



Two approximation algorithms for probabilistic coalition structure generation with quality bound

Kouki Matsumura¹ · Bojana Kodric² · Tenda Okimoto³ · Katsutoshi Hirayama³

Published online: 18 February 2020
© The Author(s) 2020

Abstract

How to form effective coalitions is an important issue in multi-agent systems. Coalition Structure Generation (CSG) is a fundamental problem whose formalization can encompass various applications related to multi-agent cooperation. CSG involves partitioning a set of agents into coalitions such that the social surplus (i.e., the sum of the values of all coalitions) is maximized. In traditional CSG, we are guaranteed that all coalitions will be successfully established, that is, the attendance rate of each agent for joining any coalition is assumed to be 1.0. Having the real world in mind, however, it is natural to consider the uncertainty of agents' availabilities, e.g., an agent might be available only two or three days a week because of his/her own schedule. Probabilistic Coalition Structure Generation (PCSG) is an extension of CSG where the attendance type of each agent is considered. The aim of this problem is to find the optimal coalition structure which maximizes the sum of the expected values of all coalitions. In PCSG, since finding the optimal coalition structure easily becomes intractable, it is important to consider approximation algorithms, i.e., to consider a trade-off between the quality of the returned solution and tractability. In this paper, a formal framework for PCSG is introduced. Approximation algorithms for PCSG called Bounded Approximation Algorithm based on Attendance Types (BAAAT) and Involved BAAAT (IBAAAT) are then presented. We prove a priori bounds on the quality of the solution returned by BAAAT and IBAAAT with respect to the optimum and perform experimental evaluations on a number of benchmarks.

Keywords Multi-agent systems · Cooperative game theory · Coalition structure generation · Uncertainty · Bounded approximation algorithm

This paper is an extended and revised version of [13].

✉ Kouki Matsumura
matsumura@ai.sanken.osaka-u.ac.jp

Extended author information available on the last page of the article

Table 1 The employees necessary for completing a request and the associated reward

Request No.	Employees needed	Reward
1	{ <i>Ali</i> }	\$ 20
2	{ <i>Bob</i> }	\$ 50
3	{ <i>Chan</i> }	\$ 10
4	{ <i>Ali, Bob</i> }	\$ 70
5	{ <i>Ali, Chan</i> }	\$ 60
6	{ <i>Bob, Chan</i> }	\$ 100
7	{ <i>Ali, Bob, Chan</i> }	\$ 110

1 Introduction

Coalition Structure Generation (CSG) [1, 18, 23] is a key issue for various applications related to multi-agent cooperation, e.g., distributed vehicle routing [25], waste-water treatment system [7], forming rescue groups in disaster areas [30] and multi-sensor networks [5]. CSG involves partitioning a set of agents into coalitions (where each coalition is a subset of the available set of agents) such that the social surplus is maximized. A partition is also called a coalition structure. In traditional CSG, the value of a coalition is assumed to be given by a black box function called the *characteristic function*, and the value of a coalition structure is provided by the sum of the values of all coalitions. It is well-known that the CSG problem is equivalent to the *complete set partitioning problem* [33]. Various sophisticated algorithms have been proposed for the CSG problem, for instance, an anytime algorithm with a worst-case guarantee [24], a dynamic programming based algorithm [33], search/dynamic programming based anytime algorithms [19, 22, 27], and a hybrid algorithm called ODP-IP which is a state-of-the-art algorithm that works by combining dynamic programming and anytime algorithms [14].

Let us consider the following simple scenario. There is a service company dispatching interpreters with three employees, *Ali*, *Bob* and *Chan*. Now, this service company has received seven job requests for simultaneous interpretation, each requiring some specific language skills. Table 1 shows the rewards of each job request.

