Boolean Attribute Models

Clemens Anzinger
TU Wien
Vienna, Austria
clemens.anzinger@tuwien.ac.at

Alexander Temper
TU Wien
Vienna, Austria
alexander.temper@tuwien.ac.at

ABSTRACT

We study the k-boolean attribute model, which is a restriction of the k-attribute model to boolean valued attributes. We show that finding a k-boolean attribute model is NP-complete, even when either the attributes of the voters or alternatives is predetermined, or when preference lists are bounded by length two. Additionally, we show that the problem is fixed-parameter tractable with respect to the number of alternatives.

KEYWORDS

Computational social choice, restricted preference domains, computational complexity, parameterized complexity

1 INTRODUCTION

Oftentimes, one can explain a set of preferences by considering a set of qualitative attributes, such that if some person v prefers some alternative a to another alternative b then a has more of the qualitative attributes that person v cares for than b has. For example, one may consider whether a restaurant

- accepts credit card as a payment method,
- serves vegan options, and
- provides table service.

All the mentioned criteria are in a sense boolean; for example, either, a restaurant accepts credit cards or not. Assume someone prefers the fast-food restaurant to a pizzeria. It would then be reasonable to assume that the fast-food restaurant has more attributes than the pizzeria that the customer cares about. This could be because the fast food restaurant, which, say, has vegan options, and accepts card payment, has more of the qualitative criteria than the pizzeria, which does not accept card and has no vegan options, even though table service is provided.

Such a rationale may be expanded to the preferences of multiple people. There may exist an underlying set of attributes, such that each alternative has some of those attributes and likewise each person cares for some of them, such that the attributes explain the preferences.

Here, we propose a definition for the underlying model of this rationale, which we call the *k-boolean attribute model* ("*k-BAM*"), for which k denotes the number of attributes. It is a restriction of the *k-attribute model* proposed originally by Bhatnagar et al. [2], which considers quantitative attributes instead of qualitative ones.

Related work. The boolean attribute model is one way to succinctly describe a set of *preferences* of a set of *voters* over a set of *alternatives*, the three of which together are called a *preference profile*.

Multiple such succinct representations have been studied. Most notable are *single-peaked* (SP) [3] and *single-crossing* (SC) [17] preference profiles, which are preference profiles that can be represented by a special linear order of the alternatives and voters, respectively. Detecting single-peakedness and single-crossingness can be done in polynomial time [10] and both are characterized through two *forbidden sub-profiles* [1, 6] each. SP and SC preference profiles are well-studied. It has been shown that they admit fair voting rules [3] and efficient algorithms for some generally NP-complete problems [3, 5]. However, most preference profiles are not single-peaked [14] and likely not single-crossing, which motivates research for more general models.

A general model related to SP and SC preferences is that of *d-Euclidean preferences* [4]. A preference profile is called *d-Euclidean* for some $d \in \mathbb{N}$ if there exists an embedding $f : V \cup C \to \mathbb{R}^d$ mapping the voters V and alternatives C to some d-dimensional real-valued vector space, such that if an arbitrary voter v prefers alternative a to alternative b, then a has a smaller Euclidean distance to v than b has to v, that is, $\|f(a) - f(v)\|_2 < \|f(b) - f(v)\|_2$. Like SC and SP preferences, 1-Euclideanness can be detected in polynomial time [10], and in fact, every 1-Euclidean preference profile is single-peaked and single-crossing [10]. However the converse does not hold: there exists a sequence of increasingly large preference profiles $\mathcal{P}_k$ that are not 1-Euclidean, but for which deleting an arbitrary voter from a profile $\mathcal{P}_k$ results in a 1-Euclidean profile [7]. Hence, 1-Euclidean preferences can not, in comparison to SP and SC preferences, be characterized by finitely many sub-profiles. Additionally, while every preference profile is d-Euclidean for some d, d can be arbitrarily large [4] and deciding whether a preference profile is d-Euclidean is beyond NP even for $d = 2$ [16]. This inspires research on ways to succinctly model preference profiles, which may be computed more easily or are useful for certain decision problems when given as part of the input.

Some such general models that are currently under investigation are the previously introduced k-attribute model, the *k-list* model and the *k-range* model, all of which were proposed by Bhatnagar et al. [2].

To the best of our knowledge, Künnemann et al. [13] first consider the idea of restricting the k-attribute model to boolean values in a paper in which they demonstrate that, unless common complexity assumptions fail, no subquadratic algorithm for finding *stable matchings* exists for sufficiently large k, even if the attributes are boolean.

In a similar vein, Cheng and Rosenbaum [8] show that some stable matching problems can not be solved more efficiently for preference profiles described by at least $k \geq 6$ real-valued attributes, whilst many stable matching problems are fixed-parameter tractable with respect to k for preference profiles with a k-range model.

Our contribution. We provide a definition of the boolean attribute model based on set membership and show a dichotomy result regarding BAM-Existence, namely that the problem can be decided in polynomial time if the number of attributes $k = 2$ and that the problem is NP-complete if $k = 3$. We show that this NP-completeness remains also for preference profiles that have preference lists of length at most two.

We also examine various restricted cases, most importantly the cases where either **has** or **cares** is already given as part of the input. We find that even in these two cases, the problem remains NP-complete. Thereafter, we consider the parameterized complexity of the problem and show a fixed-parameter tractable algorithm for the problem parameterized by the number of alternatives m.

2 PRELIMINARIES

Boolean attribute models are succinct representations of *preference profiles*. Here, we consider *potentially incomplete*, *strict* preference profiles. "Potentially incomplete" means that alternatives can be missing from preference lists. "Strict" means that there are no ties.

Definition 1 (Preference profile)**.** A *preference profile* $\mathcal{P} = (V, C, R)$ consists of a set of voters $V = \{v_1, \dots, v_n\}$, a set of alternatives $C = \{c_1, \dots, c_m\}$, and a set of voter's *preference lists* $R = \{\succ_{v_1}, \dots, \succ_{v_n}\}$. The preference lists are strict, potentially incomplete orders of the alternatives, which describe the preferences of the voters over the alternatives. For a voter $v \in V$, their preference list is denoted by

$$v : a \succ b \succ c,$$

where $a \succ_v b$ means that v prefers a to b.

The *rank* of an alternative a for a voter v is the number of alternatives that v explicitly prefers to a, i.e., $r_v(a) := |\{b \mid a \succ_v b\}|$. The rank is undefined if v's preference list does not contain a. The *length* of v's preference list is denoted as $|\succ_v|$.

Example 2. Recall the restaurant example from the introduction. Consider the following preference profile

$$\text{Linda} : \text{fast food restaurant} \succ \text{pizzeria} \succ \text{hot dog stand},$$
$$\text{David} : \text{hot dog stand} \succ \text{pizzeria}.$$

The first line reads as "Linda prefers the fast food restaurant to the pizzeria and the pizzeria to the hot dog stand". We have that $|\succ_{\text{Linda}}| = 3$ and $|r_{\text{Linda}}(\text{hot dog stand})| = 2$. Further, notice that David does not rank the fast food restaurant, which illustrates that preference profiles can be incomplete.

Definition 3 (k-boolean attribute model)**.** A *k-boolean attribute model* $\mathcal{M} = (\textbf{has}, \textbf{cares})$ for a preference profile $\mathcal{P} = (V, C, R)$ is a tuple consisting of two functions, called $\textbf{has} : C \to 2^{[k]}$ and $\textbf{cares} : V \to 2^{[k]}$ that map alternatives and voters respectively to subsets of the attributes $[k] := \{1, \dots, k\}$, with the property that for each voter v and each pair of alternatives $a, b \in C$, it holds that

$$a \succ_v b \implies |\textbf{has}(a) \cap \textbf{cares}(v)| > |\textbf{has}(b) \cap \textbf{cares}(v)|. \quad (1)$$

We define the score of an alternative c with regard to a voter v as

$$\text{score}_v(c) := |\textbf{has}(c) \cap \textbf{cares}(v)|.$$

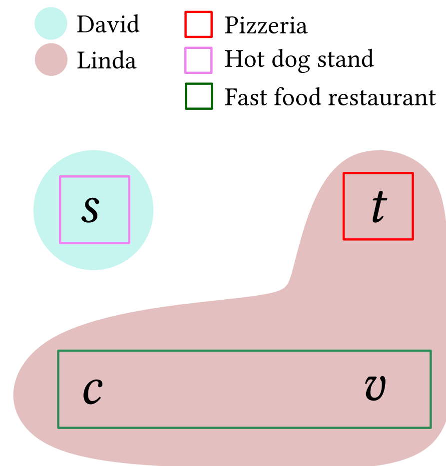

Figure 1: The 4-BAM of Example 4 for the preference profile of Example 2, illustrated as a Venn-diagram. The function has is represented by angular boxes and cares by round translucent areas.

We write that an alternative c **has** an attribute α if $\alpha \in \textbf{has}(c)$ (or **have**, depending on grammatical necessity), and that c **has** ℓ number of attributes if $|\textbf{has}(c)| = \ell$. The same holds for the use of **cares** and for expressing inequalities. We say that the functions (**has**, **cares**) *explain* a preference list if the condition of Equation 1 is true.

Example 4. Consider again the preference profile from Example 2 It can be described with four attributes $\{c, s, t, v\}$, as we can assign the attributes to the voters and alternatives as follows:

- Linda **cares** about receiving table service, being offered vegan options, and being able to pay with credit card, so $\textbf{cares}(\text{Linda}) := \{c, t, v\}$.
- David only **cares** about being offered sausages, so $\textbf{cares}(\text{David}) := \{s\}$.
- Of all criteria, the pizzeria only offers table service, so $\textbf{has}(\text{pizzeria}) := \{t\}$.
- Likewise, the hot dog stand only serves sausages, so $\textbf{has}(\text{hot dog stand}) := \{s\}$.
- Lastly, the fast food restaurant accepts card payment and offers vegan options, so $\textbf{has}(\text{fast food restaurant}) := \{c, v\}$.

This BAM is illustrated in Figure 1. Although the attributes of BAMs are defined to be subsets of the natural numbers, it may be useful to name the attributes for illustrative purposes. The above model, although it is using the attributes $\{c, s, t, v\}$, which represent whether card payment is accepted, sausages are served, table service is provided, and vegan options are available, constitutes a 4-BAM, because in general, as the number of attributes is finite, some arbitrary enumeration of the attributes suffices to make such BAMs conform to the definition.

Our main focus was investigating the complexity of determining whether a k-BAM exists for a given preference profile and number of boolean attributes, that is, the complexity of the following decision problem:

BAM-Existence
Input: A preference profile $\mathcal{P}$, an integer k
Question: Does there exist a k-boolean attribute model for $\mathcal{P}$?

We also study the complexity of *parameterized* versions of the above problem. A *parameterized problem* is a problem for which some information about every instance, called the *parameter* of the problem, is made explicit in the input. This allows for a more detailed analysis of the time-complexity of deciding the problem. For example, BAM-Existence can be parameterized by the number of alternatives. A problem is called *fixed-parameter tractable* with respect to some parameterization if there exists an algorithm that decides if every parameterized instance (I, k) is a YES-instance in time

$$f(k) \cdot |I|^{O(1)},$$

where $f : \mathbb{N} \to \mathbb{N}$ is a computable function and $|I|$ is the number of bits required to encode I as a binary string. Informally speaking, this means that when k is fixed, the time needed to solve an instance increases only polynomially when $|I|$ increases.

