. Sengupta, and S. Kambhampati. ‘Why didn’t you allocate this task to them?’ Negotiation-aware task allocation and contrastive explanation generation. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI 2024)*, volume 38, pages 10243–10251, 2024.