Assume that you are the manager of this service company and want to assign employees to jobs such that the sum of the rewards is maximized. Then, this problem can be represented as an instance of the CSG problem. If you assign three employees *Ali*, *Bob* and *Chan* to job requests 1, 2 and 3 separately, the sum of the rewards obtained by the coalition structure $\{\{Ali\}, \{Bob\}, \{Chan\}\}$ is $\$20 + \$50 + \$10 = \80 . When you assign *Ali* to request 1, and *Bob* and *Chan* to request 6, the social surplus is maximized. In this case, the coalition structure is $\{\{Ali\}, \{Bob, Chan\}\}$ and the service company earns $\$20 + \$100 = \$120$.

In what follows, we are interested in the uncertainty of agents' attendances. In traditional CSG, one does need to worry about whether all coalitions will be established or not, i.e., the probability of each agent to join any coalition he is assigned to is assumed to be 1.0. However, if we want to mimic the real world, it is natural to consider the uncertainty of agents' availabilities. For instance, it might happen that an agent is only available on some days or the week or cannot commit in advance with certainty.

In our example of a service company, it is natural that the manager asks the three employees about their schedules before assigning them to jobs. Also, setting an exact probability of each agent to join a coalition is not realistic, e.g., when *Ali* thinks that he can

probably fit job request 1 into his schedule, he will report to the manager “I can probably work for request 1” but will not answer “I can work for it with probability 66%”. Here, or more specifically in the experimental part, we assume that (1) each agent chooses in advance one of the following attendance types (including the probabilities/attendance rates) according to his/her own schedule¹:

- Type 1: {available (90%)},
- Type 2: {probably available (70%)},
- Type 3: {unsure (50%)},
- Type 4: {probably not available (30%)},
- Type 5: {not available (10%)},

and (2) the CSG maker (e.g. the manager of the service company in our example) knows all the information about the agents’ types. Such simplification leads to a lack of generality in the sense that we allow the agents’ availabilities to be chosen only from a small set of possible types, while the PCSG model allows for an infinite number of agent types. Still, it is arguably reasonable when considering applications, since allowing the agents/the CSG maker to be able to use the whole interval $[0, 1]$ when setting the availability probabilities does not seem sensible, as already discussed above.

In this paper, the main focus is laid on the *Probabilistic Coalition Structure Generation* (PCSG) problem which is an extension of CSG where the availability type of each agent is considered. First, a formal framework for PCSG which generalizes the one from [26] is introduced. The aim is to find a coalition structure that maximizes the sum of the expected values of all coalitions. Any such coalition structure will be called optimal. What exactly is meant under the term “expected value of a coalition” is an important issue here. In our framework, the expected value of a coalition is parameterized by the parameter k . If up to k agents are missing from a coalition, the contribution of the remaining agents can be read off from the characteristic function. If the number of missing agents is higher than k , we assume that the contribution of the remaining agents is 0, as their (sub)coalition differs too much from the coalition originally planned. Note that when $k = n - 1$, any subset of the original coalition gives its contribution to the expected coalition value, and this corresponds to Flexible PCSG [26]. On the other hand, if $k = 0$, the model corresponds to Cautious PCSG [26]. To illustrate the influence of k , let us consider the expected value of the coalition $\{Ali, Bob, Chan\}$ for $k = 1$ (i.e., we allow one agent to be missing). The expected value is computed by taking into consideration the following cases (1) *Ali, Bob* and *Chan* are available, (2) *Ali* and *Bob* are available, (3) *Ali* and *Chan* are available and (4) *Bob* and *Chan* are available.

Furthermore, we present approximation algorithms for solving the PCSG problem called Bounded Approximation Algorithm based on Attendance Types (BAAAT) and Involved Bounded Approximation Algorithm based on Attendance Types (IBAAAT). In the PCSG problem, since finding the optimal coalition structure becomes easily intractable, it is important to consider fast but approximate algorithms. The basic idea of the proposed algorithm BAAAT is that for a given parameter \bar{p} , (1) agents whose attendance probability is below the parameter \bar{p} are placed into singleton coalitions, (2) an optimal coalition structure of the relaxed problem with the remaining agents is computed. On the other hand,

¹ How the agents decide which one of the offered types of probability/attendance rate they will choose is beyond the scope of this paper.