3 RESULTS

We first present some bounds on the number of attributes for a given preference profile $\mathcal{P}$. Thereafter, we show the promised dichotomy result for BAM-Existence, i.e., that the problem is in P when the number of attributes k is two and NP-complete when $k \geq 3$. This result is shown to be true even when the preference lists of a given instance are bounded in length by two.

Afterward, we consider whether the problem is easier if either **has** or **cares** is given as part of the input. Finally, we study the problem from the lens of fixed-parameter tractability.
But first, we start with an observation:

Observation 5. *One can check in polynomial time whether given functions* $(\mathbf{has}, \mathbf{cares})$ *explain a preference profile* $\mathcal{P} = (V, C, R)$ *(and therefore constitute boolean attribute model).*

This follows from the fact that one can determine the score of an alternative for a voter in polynomial time, and then iterate over every voter to check if their preference list is explained.
Given this observation, stating that BAM-Existence is contained in NP, it is sufficient to prove NP-hardness in order to show that BAM-Existence (or one of its restrictions) is NP-complete.

3.1 Some bounds on the number of attributes

Before showing hardness results, we first present some rules that allow us to decide right away whether an instance is a YES- or NO-instance. The first prerequisite can be derived from the list length of voters:

Lemma 6. *If a voter v has a preference list of length $\ell + 1$, then* $|\mathbf{cares}(v)| \geq \ell$.

Proof. Since we are working with strict preference lists and the scores along a preference list must be strictly decreasing, if a preference list has $\ell + 1$ ranked alternatives, the list must be related to at least $\ell + 1$ scores. If $|\mathbf{cares}(v)| < \ell$, then there can be at most ℓ scores, as the maximum achievable score is $\ell - 1$ and there are ℓ values in the interval $[0, \ell - 1]$. ◻

From this we conclude that there is a minimum value for k, which follows from the lengths of all preference lists in a preference profile. If k is smaller, no BAM can be found:

Corollary 7. *A preference profile $\mathcal{P} = (V, C, R)$ that has a preference list of length ℓ, can only be described by an attribute model with $\ell - 1$ or more attributes. Stated differently, for a k-BAM for $\mathcal{P}$, it holds for each $v \in V$ that*

$$k + 1 \geq |\succ_v|.$$

We can formulate another lemma using the rank of a voter, concerning what needs to hold for every k-BAM for some preference profile, more specifically, what needs to hold for the **has** of each alternative.

Lemma 8. *Let $c \in C$ be an alternative, $v \in V$ a voter of a preference profile with a k-BAM $\mathcal{M} = (\mathbf{has}, \mathbf{cares})$. Then*

(1) $|\mathbf{has}(c)| \geq |\succ_v| - r_v(c) - 1$ *and*
(2) $|\mathbf{has}(c)| \leq k - r_v(c)$.

In other words, if an alternative is ranked better than ℓ others, it **has** *at least ℓ attributes, and if it is ranked worse than ℓ others, it* **has** *at most $k - \ell$ attributes.*

Furthermore, we examine the ranks of an alternative for different voters to find further requirements for k. This leads to another lower bound for k.

Lemma 9. *For every preference profile $\mathcal{P} = (V, C, R)$ with a k-BAM, for each alternative $c \in C$ and voters $v, w \in V$ for which $r_v(c) \geq r_w(c)$,*

$$k + 1 \geq |\succ_w| - r_w(c) + r_v(c).$$

Proof. Remember that $r_v(c) \geq r_w(c)$. Applying Lemma 8 for v, we find that

$$k - r_v(c) \geq |\mathbf{has}(c)|$$

and for w, we know that

$$|\mathbf{has}(c)| \geq |\succ_w| - r_w(c) - 1$$

from which we obtain

$$k - r_v(c) \geq |\succ_w| - r_w(c) - 1 \iff$$
$$k + 1 \geq |\succ_w| - r_w(c) + r_v(c).$$

◻

With Lemma 9, we examine each pair of occurrences of an alternative in pairs of voters, and then get a lower bound for k. Assume the following example, with voters v and w:

$$v : a \succ b \succ c \succ d \succ e$$
$$w : f \succ d \succ g \succ h \succ i.$$

We examine alternative d, where $r_w(d) = 1$ and $r_v(d) = 3$. Lemma 9 determines a minimum value of $k+1 \geq 5-1+3$, leading to $k \geq 6$. Intuitively, there need to be at least three attributes that d **has**, in order to be able to explain voter w, but also three attributes that d does not **have**, in order to explain voter v. This of course must hold for all alternatives.

3.2 Finding BAMs is hard in general

In this section, we show a dichotomy result for the number of attributes. It is the case, that deciding if a BAM with two attributes exists can be done in polynomial time, whereas deciding if a BAM with three attributes exists is NP-complete, even if the length of the preference lists is bound by two. We show the former by demonstrating a reduction from BAM-Existence with $k = 2$ to 2-SAT, which can be decided in polynomial time, and the latter by reducing 3-Coloring to BAM-Existence with $k = 3$, which is known to be NP-complete.

Theorem 10. *BAM-Existence is in P for $k = 2$.*

Proof. Due to Corollary 7, a preference profile that contains a preference list of length four or longer is a trivial NO-instance. On the other hand, preference lists of length one can be ignored, as they contain no pairs, so the implicit universal quantification of Equation 1 is true anyhow. Henceforth, we may concentrate only on preference profiles that contain preference lists of length three or length two. We show that BAM-Existence with $k = 2$ can be decided in polyomial time by demonstrating a reduction to 2-SAT, which is known to be decidable in polynomial time, as shown by Krom [12]. Let $(\mathcal{P}, 2)$ with $\mathcal{P} := (V, C, R)$ be a BAM-Existence instance. We now construct a 2-SAT instance Φ, such that Φ is a YES-instance if and only if $(\mathcal{P}, 2)$ is a YES-instance.

Let the attributes be $A := \{\alpha, \beta\}$. For each alternative $a \in C$ we introduce two propositional variables $h_{a,\alpha}$ and $h_{a,\beta}$, representing whether alternative a **has** α and β, respectively. Call the set of all those variables X. We model each voter v by a 2-CNF formula Φ_v

Since we are only looking to decide whether a 2-BAM exists, it is enough to assign the **has** from which the **cares** can consequently be determined.

There are two cases to be considered, namely preference lists of length three and of length two. We encode preference lists of length three of the form $v_3 : a \succ b \succ c$ as

$$\Phi_{v_3} := \phi_1 \wedge \phi_2 \wedge \phi_3, \tag{2}$$

where

$$\begin{aligned}
\phi_1 &:= h_{a,\alpha} \wedge h_{a,\beta}, \\
\phi_2 &:= (\neg h_{b,\alpha} \vee \neg h_{b,\beta}) \wedge (h_{b,\alpha} \vee h_{b,\beta}), \text{ and} \\
\phi_3 &:= \neg h_{c,\alpha} \wedge \neg h_{c,\beta}.
\end{aligned}$$

For preference lists of length two, we only need to make sure that the first ranked alternative **has** one attribute that the second ranked alternative does not **have**, therefore, we encode lists with length two of the form $v_2 : d \succ e$ as

$$\Phi_{v_2} := \phi_4 \wedge \phi_5 \wedge \phi_6 \wedge \phi_7, \tag{3}$$

where

$$\begin{aligned}
\phi_4 &:= h_{d,\alpha} \vee h_{d,\beta}, \\
\phi_5 &:= \neg h_{e,\alpha} \vee \neg h_{e,\beta}, \\
\phi_6 &:= h_{d,\alpha} \vee \neg h_{e,\beta}, \text{ and} \\
\phi_7 &:= h_{d,\beta} \vee \neg h_{e,\alpha}.
\end{aligned}$$

Each preference list is encoded as a formula Φ_v, like v_3 or v_2 respectively, depending on its length. Finally, the full 2-SAT encoding is $\Phi := \bigwedge_{v \in V} \Phi_v$.

We claim that computing the encoding Φ from $\mathcal{P}$ can be done in polynomial time and the size of Φ is polynomial in the size of $\mathcal{P}$. For each voter, the number of clauses is constant (four clauses for a preference list with length two, and three clauses for a preference list of length three). The number of variables needed is upper bounded by a constant value of six per voter (each voter can rank at most three alternatives, and two variables per unique alternative are needed). Therefore, the time needed for computing Φ is $O(|V|)$ and Φ's size is also $O(|V|)$.

We claim that Φ is a YES-instance of 2-SAT if and only if $(\mathcal{P}, 2)$ is a YES-instance of BAM-Existence. We now show correctness.

Assume $(\mathcal{P}, 2)$ is a YES-instance of BAM-Existence. Then, there exists a BAM $\mathcal{M} = (\textbf{has}, \textbf{cares})$ with two attributes for $\mathcal{P}$. We claim that Φ is also a YES-instance, because we can construct a satisfying assignment $I : X \to \{0, 1\}$ from $\mathcal{M}$. Let I be an assignment, for which holds that for each $c \in C$

$$\begin{aligned}
\alpha \in \textbf{has}(c) &\implies I(h_{c,\alpha}) = 1 \\
\beta \in \textbf{has}(c) &\implies I(h_{c,\beta}) = 1 \\
\alpha \notin \textbf{has}(c) &\implies I(h_{c,\alpha}) = 0, \text{ and} \\
\beta \notin \textbf{has}(c) &\implies I(h_{c,\beta}) = 0.
\end{aligned}$$

We claim that such an I is well-defined and a satisfying assignment to Φ. It is clearly well-defined, as $\alpha \in \textbf{has}(c)$ and $\beta \in \textbf{has}(c)$ is defined for every alternative c, and an element can not be simultaneously contained and not contained in a set. Next, we argue why I is a satisfying assignment. To do so, we consider preference lists of length two and three separately. We assign truth values in the following way:

For every preference list of length three of the form $v \in V$, $v : a \succ b \succ c$ it must hold that

$$\begin{aligned}
\text{score}_v(a) &= 2, \\
\text{score}_v(b) &= 1, \\
\text{score}_v(c) &= 0 \text{ and} \\
\textbf{cares}(v) &= \{\alpha, \beta\}. && \text{(By Lemma 6.)}
\end{aligned}$$

Note that the encoding Φ_v of v has the form of Φ_{v_1} from Equation 2. A score of two for alternative a implies $\textbf{has}(a) = \{\alpha, \beta\}$, as there is no other way to get a score of two with two attributes. Thus $I(h_{a,\alpha}) = I(h_{a,\beta}) = 1$. This leads to ϕ_1 being satisfied. A score of one for alternative b implies that either $\textbf{has}(a) = \{\alpha\}$ or $\textbf{has}(a) = \{\beta\}$, as these are the only ways to get a score of one given that $\textbf{cares}(v) = \{\alpha, \beta\}$. Thus, only either $I(h_{b,\alpha}) = 1 \wedge I(h_{b,\beta}) = 0$ or $I(h_{b,\alpha}) = 0 \wedge I(h_{b,\beta}) = 1$. This leads to ϕ_2 being satisfied. Lastly, a score of zero for alternative c can only be achieved if $\textbf{has}(a) = \emptyset$, thus $I(h_{c,\alpha}) = I(h_{c,\beta}) = 0$. This means that ϕ_3 is satisfied. Thus, formula Φ_v is satisfied.