IBAAAT has the following two steps: (1) agents are split into two sets such that one is the set of agents whose attendance probability is below the parameter \bar{p} and the other is the set of all the remaining agents, (2) optimal coalition structures of the two relaxed problems are computed. We also prove an upper bound on the error of the solution returned by BAAAT and IBAAAT with respect to the optimum. The error bound is a theoretical worst-case bound that is obtained a priori, that is, before actually running the algorithm. Finally, the performances of BAAAT and IBAAAT are evaluated on a number of benchmarks.

The rest of the paper is organized as follows. In Sect. 2, the Coalition Structure Generation (CSG) problem is briefly described. Section 3 introduces a formal framework for the Probabilistic Coalition Structure Generation (PCSG) problem that will be used in the rest of the paper. In Sect. 4, approximation algorithms for the PCSG problem called Bounded Approximation Algorithm based on Attendance Types (BAAAT) and Involved Bounded Approximation Algorithm based on Attendance Types (IBAAAT) are presented, together with their approximation guarantees. In Sect. 5, BAAAT and IBAAAT are evaluated on a number of benchmarks. Section 6 discusses related work, and, finally, Sect. 7 concludes the paper.

2 Coalition structure generation

We briefly describe the Coalition Structure Generation (CSG) problem [1, 18, 23]. CSG involves partitioning a set of agents into coalitions such that the social surplus (i.e., the sum of the values of all coalitions) is maximized. Let us start with some preliminary definitions.

Let $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ be a finite set of agents. A coalition from \mathcal{A} , denoted by C , is a non-empty subset of \mathcal{A} . A coalition structure on \mathcal{A} , denoted by CS, is a partition of \mathcal{A} , that is, a jointly exhaustive set of pairwise disjoint coalitions from \mathcal{A} . More formally, a coalition structure on \mathcal{A} is a finite set of coalitions satisfying the following two conditions:

1. $\forall i, j \in \{1, 2, \dots, m\}, i \neq j, C_i \cap C_j = \emptyset$,
2. $\bigcup_{C_i \in \text{CS}} C_i = \mathcal{A}$.

In other words, each agent belongs to exactly one coalition. Note that some agents may be alone in their coalitions. In our running example of a service company with three employees, there exist seven possible coalitions (i.e., $\{Ali\}, \{Bob\}, \{Chan\}, \{Ali, Bob\}, \{Ali, Chan\}, \{Bob, Chan\}, \{Ali, Bob, Chan\}$) and the following five coalition structures:

$$\{\{Ali\}, \{Bob\}, \{Chan\}\}, \{\{Ali\}, \{Bob, Chan\}\}, \{\{Bob\}, \{Ali, Chan\}\}, \\ \{\{Chan\}, \{Ali, Bob\}\}, \{\{Ali, Bob, Chan\}\}.$$

The Coalition Structure Generation problem description is defined as follows:

Definition 1 (CSG problem description) A coalition structure generation problem description is defined by a pair $\text{CSG} = \langle \mathcal{A}, v \rangle$ where $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ is a set of agents and $v : 2^{\mathcal{A}} \rightarrow \mathbb{R}$ is a function called the characteristic function.

The value of a coalition C , denoted by $v(C)$, is given by the characteristic function v . The value of a coalition structure CS, denoted by $V(\text{CS})$, is provided by the sum of the values of all coalitions, i.e.,

Table 2 The obtained rewards of all possible coalition structures

Coalition structure	Reward
$\{\{Ali\}, \{Bob\}, \{Chan\}\}$	\$ 80
$\{\{Ali\}, \{Bob, Chan\}\}$	\$ 120
$\{\{Bob\}, \{Ali, Chan\}\}$	\$ 110
$\{\{Chan\}, \{Ali, Bob\}\}$	\$ 80
$\{\{Ali, Bob, Chan\}\}$	\$ 110

$$V(\text{CS}) = \sum_{C_i \in \text{CS}} v(C_i). \quad (1)$$

A coalition structure is said to be optimal, denoted by CS^* , if it maximizes the social surplus, that is, if it satisfies the following condition:

$$\forall \text{CS}, V(\text{CS}) \leq V(\text{CS}^*).$$

Example 1 (CSG) Consider the service company with three employees *Ali*, *Bob* and *Chan* introduced in the previous section. Assume that you are the manager of this service company and want to assign the employees to jobs such that the sum of the rewards from Table 1 is maximized. Then, this problem can be represented as an instance of the CSG problem: let $\text{CSG} = \langle \mathcal{A}, v \rangle$ be a CSG problem description with $\mathcal{A} = \{Ali, Bob, Chan\}$, and the function v is then characterized as follows:

$$\begin{aligned} v(\{Ali\}) &= 20, v(\{Bob\}) = 50, v(\{Chan\}) = 10, \\ v(\{Ali, Bob\}) &= 70, v(\{Ali, Chan\}) = 60, v(\{Bob, Chan\}) = 100, \\ v(\{Ali, Bob, Chan\}) &= 110. \end{aligned}$$

Table 2 shows the rewards associated with all possible coalition structures. The optimal coalition structure in this example is $\text{CS}^* = \{\{Ali\}, \{Bob, Chan\}\}$, and the obtained reward is $V(\text{CS}^*) = v(\{Ali\}) + v(\{Bob, Chan\}) = \$20 + \$100 = \120 . As illustrated by this running example, the characteristic function v is supposed to be represented extensively, as the set of pairs $\{(C, v(C)) \mid C \subseteq \mathcal{A} \text{ and } C \neq \emptyset\}$.

The CSG problem is defined as follows:

Definition 2 (CSG problem)

- *Input* A CSG problem description $\text{CSG} = \langle \mathcal{A}, v \rangle$,
- *Question* Find an optimal coalition structure CS^* .

Let us now focus on the specific cases of the CSG problem, when the characteristic function v is *subadditive* or *superadditive*. For a CSG problem description $\text{CSG} = \langle \mathcal{A}, v \rangle$, the characteristic function v is said to be subadditive if for any coalitions C_i and C_j with $C_i \cap C_j = \emptyset$, it holds that $v(C_i) + v(C_j) \geq v(C_i \cup C_j)$. Conversely, v is called superadditive if it holds that $v(C_i) + v(C_j) \leq v(C_i \cup C_j)$. It is well-known that in case v is subadditive, the coalition structure formed by singleton coalitions is optimal,

i.e., $CS^* = \{\{a_i\} \mid a_i \in \mathcal{A}\}$. For the superadditive case, the grand coalition is optimal, namely $CS^* = \{\mathcal{A}\}$ [24].

3 Probabilistic coalition structure generation

We now introduce the Probabilistic Coalition Structure Generation (PCSG) problem, which will be in focus of this work. As opposed to the traditional CSG, where we are guaranteed that all coalitions will be established, we want to have the real world in mind. So, it is natural to consider the uncertainty of agents' availabilities. PCSG is an extension of CSG where exactly the attendance type of each agent is considered. The aim is to find an optimal coalition structure that maximizes the sum of the expected values of all coalitions. Note that it is not a priori clear how one should arrive at the expected value of a coalition. Therefore, an important issue in PCSG is how exactly the expected value of a coalition is computed. In our framework, the computation of the expected value depends on the parameter k . If up to k agents are missing from a coalition, the contribution of the remaining agents can be read off from the characteristic function. If the number of missing agents is higher than k , we assume that the contribution of the remaining agents is 0, as their (sub)coalition differs too much from the coalition originally planned. When $k = n - 1$, any subset of the original coalition gives its contribution to the expected coalition value, and this corresponds to Flexible PCSG [26]. On the other hand, if $k = 0$, the model corresponds to Cautious PCSG [26].