We now consider an arbitrary voter $v \in V$ with a preference list of the form $v : d \succ e$. The score of alternative d must be greater than the score of alternative e, meaning alternative d can have a score of either one or two, alternative e can have a score of either zero or one, and they can not be equal, that is

$$\begin{aligned}&\text{score}_v(d) \geq 1,\\&\text{score}_v(e) \leq 1, \text{ and}\\&\text{score}_v(d) \neq \text{score}_v(e).\end{aligned}$$

Note that the encoding Φ_v of v has the form of Φ_{v_2} from Equation 3. Alternative d must **have** least one attribute, otherwise it could not get a score of at least one. This leads to ϕ_4 being satisfied. Similarly, alternative e can not **have** both attributes, as then, it would be impossible for alternative d to have a higher score. This leads to ϕ_5 being satisfied. We still need to show that ϕ_6 and ϕ_7 are satisfied. We know that there are three possible attribute assignments for alternative d, which we will consider separately again here:

- Assume $\mathbf{has}(d) = \{\alpha, \beta\}$. This means $I(h_{d,\alpha}) = I(h_{d,\beta}) = 1$, in which case both ϕ_6 and ϕ_7 are satisfied.
- Assume $\mathbf{has}(d) = \{\alpha\}$. This means that $I(h_{d,\alpha}) = 1$ and $I(h_{d,\beta}) = 0$. Thus, formula ϕ_6 is satisfied. We also know that the score of alternative d must be one, meaning alternative e must have a score of zero. Since the voter must **care** about α in this case, we know that alternative e must not **have** α, meaning $I(h_{e,\alpha}) = 0$, which satisfies ϕ_7.
- Assume $\mathbf{has}(d) = \{\beta\}$. This works analogously to the case above.

Thus, formula Φ_v is satisfied under I. As only lists of length two and three are possible, and we have shown all of them are satisfied, formula Φ is satisfied under I, i.e., formula Φ is a YES-instance.

We have shown that if a BAM-Existence instance with $k = 2$ is a YES-instance, then the constructed 2-SAT instance is also a YES-instance. Now we examine the other direction.

Assume Φ is a YES-instance, i.e. there is some satisfying assignment $I : X \to \{0, 1\}$ for Φ. We claim that the corresponding BAM-Existence instance is also a YES-instance, because we can construct a BAM from I. Define $\mathcal{M} := (\mathbf{has}, \mathbf{cares})$, such that for each alternative $c \in C$ it holds that

$$\begin{aligned}I(h_{c,\alpha}) = 1 &\implies \alpha \in \mathbf{has}(c),\\I(h_{c,\beta}) = 1 &\implies \beta \in \mathbf{has}(c),\\I(h_{c,\alpha}) = 0 &\implies \alpha \notin \mathbf{has}(c) \text{ and}\\I(h_{c,\beta}) = 0 &\implies \beta \notin \mathbf{has}(c).\end{aligned}$$

and for each voter $v \in V$ it holds that $\mathbf{cares}(v) := \mathbf{has}(c^*)$, where c^* is $v's$ most preferred alternative.

This is well-defined, as the assignment I is defined for each variable. Next, we argue that this constitutes a BAM for $\mathcal{P}$. We again consider preference lists of length two and three separately.

Consider a preference list of length of length three of the form $v : a \succ b \succ c$. As ϕ_1, ϕ_2 and ϕ_3 are all satisfied under I, it follows that $\mathbf{has}(a) = \{\alpha, \beta\}$, $|\mathbf{has}(b)| = 1$ and $|\mathbf{has}(c)| = 0$. By definition, $\mathbf{cares}(v) = \{\alpha, \beta\}$, which means that v is explained by $\mathcal{M}$, since

$$\text{score}_v(a) = 2 > \text{score}_v(b) = 1 > \text{score}_v(c) = 0.$$

Next, consider the preference list of a voter $v \in V$ of the form $v : d \succ e$. Consider the following:

- As $\phi_4 = h_{d,\alpha} \vee h_{d,\beta}$ is satisfied, it must be the case that $|\mathbf{has}(d)| \geq 1$.
- As $\phi_5 = \neg h_{e,\alpha} \vee \neg h_{e,\beta}$ is satisfied, it must be the case that $|\mathbf{has}(e)| \leq 1$.
- Because of the above and the fact that $\phi_6 = h_{d,\alpha} \vee \neg h_{e,\beta}$ and $\phi_7 = h_{d,\beta} \vee \neg h_{e,\alpha}$ are satisfied, it can not be the case that $\mathbf{has}(d) = \mathbf{has}(e)$, as if both **have** exactly one attribute, it can not be the same one.

As a consequence, given that $\mathbf{cares}(v) = \mathbf{has}(d)$, $\text{score}_v(d) > \text{score}_v(e)$ in all cases. Thus, every preference list is explained by $\mathcal{M}$, i.e., profile $\mathcal{P}$ is a YES-instance.

As we have shown both directions, the proof is done. ◻

Next we show a reduction to BAM-Existence with $k = 3$ from 3-Coloring, which is known to be NP-complete [15], demonstrating the dichotomy for the complexity of BAM-Existence. We quickly state the problem statement without delving into much detail.

3-Coloring
Input: A graph $G = (V, E)$
Question: Does there exist a coloring $\chi : V \to \{R, G, B\}$ such that for every edge $\{a, b\} \in E$, it holds that $\chi(a) \neq \chi(b)$?

Theorem 11. *BAM-Existence is NP-complete for $k = 3$.*

Proof. We show that 3-Coloring reduces to BAM-Existence with $k = 3$. We first demonstrate the reduction and afterward prove its correctness.

First, we construct a preference profile $\mathcal{P} = (V, C, R)$ from a given graph $G = (U, E)$ through the following steps:

- To obtain C, add each vertex $u \in U$ and three dummy alternatives d_3, d_2, d_0 as an alternative, so $C := U \cup \{d_3, d_2, d_0\}$.
- To obtain V and R, for each vertex $u \in U$, add a voter with the preference list
$$v_u : d_3 \succ d_2 \succ u \succ d_0$$
and for each edge $\{a, b\} \in E$, introduce the voters
$$\begin{aligned}&v_{(a,b)} : a \succ b, \text{ and}\\&v_{(b,a)} : b \succ a.\end{aligned}$$

Further, let $A := \{R, G, B\}$ be the attributes, thus $k = 3$. This concludes the description of the reduction. Profile $\mathcal{P}$ can be computed in $O(|U| + |E|)$ time and has size $O(|U| + |E|)$.

We claim that G is 3-colorable if and only if $\mathcal{P}$ admits a 3-BAM.

Assuming that G is 3-colorable, there exists a coloring $\chi : U \to \{R, G, B\}$ such that no two adjacent vertices are mapped to the same color. From χ, one can construct a 3-BAM as follows:

- For $u \in U$, let $\mathbf{has}(u) := \{\chi(u)\}$ and $\mathbf{cares}(v_u) := \{R, G, B\}$.
- Let $\mathbf{has}(d_3) := \{R, G, B\}$, $\mathbf{has}(d_2) := \{R, G\}$ and $\mathbf{has}(d_0) := \emptyset$.
- For $\{a, b\} \in E$, let $\mathbf{cares}(v_{(a,b)}) := \{\chi(a)\}$ and $\mathbf{cares}(v_{(b,a)}) := \{\chi(b)\}$.

We claim that $\mathcal{M} := (\mathbf{has}, \mathbf{cares})$ constitutes a 3-BAM for $\mathcal{P}$. First, it uses only three attributes. Further, each voter's preference list is explained by $\mathcal{M}$, because for every v_u,

$$\begin{aligned}|\mathbf{has}(d_3) \cap \mathbf{cares}(v_u)| = 3 >\\|\mathbf{has}(d_2) \cap \mathbf{cares}(v_u)| = 2 >\\|\mathbf{has}(u) \cap \mathbf{cares}(v_u)| = 1 >\\|\mathbf{has}(d_0) \cap \mathbf{cares}(v_u)| = 0.\end{aligned}$$

Additionally, as for each edge $\{a, b\} \in E$, it holds that $\chi(a) \neq \chi(b)$, both $v_{(a,b)}$ and $v_{(b,a)}$ are explained, because

$$|\mathbf{has}(a) \cap \mathbf{cares}(v_{(a,b)})| = 1 > |\mathbf{has}(b) \cap \mathbf{cares}(v_{(b,a)})| = 0 \text{ and}$$
$$|\mathbf{has}(b) \cap \mathbf{cares}(v_{(b,a)})| = 1 > |\mathbf{has}(a) \cap \mathbf{cares}(v_{(a,b)})| = 0.$$

On the other hand, assume that $\mathcal{M} = (\mathbf{has}, \mathbf{cares})$ is a 3-BAM of $\mathcal{P}$. We claim that we can easily construct a coloring from $\mathcal{M}$. For every $u \in U$ it must be that $\mathbf{cares}(v_u) = \{R, G, B\}$ because $| \succ_{v_u} | = 4$. Further, it must also be that $|\mathbf{has}(u) \cap \mathbf{cares}(v_u)| = 1$. Combining the two facts implies that $|\mathbf{has}(u)| = 1$, that is, each vertex **has** exactly one attribute. Further, for every edge $\{a, b\} \in E$, we have that $|\mathbf{has}(a)| \cap |\mathbf{has}(b)| = \emptyset$. Thus, a function mapping each $u \in U$ to its respective single attribute is a well-defined 3-coloring for G, as no two adjacent vertices obtain the same color.

This concludes the proof of correctness. □

This result holds even whenever the preference lists are bounded in length by two.

Theorem 12. *BAM-Existence is NP-complete for $k = 3$, even for preference profiles for which every preference list has length at most two.*

See proof on page 8.

However, there is a special case, where finding a BAM is solvable in polynomial time. That is, if all preference list lengths share the following relation to k:

Lemma 13. *If all preference lists in a preference profile are of length $k + 1$, BAM-Existence can be solved in polynomial time.*

Proof. We know that the **cares** sets of all voters must contain all attributes due to Lemma 6. Due to Lemma 9, we also know that each alternative can only appear at the same rank for all voters, as $|r_w(a) - r_v(a)|$ must be zero for all alternatives a and voter pairs v, w.

Therefore, we can simply iterate over all preference lists in polynomial time, and check whether an alternative appears in different positions. If the alternative is found in different positions, the given instance is a NO-instance, otherwise, it is a YES-instance, and the attributes can be assigned trivially based on the rank of an alternative. □

3.3 Finding BAMs is hard for partially given solutions

The next question that might arise is which information makes finding a BAM easier and which makes it harder. We have investigated the two cases when either **has** or **cares** was given as part of the input. Let us state the respective computational problems.

BAM-Existence with Has/Cares
Input: A preference profile $\mathcal{P} = (V, C, R)$, a function **has**/**cares** : $C/V \to 2^{[k]}$ with $k \in \mathbb{N}$
Question: Does a function **cares**/**has** : $V/C \to 2^{[k]}$ exist such that (**has**, **cares**) is a k-BAM for $\mathcal{P}$?

We show that the problem remains NP-complete for $k \geq 3$ in either case. We start with a reduction from 3-SAT, an NP-complete problem [9] to BAM-Existence with Cares.

Theorem 14. *BAM-Existence with Cares is NP-complete.*

See proof on page 9.

The counterpart BAM-Existence with Has is also NP-complete. We show this by demonstrating a reduction from Restricted Exact 3-set Cover, which was shown to be NP-complete [11].

Restricted Exact 3-set Cover (RXC3)
Input: A set X of elements with cardinality $|X| = 3 \cdot q$ where $q \in \mathbb{N}$, a collection $\mathcal{S} = \{S_1, \dots, S_{3q}\}$, where for each subset $S_i \in \mathcal{S}$, it holds that $S_i \subset \mathcal{X}$, and $|S_i| = 3$, and each element is contained in exactly three subsets S_i.
Question: Is there a collection of subsets $\mathcal{S}' \subset \mathcal{S}$ such that each $x_i \in X$ is contained in exactly one subset $S'_i \in \mathcal{S}'$.

We call a collection of subsets with the above properties a *restricted exact 3-set cover* for its input. Additionally, note that $|S'| = q$ follows from the definition of each subset having cardinality exactly three.

Theorem 15. *BAM-Existence with Has is NP-complete.*

See proof on page 9.

The fact that the reduction maps every RXC3 instance to a preference profile with only a single voter gives yet another hardness result.

Corollary 16. *BAM-Existence with Has is NP-hard even for one voter.*

This hardness result actually talks about the parameterized complexity, which will be the main topic of the next section.

3.4 Parameterized complexity

In this section, we show that BAM-Existence is fixed-parameter tractable for the number of alternatives. This is due to the fact that the number of alternatives m bounds the other parameters. It is easy to see that there are only $O(m!)$ many different preference lists, thus $n \leq O(m!)$. Further, we show the following upper bound.

Lemma 17. *Let $\mathcal{P} = (V, C, R)$ with $|C| = m$. The preference profile $\mathcal{P}$ admits a BAM with $k = (m - 1) \cdot m$ and one with $k = (m - 1) \cdot n$.*

See proof on page 11.

This yields the fact that BAM-Existence is fixed-parameter tractable with respect to m, simply due to brute force. Notice that this brute force algorithm "guesses" the **has** function; we denote this choosing an arbitrary element of

$$\underbrace{2^{[k]} \times \cdots \times 2^{[k]}}_{m \text{ times}} = 2^{[k]^m},$$

of which there are $2^{k \cdot m}$ many.

Theorem 18. *There is an algorithm deciding BAM-Existence for a preference profile with m alternatives in less than $2^{m^5} \cdot |\mathcal{P}|^{O(1)}$ time.*

Proof. We show that Algorithm 1 is correct and has the aforementioned time-complexity.

If $k \geq m(m - 1)$, the algorithm is correct due to Lemma 17 and runs in constant time. Otherwise, the algorithm is a triply nested for-loop with at most $2^k \cdot 2^m \cdot n \cdot 2^k < 2^{m^2} \cdot 2^m \cdot n \cdot 2^{m^2}$ iterations and a $|\mathcal{P}|^{O(1)}$ body. Thus, the algorithm needs fewer than $2^{O(m^5)} \cdot |\mathcal{P}|^{O(1)}$ many steps. Further, assume that a k-BAM $\mathcal{M} = (\mathbf{has}, \mathbf{cares})$ exists. Then, either another k-BAM is found, or

Algorithm 1: BruteForceM

Input: A preference profile $\mathcal{P} = (V, C, R)$, an integer k

Output: $\top$ if $\mathcal{P}$ admits a k-BAM else $\bot$

$m \leftarrow |C|$;
if $k \geq m(m-1)$ **then**
  return $\top$
end
foreach has $\subseteq 2^{[k]^m}$ **do**
  foreach $v \in V$ **do**
    foreach $S \subseteq [k]$ **do**
      if S *explains* v *under* **has** **then**
        cares$(v) \leftarrow S$;
      end
    end
  end
  if (**has**, **cares**) *is BAM for* $\mathcal{P}$ **then**
    return $\top$
  end
end
return $\bot$

has will be guessed and **cares** will consequently be found, so the algorithm returns $\top$. Further, $\top$ is returned only if a BAM has been found. Consequently, the algorithm is correct. □

Lastly, notice that for complete preference profiles, the smallest k such that a profile admits a k-BAM is bounded both from below by a function of m and from above. This yields the following equivalence.

Corollary 19. *Let k be the minimum number of attributes such that a given preference profile admits a k-BAM and L be a decision problem defined on complete preference profiles.*

$$L \text{ is FPT w.r.t } m \iff L \text{ is FPT w.r.t } k.$$

Proof. This follows from $m \leq k \leq (m-1) \cdot m$ for complete preference profiles. □

It is, however, not clear whether BAM-Existence is fixed-parameter tractable with respect to number of voters n, which is yet to be thoroughly investigated. First attempts at showing $W[1]$-hardness, have been unsuccessful. We weakly conjecture that the problem is fixed-parameter tractable with respect to n and provide a starting point for investigating the problem.

Theorem 20. *BAM-Existence is decidable in linear time if $n = 2$.*

See proof on page 11.

4 CONCLUSION

In this work, we formally introduced boolean attribute models, a way to explain preference profiles with boolean attributes, and the corresponding decision problem BAM-Existence, which we have proven to be NP-complete in general. It remains NP-complete, even when given the **has** or **cares** sets as part of the input. However, we found some special cases where the problem is solvable in polynomial time. Lastly, we explored the problem regarding parameterized complexity and found fixed-parameter tractability for various parameters, most importantly for the number of alternatives m.

Open questions. While the general problem is NP-complete, there may be more special cases where the problem is tractable. Finding such cases, that are real-world applicable, could lead to a big boost for the usefulness of BAMs.

In terms of parameterized complexity, the most important open question is the complexity of BAM-Existence with respect to the number of voters n. Additionally, more work on the parameterized problem with number of alternatives m as input could lead to a better algorithm than the one proposed before.

Lastly, we only focused on the hardness aspect of finding boolean attribute models. The next step is to research whether such models prove to be useful for other problems. More specifically, the question is whether there are NP-hard problems that become solvable in polynomial time when a BAM is given as part of the input. Some candidates to study for this are various voting problems, such as NP-complete winner-determination problems.

5 CITATIONS AND REFERENCES

REFERENCES

[1] Miguel A. Ballester and Guillaume Haeringer. [n.d.]. A Characterization of the Single-Peaked Domain. 36, 2 ([n. d.]), 305–322. https://doi.org/10.1007/s00355-010-0476-3

[2] Nayantara Bhatnagar, Sam Greenberg, and Dana Randall. 2008. Sampling stable marriages: why spouse-swapping won't work. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms.* 1223–1232.

[3] Duncan Black. [n.d.]. The Decisions of a Committee Using a Special Majority. 16, 3 ([n. d.]), 245–261. https://doi.org/10.2307/1907278 arXiv:1907278

[4] Anna Bogomolnaia and Jean-François Laslier. 2007. Euclidean preferences. *Journal of Mathematical Economics* 43, 2 (2007), 87–98.

[5] Robert Bredereck, Jiehua Chen, Ugo Paavo Finnendahl, and Rolf Niedermeier. [n.d.]. Stable Roommate with Narcissistic, Single-Peaked, and Single-Crossing Preferences. In *Algorithmic Decision Theory* (Cham, 2017), Jörg Rothe (Ed.). Springer International Publishing, 315–330. https://doi.org/10.1007/978-3-319-67504-6_22

[6] Robert Bredereck, Jiehua Chen, and Gerhard J. Woeginger. [n.d.]. A Characterization of the Single-Crossing Domain. 41, 4 ([n. d.]), 989–998. https://doi.org/10.1007/s00355-012-0717-8

[7] Jiehua Chen, Kirk R. Pruhs, and Gerhard J. Woeginger. [n.d.]. The One-Dimensional Euclidean Domain: Finitely Many Obstructions Are Not Enough. 48, 2 ([n. d.]), 409–432. https://doi.org/10.1007/s00355-016-1011-y

[8] Christine T. Cheng and Will Rosenbaum. [n.d.]. Stable Matchings with Restricted Preferences: Structure and Complexity. 10, 3 ([n. d.]), 13:1–13:45. https://doi.org/10.1145/3565558

[9] Stephen A Cook. 2021. The complexity of theorem-proving procedures (1971). (2021).

[10] J. P. Doignon and J. C. Falmagne. [n.d.]. A Polynomial Time Algorithm for Unidimensional Unfolding Representations. 16, 2 ([n. d.]), 218–233. https://doi.org/10.1006/jagm.1994.1010

[11] Teofilo F. Gonzalez. 1985. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science* 38 (1985), 293–306. https://doi.org/10.1016/0304-3975(85)90224-5

[12] M. R. Krom. 1967. The Decision Problem for a Class of First-Order Formulas in Which all Disjunctions are Binary. *Mathematical Logic Quarterly* 13, 1-2 (1967), 15–20.

[13] Marvin Künnemann, Daniel Moeller, Ramamohan Paturi, and Stefan Schneider. [n.d.]. Subquadratic Algorithms for Succinct Stable Matching. 81, 7 ([n. d.]), 2991–3024. https://doi.org/10.1007/s00453-019-00564-x

[14] Marie-Louise Lackner and Martin Lackner. [n.d.]. On the Likelihood of Single-Peaked Preferences. 48, 4 ([n. d.]), 717–745. https://doi.org/10.1007/s00355-017-1033-0

[15] Christos H. Papadimitriou. [n.d.]. *Computational Complexity.* Addison-Wesley.

[16] Dominik Peters. [n.d.]. *Recognising Multidimensional Euclidean Preferences.* AAAI. https://aaai.org/papers/10616-aaai-31-2017/

[17] Kevin WS Roberts. 1977. Voting over income tax schedules. *Journal of public Economics* 8, 3 (1977), 329–340.

APPENDIX

Theorem 12. *BAM-Existence is NP-complete for $k = 3$, even for preference profiles for which every preference list has length at most two.*

Proof of Theorem 12. We demonstrate a reduction from 3-Coloring, which is known to be NP-complete [15], similar to the proof above. Assume we are given a graph $G = (U, E)$. For simplicity, we assume that no vertices of degree zero exist in G (these vertices can be colored with any color). We construct an instance of BAM-Existence with $k = 3$ and a preference profile $\mathcal{P} = (V, C, R)$ the following way:

- As $k = 3$, name the attributes $A := \{R, G, B\}$.
- Let $C := U \cup D_1 \cup D_2 \cup \{d_3\}$, where $D_1 := \{d_{1,1}, d_{1,2}, d_{1,3}\}$, and $D_2 := \{d_{2,1}, d_{2,2}, d_{2,3}\}$, that is, the alternatives consist of the vertices of the graph and dummy alternatives. We aim to construct a preference profile in a way such that for every BAM over 3 attributes, alternatives in D_1 must **have** exactly one attribute, alternatives in D_2 must **have** exactly two attributes, and the alternative d_3 must **have** all three attributes.
- Introduce anonymous dummy voters that require that, for every BAM with $k = 3$, the number of attributes per alternative is necessarily as described before. For each alternative $d_2 \in D_2$ and each alternative in $d_1 \in D_1$, introduce a voter preferring d_2 over d_1. In total, this means we introduce voters with the preference lists

$$\begin{array}{lll} d_{2,1} \succ d_{1,1}, & d_{2,1} \succ d_{1,2}, & d_{2,1} \succ d_{1,3}, \\ d_{2,2} \succ d_{1,1}, & d_{2,2} \succ d_{1,2}, & d_{2,2} \succ d_{1,3}, \\ d_{2,3} \succ d_{1,1}, & d_{2,3} \succ d_{1,2}, & d_{2,3} \succ d_{1,3}, \end{array}$$

  all of which ensure that every alternative in D_2 **has** more attributes than each alternative in D_1. Further, introduce voters with the preference lists

$$\begin{array}{lll} d_{1,1} \succ d_{1,2}, & d_{1,2} \succ d_{1,1}, & d_{1,3} \succ d_{1,1}, \\ d_{1,1} \succ d_{1,3}, & d_{1,2} \succ d_{1,3}, & d_{1,3} \succ d_{1,2}, \end{array}$$

  all of which ensure that no two alternatives in D_1 **have** the same attribute in all BAMs. We also introduce voters in a similar vain for D_2, that is, we introduce voters with the preference lists

$$\begin{array}{lll} d_{2,1} \succ d_{2,2}, & d_{2,2} \succ d_{2,1}, & d_{2,3} \succ d_{2,1}, \\ d_{2,1} \succ d_{2,3}, & d_{2,2} \succ d_{2,3}, & d_{2,3} \succ d_{2,2}. \end{array}$$