In the experimental part, we in addition assume that (1) each agent chooses in advance one of the following attendance types (including the probabilities/attendance rates) according to his/her own schedule:

- Type 1: {available (90%)},
- Type 2: {probably available (70%)},
- Type 3: {unsure (50%)},
- Type 4: {probably not available (30%)},
- Type 5: {not available (10%)},

and (2) the CSG maker (e.g. the manager of the service company in our example) knows all the information about the agents' types.

The Probabilistic Coalition Structure Generation problem description is then defined as follows:

Definition 3 (PCSG problem description) A probabilistic coalition structure generation problem description is defined by a tuple $PCSG = \langle \mathcal{A}, v, f, k \rangle$ where $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ is a set of agents, $v : 2^{\mathcal{A}} \rightarrow \mathbb{R}$ is a characteristic function, $f : \mathcal{A} \rightarrow [0, 1]$ is a function that gives the probability/attendance rate of each agent and k is an integer, $0 \leq k \leq n - 1$.

Here, the participation of each agent a is a binary random variable that takes value 1 with probability $f(a)$ and value 0 with the remaining probability. Furthermore, we assume these random variables to be mutually independent, that is, the participation of one agent has no influence on those of other agents.

For a coalition C , let $\tilde{A} \subseteq C$ be the set of absent agents where $|\tilde{A}| \leq k$, $0 \leq k \leq n - 1$. For any such \tilde{A} , the coalition that remains after removing \tilde{A} from C is denoted by $C \setminus \tilde{A}$,

and the value of this coalition is given by $v(C \setminus \tilde{A})$. The contribution of this coalition to the expected value of coalition C , denoted by $v_e^k(C, \tilde{A})$, is

$$v_e^k(C, \tilde{A}) = v(C \setminus \tilde{A}) \cdot \prod_{a \in C \setminus \tilde{A}} f(a) \cdot \prod_{a' \in \tilde{A}} (1 - f(a')). \tag{2}$$

If $|\tilde{A}| > k$, then we define $v_e^k(C, \tilde{A}) = 0$.

Now, the expected value of a coalition C , denoted by $v_{e,k}(C)$, is given by

$$v_{e,k}(C) = \sum_{\tilde{A} \in 2^C} v_e^k(C, \tilde{A}). \tag{3}$$

Finally, the expected value of a coalition structure CS, denoted by $V_{e,k}(\text{CS})$, is computed as the sum of the expected values of all coalitions, i.e.

$$V_{e,k}(\text{CS}) = \sum_{C \in \text{CS}} v_{e,k}(C). \tag{4}$$

A coalition structure is said to be optimal, denoted by $\text{CS}_{e,k}^*$, if it maximizes the sum of the expected values of all coalitions, i.e. if the following condition holds:

$$\forall \text{CS}, V_{e,k}(\text{CS}) \leq V_{e,k}(\text{CS}_{e,k}^*).$$

Example 1 (continued) Consider our running example of a service company with three employees. Assume that *Ali* reported Type 1 (i.e., {available (90%)}), *Bob* chose Type 3 (i.e., {unsure (50%)}), and *Chan* selected Type 4 (i.e., {probably not available (30%)}). Moreover, the manager sets the parameter $k = 1$, that is, he/she wants to maximize the expected value of a coalition structure where at most one employee may be absent from a coalition in order for the remaining agents to still have a positive contribution to the expected coalition value. The expected value of each coalition is then computed by using the rewards from Table 1 as in Eqs. (2) and (3):

$$\begin{aligned} v_{e,1}(\{Ali\}) &= 18, v_{e,1}(\{Bob\}) = 25, v_{e,1}(\{Chan\}) = 3, \\ v_{e,1}(\{Ali, Bob\}) &= 43, v_{e,1}(\{Ali, Chan\}) = 29.1, v_{e,1}(\{Bob, Chan\}) = 34, \\ v_{e,1}(\{Ali, Bob, Chan\}) &= 46.5. \end{aligned}$$