  Further, introduce for each alternative in D_2 a voter preferring d_3 over that alternative, that is, introduce voters with the preference lists

$$d_3 \succ d_{2,1}, \quad d_3 \succ d_{2,2}, \quad d_3 \succ d_{2,3}.$$

  Next, introduce for each vertex $u \in U$ the voters

$$\begin{aligned} &v_{u,1} : d_{2,1} \succ u, \\ &v_{u,2} : d_{2,2} \succ u, \text{ and} \\ &v_{u,3} : d_{2,3} \succ u. \end{aligned}$$

  Then, introduce for each edge $\{a, b\} \in E$ the voters

$$\begin{aligned} &v_{(a,b)} : a \succ b, \text{ and} \\ &v_{(b,a)} : b \succ a. \end{aligned}$$

This concludes the construction. Clearly, profile $\mathcal{P}$ can be computed in $O(|V| + |E|)$ and has size $O(|V| + |E|)$.

Before we go into the correctness proof, we make a claim about the just constructed BAM-Existence instance.

Claim 21. *For Each Model $\mathcal{M}$ for $\mathcal{P}$, it must hold that alternatives $d_{1,i} \in D_1$* **have** *exactly one attribute, alternatives $d_{2,i} \in D_2$* **have** *exactly two attributes, and alternative d_3* **has** *all three attributes. Furthermore, all dummy alternatives are different.*

Each alternative in D_1 is ranked above two other alternatives, meaning it must **have** at least one attribute. Also, they are ranked above each other, meaning they must be different. From this, it follows that each alternative in D_2 must **have** at least two attributes. Otherwise it could not get ranked above each alternative in D_1.

W.l.o.g, assume now, that $\textbf{has}(d_{1,1}) = \{R, G\}$, i.e., that it **has** two attributes. This would mean that one alternative in D_2 must have three attributes, as all alternatives in D_2 must **have** at least two attributes, be different and be ranked above $d_{1,1}$. However, then a conflict would occur, as d_3 can not be ranked above all alternatives in D_2, if one of them already **has** three attributes. This means that d_3 must have all three attributes, alternatives in D_2 have all combinations of two attributes, and alternatives in D_1 all have one different attribute. This concludes the proof for Claim 21. ◇

We claim that G is a YES-instance if and only if $\mathcal{P}$ admits a 3-BAM.

First, assume that the graph is 3-colorable. Then, there exists a 3-coloring $\chi : U \to \{R, G, B\}$. From χ we construct a 3-BAM in the following way:

- For all $u \in U$, let $\textbf{has}(u) := \{\chi(u)\}$.
- For all $\{a, b\} \in E$, let $\textbf{cares}(v_{(a,b)}) := \{\chi(a)\}$ and $\textbf{cares}(v_{(b,a)}) := \{\chi(b)\}$.
- For all $u \in U$, let

$$\textbf{cares}(v_{u,1}) := \textbf{cares}(v_{u,2}) := \textbf{cares}(v_{u,3}) := \{R, G, B\}.$$

- Let $\textbf{has}(d_3) := \{R, G, B\}$.
- Let the alternatives in D_2 **have** all possible combinations of two attributes, that is, $\textbf{has}(d_{2,1}) := \{R, G\}$, $\textbf{has}(d_{2,2}) := \{R, B\}$, and $\textbf{has}(d_{2,3}) := \{G, B\}$.
- Let the alternatives in D_1 **have** all possible combinations of one attribute, that is, $\textbf{has}(d_{1,1}) := \{R\}$, $\textbf{has}(d_{1,2}) := \{G\}$, and $\textbf{has}(d_{1,3}) := \{B\}$.
- Let $v \in V$ be some voter. If some dummy alternative $d \in D_1 \cup D_2 \cup \{d_3\}$ is voter v's most preferred alternative, then let $\textbf{cares}(v) := \textbf{has}(d)$.

We claim that $\mathcal{M} := (\textbf{has}, \textbf{cares})$ constitutes a 3-BAM for $\mathcal{P}$. First, it only uses three attributes. Furthermore, each voter's preference list is explained by $\mathcal{M}$. All dummy voters are explained. For dummy voters, an alternative that is ranked first always **has** at least as many attributes as the second ranked alternative. Furthermore, no two dummy alternatives **have** the same attributes. Finally, for all dummy voters v, we set $\textbf{cares}(v)$ to **has** of the alternative ranked first. This leads to the first ranked alternative always having a higher score.

Secondly, all other voters concerning edges are explained. For each edge $\{a, b\} \in E$, two voters are present. W.l.o.g. assume $\chi(a) = R$ and $\chi(b) = B$, this means that $\textbf{cares}(a) = \{R\}$ and $\textbf{cares}(b) = \{B\}$.

Additionally, we have set $\mathbf{cares}(v_{a,b}) = \{R\}$ and $\mathbf{cares}(v_{b,a}) = \{B\}$, leading to a score of one for alternative a in $v_{a,b}$ and a score of zero for alternative b in $v_{a,b}$. It is the opposite for voter $v_{b,a}$.

Finally, all voters concerning vertices are explained. Each alternative $u \in U$ **has** exactly one attribute, and each dummy alternative in D_2 **has** exactly two attributes. The voters $v_{u,1}, v_{u,1}, v_{u,1}$ **care** about all attributes, leading to scores of two for the dummies and one for u, explaining the voters.

Therefore, the tuple $\mathcal{M}$ constitutes a 3-BAM for profile $\mathcal{P}$.

Now, assume that some $\mathcal{M} = (\mathbf{has}, \mathbf{cares})$ *constitutes a 3-BAM for* $\mathcal{P}$. We claim that by giving each vertex in U a color based on the attributes, the corresponding $\chi(c) := \mathbf{has}(c)$ gives a 3-coloring for G, with a slight abuse of notation.

First, we show that each alternative $u \in U$ **has** exactly one attribute. To start, u must **have** at least one attribute, as it is ranked above a neighbor in some voter. Furthermore, u can **have** at most one attribute, which can be concluded from the following:

Using Claim 21, we can conclude that u can **have** at most one attribute. Since u is ranked below each alternative in D_2, which **have** all possible combinations of two attributes, it can not **have** exactly two attributes, otherwise it would be identical to another alternative in D_2 and could not be ranked below it. Therefore, u must have exactly one attribute.

Lastly, for each $\{a, b\} \in E$, alternatives a and b must have different attributes. This follow from the voters $v_{a,b}$ and $v_{b,a}$. We know that a and b have exactly one attribute. If $\mathbf{has}(a) = \mathbf{has}(b)$, then both of these voters could not be explained, meaning a and b must have different attributes, and in turn, different colors.

This concludes the correctness proof. ◻

Theorem 14. *BAM-Existence with Cares is NP-complete.*

Proof of Theorem 14. We claim that 3-SAT reduces to BAM-Existence with Cares. Let Φ be a 3-CNF formula with variables $X = \{x_1, \ldots, x_n\}$ and clauses $\mathcal{C} = \{C_1, \ldots, C_m\}$, with each clause being a set of literals.

We construct a preference profile $\mathcal{P}$ and specify the **cares** sets as follows:

- Let the alternatives consist of a choice gadget c and dummies d_0 and d_2.
- Let the attributes be represented by all $2n$ literals of X, that is, $A := \{\neg x \mid x \in X\} \cup X$.
- For each $x \in X$, introduce a voter with the preference list $v_x : d_2 \succ a \succ d_0$ who **cares** only about x and its negation, that is, $\mathbf{cares}(v_x) := \{x, \neg x\}$.
- For each clause $C \in \mathcal{C}$, introduce a voter $v_C : d_2 \succ c$ who only **cares** about the negations of C's literals, that is, $\mathbf{cares}(C) := \{\neg \ell \mid \ell \in C\}$.

This finishes the construction of $\mathcal{P}$. The obtained preference profile is polynomial in size with respect to the length of Φ and can be constructed in polynomial time. We claim that $\mathcal{P}$ with the given **cares** function admits a **has** function such that $(\mathbf{has}, \mathbf{cares})$ constitutes a BAM if and only if Φ has a satisfying assignment.

Assuming there exists a variable assignment $I : X \longrightarrow \{0, 1\}$ *satisfying* Φ, *then* **has** *can be determined as follows.* Define $\mathbf{has}(c)$ such that for each literal ℓ, it holds that $\ell \in \mathbf{has}(c) \iff I(\ell) = 1$ and let d_2 **have** all attributes and d_0 **have** no attributes, i.e.,

$$\mathbf{has}(d_2) := A,\ \mathbf{has}(d_0) := \emptyset.$$

We claim that, for every $x \in X$, the preferences of voter v_x are explained by **has**. This is because v_x **cares** only about x and its negation, and exactly one of x or $\neg x$ is contained in $\mathbf{has}(c)$, thus

$$|\mathbf{cares}(v_x) \cap \mathbf{has}(d_2)| = 2 >$$
$$|\mathbf{cares}(v_x) \cap \mathbf{has}(c)| = 1 >$$
$$|\mathbf{cares}(v_x) \cap \mathbf{has}(d_0)| = 0.$$

Further, for each clause $C \in \mathcal{C}$, at least one of C's literals evaluates to 1 under I, which means that at most two of them evaluate to 0. This means that $|\mathbf{has}(c) \cap \mathbf{cares}(v_C)| < 3 = |\mathbf{has}(d_2) \cap \mathbf{cares}(v_C)|$, which shows that v_C is explained.

On the other hand, assume $(\mathbf{has}, \mathbf{cares})$ *is a BAM for* $\mathcal{P}$. We claim that an assignment I for which for every literal ℓ it is true that $\ell \in \mathbf{has}(a) \iff I(\ell) = 1$ is a satisfying assignment for Φ. For every variable $x \in X$, $I(x)$ is well-defined, as either $x \in \mathbf{has}(c)$ or $\neg x \in \mathbf{has}(a)$, because of the facts that $|\mathbf{cares}(v_x)| = 2$, $|\succ_{v_x}| = 3$ and $|r_{v_x}(x) = 1$. It is a satisfying assignment, as for each clause C, voter v_C's preferences imply that $\mathbf{has}(c)$ contains the negations of at most two literals of C, which in turn means that $\mathbf{has}(c)$ contains at least one of C's literals. As this is the case for an arbitrary clause of Φ, I is a satisfying assignment.

Thus, we have shown that the reduction is correct and runs in polynomial time and space. This means BAM-Existence with Cares is NP-hard. As one can check if $(\mathbf{has}, \mathbf{cares})$ constitute a BAM for $\mathcal{P}$, the problem is also NP-complete. ◻

Theorem 15. *BAM-Existence with Has is NP-complete.*

Proof of Theorem 15. We show a reduction from an instance of RXC3 to an instance of BAM-Existence with Has.

Let $I = (X, \mathcal{S})$ be an instance of RXC3, where we denote $X = \{x_1, \ldots x_{3q}\}$ and $S = \{S_1, \ldots, S_{3q}\}$. From I, we construct a preference profile $\mathcal{P} := (V, R, C)$. We choose $C := X \cup \{d_1, d_2, d_3\}$, that is, we introduce an alternative for each element $x \in X$ and three dummy alternatives. The set of voters $V := \{v\}$ consists of exactly one voter with the preference list

$$v : d_2 \succ x_{3q} \succ \cdots \succ a_q \succ d_1 \succ x_{q-1} \succ \cdots \succ x_1 \succ d_0.$$

Next, we describe the **has** of the instance. It can be obtained as follows:

- For each set $S_j \in \mathcal{S}$, introduce an attribute α_j denoting membership in S_j, that is, $x_j \in S \iff \alpha_j \in \mathbf{has}(x_j)$ for every $x_j \in X$.
- Additionally, let $\alpha_j \in \mathbf{has}(d_1)$ for all S_j.
- For each element x_i, introduce i many dummy attributes $\delta_i^1 \ldots, \delta_i^{i-1}$ that only x_i **has**, meaning that
$$\delta_i^j \in \mathbf{has}(x_\ell) \iff i = \ell \wedge j < i.$$
Additionally, if $i \geq q$, introduce in the same way one more attribute δ_i^i.
- Introduce $3q + 2$ dummy attributes $\delta_{3q+1}^1, \ldots, \delta_{3q+1}^{3q+2}$ that only alternative d_2 **has**, so $\delta_{3q+1}^j \in \mathbf{has}(c) \iff c = d_2 \wedge 1 \leq j \leq 3q + 2$.

	x_1	x_2	x_3	$\dots$	x_{3q-1}	x_{3q}	d_2	d_1	d_0
α_1							0	1	0
$\dots$							0	1	0
α_{3q}							0	1	0
$\delta^1_{a_2}$	0	1	0	0	0	0	0	0	0
$\delta^1_{a_3}$	0	0	1	0	0	0	0	0	0
$\delta^2_{a_3}$	0	0	1	0	0	0	0	0	0
$\dots$	0	0	0	$\dots$	0	0	0	0	0
$\delta^1_{a_{3q-1}}$	0	0	0	0	1	0	0	0	0
$\dots$	0	0	0	0	1	0	0	0	0
$\delta^{3q-1}_{a_{3q-1}}$	0	0	0	0	1	0	0	0	0
$\delta^1_{a_{3q}}$	0	0	0	0	0	1	0	0	0
$\dots$	0	0	0	0	0	1	0	0	0
$\delta^{3q}_{a_{3q}}$	0	0	0	0	0	1	0	0	0
$\delta^1_{D_1}$	0	0	0	0	0	0	1	0	0
$\dots$	0	0	0	0	0	0	1	0	0
$\delta^{3q+2}_{D_1}$	0	0	0	0	0	0	1	0	0

Table 1: Table showing the creation of a k-Cares Existence instance from RXC3 instance. A 1 in column c and row α indicates that attribute $\alpha \in \text{has}(c)$. The empty fields depend on the original RXC3 instance and are therefore left blank in this example.

To summarize, each alternative $x_i \in X$ **has** exactly three of the attributes $\alpha_1, \dots, \alpha_{3q}$ and additionally it **has** either $i-1$ dummy attributes or i dummy attributes δ_i, depending on whether $i \geq q$ or not. Alternative d_2 **has** $3q+2$ attributes that no other alternative **has**, d_1 **has** all of $\alpha_1, \dots, \alpha_{3q}$, and d_0 **has** no attributes. An illustration thereof is given by Table 1. This concludes the reduction.

Clearly, the reduction takes polynomial time and the BAM-Existence with Has instance $(\mathcal{P}, \mathbf{has})$ has polynomial size. Next, we show the correctness of the reduction.

Assume that $S' \subseteq S$ is a restricted exact 3-set cover of X. Call the set of all dummy attributes D. We claim that $\mathbf{cares}(v) := \{\alpha_j | S_j \in \mathcal{S}'\} \cup D$ constitutes a BAM for $(\mathcal{P}, \mathbf{has})$. Recall the preference list of v, which is

$$v : d_2 \succ x_{3q} \succ \dots \succ x_q \succ d_1 \succ x_{q-1} \succ \dots \succ x_1 \succ d_0.$$

The alternatives obtain the following score for v:

- The score of d_2 is $3q+2$ as d_2 **has** $3q+2$ dummy attributes, all of which v **cares** about.
- If $i \geq q$, the score of x_i is $i+1$, as $\mathbf{has}(x_i)$ contains exactly one of $\alpha_1, \dots, \alpha_{3q}$. This in turn is because x_i is contained in only one of subset contained in $\mathcal{S}'$ by the definition of a restricted exact 3-set cover. The same argument applies for showing that $|\mathbf{has}(x_i) \cap \mathbf{cares}(v)| = i$ for $i < q$.
- The dummy d_1 has a score of q, because $\mathbf{has}(d_1) = \{\alpha_1, \dots \alpha_{3q}\}$ and $|\mathbf{cares}(v) \cap \{\alpha_1, \dots \alpha_{3q}\}| = q$.
- The dummy d_0 has a score of 0, because $\mathbf{has}(d_0) = \emptyset$.

Thus, we have that v's preference list is explained, as

$$\begin{aligned}
|\mathbf{has}(d_2) \cap \mathbf{cares}(v)| &= 3q+2 > \\
|\mathbf{has}(x_{3q}) \cap \mathbf{cares}(v)| &= 3q+1 > \\
|\mathbf{has}(x_{3q-1}) \cap \mathbf{cares}(v)| &= 3q > \\
&\dots \\
|\mathbf{has}(x_q) \cap \mathbf{cares}(v)| &= q+1 > \\
|\mathbf{has}(d_1) \cap \mathbf{cares}(v)| &= q > \\
|\mathbf{has}(x_{q-1}) \cap \mathbf{cares}(v)| &= q-1 > \\
&\dots \\
|\mathbf{has}(x_1) \cap \mathbf{cares}(v)| &= 1 > \\
|\mathbf{has}(d_0) \cap \mathbf{cares}(v)| &= 0.
\end{aligned}$$

On the other hand, assume that $\mathcal{M} := (\mathbf{has}, \mathbf{cares})$ is a BAM for $\mathcal{P}$. We claim that the sets $\mathcal{S}^*$ corresponding to the non-dummy attributes v **cares** for, that is, $\mathbf{cares}(v) \setminus D$, are a restricted exact 3-set cover for X. This relies on the following observation.

Claim 22. *The scores for the alternatives ranked by voter v are $(3q+2, 3q+1, 3q, \dots, 2, 1, 0)$, meaning they are descending by ones.*

Alternative d_2 must have a score of $3q+2$. It can not have a score higher than $3q+2$, because $|\mathbf{has}(d_2)| = 3q+2$. However, it also can not have a score of less than $3q+2$, as it is preferred to $3q+1$ other alternatives.

Since the maximum score is $3q+2$, and in total there are $3q+3$ alternatives ranked, the scores must be descending by ones from $\{3q+2, \dots, 0\}$, as no draws or scores below zero are allowed. $\diamond$

From this we obtain that the score of d_1 must be q, as it is in the q^{th} position counting from 0. Observe that d_1 only **has** non-dummy attributes. As a consequence, v must **care** about exactly q non-dummy attributes, that is, $|\mathbf{cares}(v) \setminus D| = q$. Thus, $\mathcal{S}^*$ contains exactly q subsets.

It is left to show that each element $x_i \in X$ is contained in exactly one subset $S \in S^*$. Consider x_i such that $i \geq q$. Its score is $i+1$, but $|\mathbf{has}(x_i) \cap D| = i$, so $|(\mathbf{has}(x_i) \cap \mathbf{cares}(v)) \setminus D| \geq 1$, so x_i is contained in at least one of $\mathcal{S}^*$. An analogous argument can be made for $i < q$.

Finally, we claim that no element is contained in two or more subsets of $\mathcal{S}^*$. Aiming for a contradiction, assume x_i is contained in the two subsets S_1, S_2 of $\mathcal{S}^*$, and each other element is contained in exactly one subset of $\mathcal{S}^*$. However, this implies that

$$\begin{aligned}
3q = \left| \bigcup_{S \in \mathcal{S}^*} S \right| &\leq |S_1 \cup S_2| + \left| \bigcup_{S \in \mathcal{S}^* \setminus \{S_1, S_2\}} S \right| \\
&\leq 5 + 3 \cdot (q-2) \\
&= 3q - 1 \\
&< 3q,
\end{aligned}$$

which is a contradiction ($3q \nless 3q$.). Consequently, every element must appear in exactly one subset of $\mathcal{S}^*$. This concludes the proof of correctness.

Thus, Restricted Exact 3-Set Cover reduces to BAM-Existence with Has, which means that the latter is NP-hard. It is also NP-complete, as checking if (**has**, **cares**) is a BAM can be done in polynomial time. □

Lemma 17. *Let $\mathcal{P} = (V, C, R)$ with $|C| = m$. The preference profile $\mathcal{P}$ admits a BAM with $k = (m-1) \cdot m$ and one with $k = (m-1) \cdot n$.*

Proof of Lemma 17. This can be shown constructively. Let each alternative c have $m-1$ attributes $\alpha_c^1, \dots \alpha_c^{m-1}$ that only c **has**, i.e.,

$$A := \{\alpha_c^i | c \in C, i \in [m-1]\}$$

such that for all $a, c \in C$,

$$\alpha_a^i \in \mathbf{has}(c) \iff a = c \wedge 1 \leq i \leq m.$$

Then, let every voter $v \in V$ **care** about as many of the attributes of any given alternative as they need. This can be described formally by

$$\mathbf{cares}(v) := \{\alpha_c^i | v \text{ ranks } c \text{ and } 1 \leq i \leq |\succ_v| - r_v(c)\}.$$

The above is well-defined, as each voter needs to **care** about at most $(m-1)$ many private attributes of a given alternative, because a list of m alternatives is can be described by $m-1$ attributes.

We claim that the functions (**has**, **cares**) constitute a BAM. For every $v \in V$, $c \in C$, if v ranks c, the score obtained by c for v equals $|\succ_v| - r_v(c)$, which strictly decreases when traversing down the preference list and non-negative by the definition of the rank.

The analogous statement for $k = (m-1) \cdot n$ can be shown similarly instead of introducing $(m-1)$ private attributes for each alternative, introduce $(m-1)$ private attributes for each voter. □

Theorem 20. *BAM-Existence is decidable in linear time if $n = 2$.*

Proof of Theorem 20. As this proof will be rather long, we start off by giving an overview of the things that will be shown. First, we view BAMs through a slightly different lens, by partitioning the attributes into *attribute types*. Thereafter, we give a construction which corresponds to a BAM for a given preference profile with two voters. We will prove that it is a BAM by showing that the scores of the alternatives correspond to their positions in the preferences of the voters, which implies that the scores are descending. Lastly, we show that the number of attributes is the least possible for all BAMs of the preference profile by showing that there exists a pair of alternatives which require at least that number of attributes.