Notice that when compared to Flexible PCSG [26], the parameter $k = 1$ only has an influence on the expected value of the grand coalition in this example. All other coalitions have size of at most 2, such that if more than one agents is missing, the set of the remaining agents is empty, and the contribution of the empty set is 0, independently of k . Consider now the grand coalition (i.e., {*Ali, Bob, Chan*}). The positive contributions to the expected value of the grand coalition are given as follows:

- *Ali*, *Bob* and *Chan* are available:
 $v(\{Ali, Bob, Chan\}) \cdot f(Ali) \cdot f(Bob) \cdot f(Chan) = 110 \cdot (0.9 \cdot 0.5 \cdot 0.3) = 14.85$
- *Ali* and *Bob* are available:
 $v(\{Ali, Bob\}) \cdot f(Ali) \cdot f(Bob) \cdot (1 - f(Chan)) = 70 \cdot (0.9 \cdot 0.5 \cdot 0.7) = 22.05.$
- *Ali* and *Chan* are available:
 $v(\{Ali, Chan\}) \cdot f(Ali) \cdot (1 - f(Bob)) \cdot f(Chan) = 60 \cdot (0.9 \cdot 0.5 \cdot 0.3) = 8.1$

Table 3 The expected values of all possible coalition structures when $k = 1$

Coalition structure	Expected value
$\{\{Ali\}, \{Bob\}, \{Chan\}\}$	\$ 46
$\{\{Ali\}, \{Bob, Chan\}\}$	\$ 52
$\{\{Bob\}, \{Ali, Chan\}\}$	\$ 54.1
$\{\{Chan\}, \{Ali, Bob\}\}$	\$ 46
$\{\{Ali, Bob, Chan\}\}$	\$ 46.5

– *Bob* and *Chan* are available:
 $v(\{Bob, Chan\}) \cdot (1 - f(Ali)) \cdot f(Bob) \cdot f(Chan) = 100 \cdot (0.1 \cdot 0.5 \cdot 0.3) = 1.5$

The expected value of the grand coalition is then $v_{e,1}(\{Ali, Bob, Chan\}) = 14.85 + 22.05 + 8.1 + 1.5 = 46.5$.

Table 3 shows the expected values of all possible coalition structures for $k = 1$. Compared to the optimal coalition structure $CS^* = \{\{Ali\}, \{Bob, Chan\}\}$ in our CSG example, the optimal coalition structure here is $CS_{e,1}^* = \{\{Bob\}, \{Ali, Chan\}\}$, and the expected value obtained by $CS_{e,1}^*$ is $V_e(CS_{e,1}^*) = v_{e,1}(\{Bob\}) + v_{e,1}(\{Ali, Chan\}) = 25 + 29.1 = 54.1$.

The PCSG problem is defined as follows:

Definition 4 (PCSG problem)

- *Input* A PCSG problem description $PCSG = \langle A, v, f, k \rangle$
- *Question* Find an optimal coalition structure $CS_{e,k}^*$.

The PCSG problem is a generalization of the CSG problem. In case the attendance rate of each agent is 1.0, that is, it is guaranteed that every agent will join any coalition he/she is assigned to, then the PCSG problem reduces to the standard CSG problem. This fact is independent of the choice of parameter k , i.e., it holds for any $k, 0 \leq k \leq n - 1$.

Let us now focus on the special case of the PCSG problem when the characteristic function v is subadditive and $k = n - 1$. In that case, the analog of the result in the standard CSG framework also holds in the PCSG framework (which for $k = n - 1$ corresponds to Flexible PCSG [26]). More precisely, if v is subadditive, the coalition structure formed by singleton coalitions is optimal (i.e., $CS_{e,n-1}^* = \{\{a_i\} | a_i \in A\}$) [26].

We now investigate the influence of the parameter k on the optimal solution in the PCSG problem, when the characteristic function v is subadditive.