Let us start with the proof. Let $\mathcal{P} := (\{u, w\}, C, R)$. Generally, observe that the attributes A of every BAM $\mathcal{M} := (\mathbf{has}, \mathbf{cares})$ for $\mathcal{P}$ can be partitioned into three disjoint sets, one set $A_{u,w}$ consisting of the attributes that both u and w **care** about, one set A_u consisting of the attributes that only u **cares** about and one set A_w consisting of the attributes that only w **cares** about. Formally, this means that $A =: A_u \uplus A_w \uplus A_{u,w}$ such that

$$A_u := \mathbf{cares}(u) \setminus \mathbf{cares}(w),$$
$$A_w := \mathbf{cares}(w) \setminus \mathbf{cares}(u),$$
$$A_{u,w} := \mathbf{cares}(u) \cap \mathbf{cares}(w).$$

With slight abuse of notation, define for every alternative c that

$$\mathbf{has}_{u,w}(c) := \mathbf{has}(c) \cap A_{u,w},$$
$$\mathbf{has}_u(c) := \mathbf{has}(c) \cap A_u, \text{ and}$$
$$\mathbf{has}_w(c) := \mathbf{has}(c) \cap A_w.$$

It turns out that the scores of alternatives can be computed differently through the above insight.

Claim 23. *The score of an alternative c for voter u is equal to the sum of the number of A_u attributes alternative c* **has** *plus the number of $A_{u,w}$ attributes alternative c* **has***, that is,*

$$\mathrm{score}_u(c) = |\mathbf{has}_u(c)| + |\mathbf{has}_{u,w}(c)|.$$

Likewise,

$$\mathrm{score}_w(c) = |\mathbf{has}_w(c)| + |\mathbf{has}_{u,w}(c)|.$$

We only show the first equation for $\mathrm{score}_u(c)$, as the argument is analogous for $\mathrm{score}_w(c)$. Observe that

$$\mathrm{score}_u(c) \tag{4}$$
$$= |\mathbf{has}(c) \cap \mathbf{cares}(u)| \tag{5}$$
$$= |\mathbf{has}(c) \cap ((\mathbf{cares}(u) \cap \mathbf{cares}(w)) \cup (\mathbf{cares}(u) \setminus \mathbf{cares}(w)))| \tag{6}$$
$$= |\mathbf{has}(c) \cap (A_u \cup A_{u,w})| \tag{7}$$
$$= |(\mathbf{has}(c) \cap A_u) \cup (\mathbf{has}(c) \cap A_{u,w})| \tag{8}$$
$$= |\mathbf{has}(c) \cap A_u| + |\mathbf{has}(c) \cap A_{u,w}| \tag{9}$$
$$= |\mathbf{has}_u(c)| + |\mathbf{has}_{u,w}(c)|, \tag{10}$$

which shows that the claim is true. Notice how A_u and $A_{u,w}$ are disjoint, which is why the equality of (8) between (9) is true. ◇

Thus, an assignment of $|\mathbf{has}_u(c)|, |\mathbf{has}_w(c)|, |\mathbf{has}_{u,w}(c)|$ for every alternative $c \in C$ is a certificate for BAM-Existence with $n = 2$ if

- they are all non-negative,
- for each pair $a, b \in C$, whenever $a \succ_u b$ then
$$|\mathbf{has}_{u,w}(a)| + |\mathbf{has}_u(a)| > |\mathbf{has}_{u,w}(b)| + |\mathbf{has}_u(b)|, \tag{11}$$
- the analogous holds whenever $a \succ_w b$, and
- $|A_u| + |A_w| + |A_{u,w}| = k$.

Next we give a construction that gives values for $|\mathbf{has}_u(c)|, |\mathbf{has}_w(c)|, |\mathbf{has}_{u,w}(c)|$ for every alternative $c \in C$. To build some intuition, we will first talk informally about the construction. Generally, the important point is that the values will be assigned in a way such that the scores will be minimal in a sense that they will correspond to the positions of the alternatives. For example, if $u : c_0 \succ c_1 \succ c_2$, then c_2 will obtain a score of 0 for u, c_1 a score of 1 and c_0 a score of 2.

The construction proceeds as follows: First, the values are being assigned to the alternatives which are ranked by both u and w. Initially, as many attributes of $A_{u,w}$ will be assigned to each alternative as possible, without exceeding the corresponding least needed score for both voters. Then, the gap to the larger least needed score for either u or w will be achieved through newly allocated attributes in A_u or A_w respectively.

Then, for each alternative, we "convert" as many attributes in $A_{u,w}$ to one already existing attribute in A_u and one already existing attribute in A_w as possible, with the intent to reduce the number of $A_{u,w}$ attributes without increasing the number of A_u and A_w attributes.

After this, the alternatives ranked by only one voter will receive their attributes. Again, we do so by giving them exactly as many attributes as needed for their position. We try to reuse all attributes that have already been assigned; if that is not enough, new attributes are allocated in $A_{u,w}$.

Formally, define the following. For every $c \in C$, let

$$\lambda_u(c) := \begin{cases} |\succ_u| - r_u(c) & u \text{ ranks } c\text{, and} \\ 0 & \text{otherwise,} \end{cases}$$

which is the score needed to justify c's position in u's preferences. Define $\lambda_w(c)$ analogously.

Then, for every alternative $c \in C$ which is ranked by both u and w, let

$$\begin{aligned} t_{c,u,w} &:= \min(\lambda_u(c), \lambda_w(c)), \\ t_{c,u} &:= \lambda_u(c) - t_{c,u,w}, \\ t_{c,w} &:= \lambda_w(c) - t_{c,u,w},, \\ |A_u| &:= \max_{c \in C} t_{c,u}, \\ |A_w| &:= \max_{c \in C} t_{c,w}, \\ \mathrm{convert}_c &:= \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}), \\ |\mathbf{has}_{u,w}(c)| &:= t_{c,u,w} - \mathrm{convert}_c, \\ |\mathbf{has}_u(c)| &:= t_{c,u} + \mathrm{convert}_c, \text{ and} \\ |\mathbf{has}_w(c)| &:= t_{c,w} + \mathrm{convert}_c. \end{aligned}$$

Then, for each alternative c ranked only by u, assign as many $A_{u,w}$ attributes as needed to achieve $\lambda_u(c)$, whilst using as many A_u attributes as needed and possible, that is,

$$\begin{aligned} |\mathbf{has}_w(c)| &:= 0, \\ |\mathbf{has}_u(c)| &:= \min(\lambda_u(c), |A_u|), \text{ and} \\ |\mathbf{has}_{u,w}(c)| &:= \lambda_u(c) - |\mathbf{has}_u(c)|. \end{aligned}$$

Do the analogous for alternatives ranked only by w. Lastly, define

$$\begin{aligned} |A_{u,w}| &:= \max_{c \in C} |\mathbf{has}_{u,w}(c)|, \text{ and} \\ k &:= |A_{u,w}| + |A_u| + |A_w|. \end{aligned}$$

This construction can clearly be computed in linear time.

We proceed to show that the above construction yields the "smallest" BAM for $\mathcal{P}$, which involves showing three things:

- For every $c \in C$, the assignments of $|\mathbf{has}_u(c)|, |\mathbf{has}_w(c)|$, and $|\mathbf{has}_{u,w}|$ are in a sense well behaved. This means the following.
  - They are well-defined. This is clearly the case, because every alternative is ranked by either both or only one of the voters.
  - They are all non-negative. We show this in Claim 24.
  - They are smaller than $|A_u|, |A_w|$ and $|A_{u,w}|$, respectively. We show this in Claim 25.
- The scores are descending for both voters, i.e., fulfill Property (11). This follows from the central observation we make in Claim 26, which is that the scores correspond to the positions.
- Every BAM for $\mathcal{P}$ consists of at least k attributes. We show this in Claim 29.

Claim 24. *For every alternative $c \in C$,*

- $|\mathbf{has}_u(c)| \geq 0$,
- $|\mathbf{has}_w(c)| \geq 0$, *and*
- $|\mathbf{has}_{u,w}(c)| \geq 0$.

Consider an arbitrary alternative c that is ranked by both u and w. It is the case that

$$\begin{aligned} |\mathbf{has}_u(c)| &= t_{c,u} + \mathrm{convert}_c \\ &= \lambda_u(c) - t_{c,u,w} + \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}) \\ &= \underbrace{\lambda_u(c) - \min(\lambda_u(c), \lambda_w(c))}_{\geq 0} + \min(\underbrace{|A_w| - t_{c,w}}_{\geq 0}, \underbrace{|A_u| - t_{c,u}}_{\geq 0}, \underbrace{t_{c,u,w}}_{\geq 0}) \\ &\geq 0. \end{aligned}$$

The same argument holds for $|\mathbf{has}_w(c)|$. Additionally, it is the case that

$$\begin{aligned} |\mathbf{has}_{u,w}(c)| &= t_{c,u,w} - \mathrm{convert}_c \\ &= t_{c,u,w} - \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}) \\ &\geq 0. \end{aligned}$$

On the other hand, for an alternative c ranked only by u or only by w, $|\mathbf{has}_w(c)| \geq 0$, $|\mathbf{has}_u(c)| \geq 0$, and $|\mathbf{has}_{u,w}(c)| \geq 0$ by definition. ◇

Claim 25. *For every alternative $c \in C$,*

- $|\mathbf{has}_u(c)| \leq |A_u|$,
- $|\mathbf{has}_w(c)| \leq |A_w|$, *and*
- $|\mathbf{has}_{u,w}(c)| \leq |A_{u,w}|$.

For every alternative c, it is the case that $|A_{u,w}| \geq |\mathbf{has}_{u,w}(c)|$ by definition. Consider $|A_u|$; For any alternative $c \in C$, it is the case that $|A_u| \geq |\mathbf{has}_u(c)|$ because of the following. If c is ranked by both u and w, then

$$\begin{aligned} &|A_u| - |\mathbf{has}_u(c)| \\ &= |A_u| - t_{c,u} - \mathrm{convert}_c \\ &= |A_u| - t_{c,u} - \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}) \\ &\geq 0. \end{aligned}$$

Otherwise, if only u ranks c, then

$$\begin{aligned} &|A_u| - |\mathbf{has}_u(c)| \\ &= |A_u| - \min(\lambda_u(c), |A_u|) \\ &\geq 0. \end{aligned}$$

The same arguments can be made for $|A_w|$. ◇

Next, we show that with this construction, the scores of the alternatives correspond to their positions.

Claim 26. *In the above construction, each alternative c ranked by u has $\mathrm{score}_u(c) = \lambda_u(c)$. The analogous is true for $\mathrm{score}_w(c)$.*

First, suppose c is ranked by both u and w. Then,

$$\begin{aligned} \mathrm{score}_u(c) &= |\mathbf{has}_u(c)| + |\mathbf{has}_{u,w}(c)| \\ &= t_{c,u} - \mathrm{convert}_c + t_{c,u,w} + \mathrm{convert}_c \\ &= t_{c,u} + t_{c,u,w} \\ &= \lambda_u(c) - t_{c,u,w} + t_{c,u,w} \\ &= \lambda_u(c). \end{aligned}$$

Otherwise, suppose that c is ranked only by u. Then,

$$\begin{aligned}\text{score}_u(c) &= |\mathbf{has}_u(c)| + |\mathbf{has}_{u,w}(c)| \\ &= \min(\lambda_u(c), |A_u|) + \lambda_u(c) - |\mathbf{has}_u(c)| \\ &= \min(\lambda_u(c), |A_u|) + \lambda_u(c) - \min(\lambda_u(c), |A_u|) \\ &= \lambda_u(c).\end{aligned}$$

The analogous claim for the scores of w can be shown analogously. $\diamond$

This gives that Property (11) is fulfilled, as whenever $a \succ_u b$, then $\lambda_u(a) > \lambda_u(b)$, thus $\text{score}_u(a) > \text{score}_u(b)$.

Lastly, we show that k is the least number of attributes needed. To show this, we first have to prove the existence of two specific alternatives. We will show thereafter that the constellation of the two alternatives requires k attributes to be explained.

Claim 27. *There exists an alternative a for which*

$$|\mathbf{has}_{u,w}(a)| = |A_{u,w}| \textit{ and } |\mathbf{has}_u(a)| = |A_u| \tag{12}$$

or

$$|\mathbf{has}_{u,w}(a)| = |A_{u,w}| \textit{ and } |\mathbf{has}_w(a)| = |A_w|. \tag{13}$$

If $|A_{u,w}| = 0$, the alternative c for which $t_{c,u}$ is maximal clearly has Property 12. Thus, consider the case that $|A_{u,w}| > 0$. Let c be the alternative for which $|\mathbf{has}_{u,w}(c)|$ is maximal. By definition, $|A_{u,w}| = |\mathbf{has}_{u,w}(c)|$. We claim that c also **has** either all attributes of A_u or all of A_w.

If c is ranked by only u, then

$$|\mathbf{has}_{u,w}(c)| = |A_{u,w}| = \lambda_u(c) - \min(\lambda_u(c), |A_u|) > 0.$$

implies that $\min(\lambda_u(c), |A_u|) = |A_u|$, and coincidentally, $|\mathbf{has}_u(c)| = \min(\lambda_u(c), |A_u|)$ by definition. The analogous holds for when c is ranked only by w.

On the other hand, we argue for the case when c is ranked by both u and w. Assume that

$$|\mathbf{has}_u(c)| < |A_u| \text{ and } |\mathbf{has}_w(c)| < |A_w|,$$

aiming for a contradiction. From this, we get

$$\begin{aligned}|\mathbf{has}_u(c)| = t_{c,u} + \text{convert}_c < |A_u| \quad &\iff \\ \text{convert}_c < |A_u| - t_{c,u}.&\end{aligned}$$

and similarly that $\text{convert}_c < |A_w| - t_{c,w}$.

Since we also assume that $|A_{u,w}| > 0$, it is the case that

$$|\mathbf{has}_{u,w}(c)| = |A_{u,w}| = t_{c,u,w} - \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}) > 0$$

implying that

$$\text{convert}_c < t_{c,u,w}.$$

This is contradictory, as the following (in-)equalities

$$\begin{aligned}&\text{convert}_c = \min(|A_w| - t_{c,w}, |A_u| - t_{c,u}, t_{c,u,w}), \\ &\text{convert}_c < |A_w| - t_{c,w}, \\ &\text{convert}_c < |A_u| - t_{c,u}, \text{ and} \\ &\text{convert}_c < t_{c,u,w}\end{aligned}$$

can not be true simultaneously. This concludes the proof of the claim. $\diamond$

Claim 28. *The alternative $b_w := \text{argmax}_{c \in C}\, t_{c,w}$* **has** *all attributes of A_w and none of A_u, i.e.,*

$$|\mathbf{has}_w(b_w)| = |A_w| \textit{ and } |\mathbf{has}_u(b_w)| = 0.$$

The analogous holds for $b_u := \text{argmax}_{c \in C}\, t_{c,u}$.

We only show the statement for b_w. To see why it is true, first expand the definitions. Notice that $|A_w| = t_{b,w}$ per definition. From this follows that

$$\text{convert}_b = \min(|A_w| - t_{b,w}, |A_u| - t_{b,u}, t_{b,u,w}) = 0.$$

From this follows that

$$|\mathbf{has}_w(b)| = t_{b,w} + \text{convert}_b = |A_w|.$$

With these insights, assume that $|\mathbf{has}_u(b)| > 0$, aiming for a contradiction. This gives us

$$|\mathbf{has}_u(b)| = t_{b,u} + \text{convert}_b = t_{b,u} > 0$$

and thus because

$$t_{b,u} = \lambda_u(b) - t_{b,u,w} = \lambda_u(b) - \min(\lambda_u(b), \lambda_w(b)).$$

it must be the case that $\lambda_u(b) > \lambda_w(b)$, which implies that

$$\begin{aligned}|A_w| &= t_{b,w} \\ &= \lambda_w(b) - t_{c,u,w} \\ &= \lambda_w(b) - \min(\lambda_u(b), \lambda_w(b)) \\ &= 0,\end{aligned}$$

which we have assumed to be not the case. Thus, $|\mathbf{has}_u(b)| = 0$. $\diamond$

Claim 29. *Every BAM for $\mathcal{P}$ consists of at least k attributes.*

Let k' be the number of attributes for an arbitrary BAM for $\mathcal{P}$. Let a be the alternative a from Claim 27 and assume without loss of generality that

$$|\mathbf{has}_{u,w}(a)| = |A_{u,w}| \text{ and } |\mathbf{has}_u(a)| = |A_u|.$$

Lastly, let b be the alternative b_w from Claim 28, that is

$$|\mathbf{has}_w(b)| = |A_w| \text{ and } |\mathbf{has}_u(b)| = 0.$$

In general, every BAM needs to take into account at least the least needed score of a, i.e., $k' \geq \lambda_u(a)$. If $|A_w| = 0$, then $k' \geq \lambda_u(a) = |A_{u,w}| + |A_u| = k$, which means that the claim holds for this case. Thus, assume that $|A_w| > 0$.

Informally speaking, we claim that the constellation of a and b needs at least k attributes to be explained by every BAM.

To see why, first observe that, generally, it must be the case that $\lambda_u(a) \geq \lambda_u(b)$, since a **has** all of $A_{u,w}$ and A_u. Also, it must be the case that $\lambda_w(b) > 0$, because b **has** all A_w attributes and we assume that $|A_w| > 0$.

We distinguish between the cases that $\lambda_w(a) > \lambda_w(b)$ and $\lambda_w(a) < \lambda_w(b)$; it can not be the case that $\lambda_w(a) = \lambda_w(b)$, as then a and b would have a tie in w's preferences, which can not be an input of BAM-Existence per definition.

Case 1: $\lambda_w(a) > \lambda_w(b)$: First, note that w must rank a, as $\lambda_w(a)$ can not be strictly larger than $\lambda_w(b)$ unless $\lambda_w(a) \geq 1$, which is only the case whenever w ranks a.

Informally speaking, every BAM needs to explain both the larger gap between a and b, that is, there must be

$$\max(\lambda_u(a) - \lambda_u(b), \lambda_w(a) - \lambda_w(b))$$

attributes that a **has** and b does not **have**. Otherwise, the gaps between them could not be explained.

In addition to these attributes, there need to be

$$\max(\lambda_u(b), \lambda_w(b))$$

other attributes to explain how b is ranked where it is.

We claim that $\lambda_w(b) \geq \lambda_u(b)$. This is clearly the case if u does not rank b, as then $\lambda_u(b) = 0$. Otherwise, the statement holds, because,

$$\begin{aligned}
\lambda_w(b) \geq \lambda_u(b) && \iff \\
|\mathbf{has}_{u,w}(b)| + |\mathbf{has}_w(b)| \geq |\mathbf{has}_{u,w}(b)| + |\mathbf{has}_u(b)| && \iff \\
|\mathbf{has}_{u,w}(b)| + |A_w| \geq |\mathbf{has}_{u,w}(b)| + 0. &&
\end{aligned}$$

Putting these two insights together, i.e., that the gaps between a and b and b's position for w need to be explained by two disjoint sets of attributes by every k'-BAM for $\mathcal{P}$, we have that

$$\begin{aligned}
k' &\geq \max(\lambda_u(a) - \lambda_u(b), \lambda_w(a) - \lambda_w(b)) + \lambda_w(b) \\
&= \max(|A_u| + |A_{u,w}| - \lambda_u(b), |A_{u,w}| + |\mathbf{has}_w(a)| - \lambda_w(b)) + \\
&\quad \lambda_w(b) \\
&= |A_{u,w}| + \max(|A_u| - \lambda_u(b), |\mathbf{has}_w(a)| - \lambda_w(b)) + \lambda_w(b) \\
&\geq |A_{u,w}| + \max(|A_u| - |\mathbf{has}_{u,w}(b)| - |\mathbf{has}_u(b)|, \\
&\quad |\mathbf{has}_w(a)| - |\mathbf{has}_{u,w}(b)| - |\mathbf{has}_w(b)|) + \lambda_w(b) \\
&= |A_{u,w}| + \max(|A_u| - |\mathbf{has}_{u,w}(b)|, \\
&\quad |\mathbf{has}_w(a)| - |\mathbf{has}_{u,w}(b)| - |\mathbf{has}_w(b)|) + \lambda_w(b) \\
&= |A_{u,w}| - |\mathbf{has}_{u,w}(b)| + \max(|A_u|, |\mathbf{has}_w(a)| - |\mathbf{has}_w(b)|) + \\
&\quad \lambda_w(b) \\
&= |A_{u,w}| - |\mathbf{has}_{u,w}(b)| + \max(|A_u|, |\mathbf{has}_w(a)| - |A_w|) + \\
&\quad \lambda_w(b) \\
&= |A_{u,w}| - |\mathbf{has}_{u,w}(b)| + |A_u| + \lambda_w(b) \\
&= |A_{u,w}| - |\mathbf{has}_{u,w}(b)| + |A_u| + |\mathbf{has}_{u,w}(b)| + |\mathbf{has}_w(b)| \\
&= |A_{u,w}| + |A_u| + |A_w| \\
&= k.
\end{aligned}$$

Notice that the second inequality is due to the fact that u might not rank b.

Case 2: $\lambda_w(a) < \lambda_w(b)$: In this case, similar to above, *both* gaps need to be explained; there need to be $\lambda_u(a) - \lambda_u(b)$ attributes that a **has** that b does not **have**, and $\lambda_w(b) - \lambda_w(a)$ attributes that b **has** and a does not **have**.

In addition, both b's position for u and a's position for w need to be explained by distinct attributes.

Putting this together as was done previously, we obtain that for every k'-BAM for $\mathcal{P}$

$$\begin{aligned}
k' &\geq \lambda_u(a) - \lambda_u(b) + \lambda_w(b) - \lambda_w(a) + \max(\lambda_u(b), \lambda_w(a)) \\
&\geq \lambda_u(a) - \lambda_u(b) + \lambda_w(b) - \max(\lambda_u(b), \lambda_w(a)) + \\
&\quad \max(\lambda_u(b), \lambda_w(a)) \\
&= \lambda_u(a) - \lambda_u(b) + \lambda_w(b) \\
&\geq |A_{u,w}| + |A_u| - |\mathbf{has}_{u,w}(b)| - |\mathbf{has}_u(b)| + |\mathbf{has}_{u,w}(b)| + |\mathbf{has}_w(b)| \\
&= |A_{u,w}| + |A_u| + |A_w| \\
&= k.
\end{aligned}$$

In either case, $k' \geq k$. This concludes the proof of the claim. ◇

We have shown that there exists a linear algorithm that computes the least possible amount of attributes for a given preference profile $\mathcal{P}$ with $n = 2$. Thus, BAM-Existence is decidable in linear time for $n = 2$.

□