Proposition 1 *Let $PCSG = \langle A, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. If v is subadditive, then the coalition structure formed by singleton coalitions is optimal for any $k \in [0, n - 1]$. More precisely,*

$$CS_{e,k}^* = \{\{a_i\} | a_i \in A\}, \forall k \in [0, n - 1].$$

Proof Since v is subadditive, we know that for any $C_1, C_2 \subseteq A$ such that $C_1 \cap C_2 = \emptyset$, $v(C_1) + v(C_2) \geq v(C_1 \cup C_2)$. We want to compare $v_{e,k}(C_1) + v_{e,k}(C_2)$ and $v_{e,k}(C_1 \cup C_2)$. Namely, if we prove that $v_{e,k}(C_1) + v_{e,k}(C_2) \geq v_{e,k}(C_1 \cup C_2)$, the claim will follow. We know that

$$v_{e,k}(C_1 \cup C_2) = \sum_{\substack{\tilde{C} \subseteq (C_1 \cup C_2) \\ |\tilde{C}| \leq k}} \left(v((C_1 \cup C_2) \setminus \tilde{C}) \cdot \prod_{c \in (C_1 \cup C_2) \setminus \tilde{C}} f(c) \cdot \prod_{c' \in \tilde{C}} (1 - f(c')) \right). \tag{5}$$

Now, notice that for every $\tilde{C} \subseteq (C_1 \cup C_2), |\tilde{C}| \leq k$, there exist exactly one $\tilde{C}_1 \subseteq C_1$ and exactly one $\tilde{C}_2 \subseteq C_2$ such that $\tilde{C} = \tilde{C}_1 \cup \tilde{C}_2$. Then,

$$\begin{aligned} & v((C_1 \cup C_2) \setminus \tilde{C}) \cdot \prod_{c \in (C_1 \cup C_2) \setminus \tilde{C}} f(c) \cdot \prod_{c' \in \tilde{C}} (1 - f(c')) \\ &= v((C_1 \cup C_2) \setminus (\tilde{C}_1 \cup \tilde{C}_2)) \cdot \prod_{c \in (C_1 \cup C_2) \setminus (\tilde{C}_1 \cup \tilde{C}_2)} f(c) \cdot \prod_{c' \in \tilde{C}_1 \cup \tilde{C}_2} (1 - f(c')) \\ &\leq (v(C_1 \setminus \tilde{C}_1) + v(C_2 \setminus \tilde{C}_2)) \cdot \prod_{c \in (C_1 \cup C_2) \setminus (\tilde{C}_1 \cup \tilde{C}_2)} f(c) \cdot \prod_{c' \in \tilde{C}_1 \cup \tilde{C}_2} (1 - f(c')) \tag{6} \\ &\leq v(C_1 \setminus \tilde{C}_1) \cdot \prod_{c \in C_1 \setminus \tilde{C}_1} f(c) \cdot \prod_{c' \in \tilde{C}_1} (1 - f(c')) \\ &\quad + v(C_2 \setminus \tilde{C}_2) \cdot \prod_{c \in C_2 \setminus \tilde{C}_2} f(c) \cdot \prod_{c' \in \tilde{C}_2} (1 - f(c')). \end{aligned}$$

But now, since $|\tilde{C}_1| \leq k$ and $|\tilde{C}_2| \leq k$, the two summands in the last expression of Eq. (6) must also appear as summands in $v_{e,k}(C_1)$ and $v_{e,k}(C_2)$, respectively. This means that every summand in Eq. (5) is upper bounded by one summand of $v_{e,k}(C_1)$ and one summand of $v_{e,k}(C_2)$. Therefore, we can conclude that $v_{e,k}(C_1) + v_{e,k}(C_2) \geq v_{e,k}(C_1 \cup C_2)$. \square

When v is superadditive, on the other hand, we cannot come to an immediate conclusion that $CS_{e,k}^* = \{\mathcal{A}\}$. As a preliminary example, let us focus on the case when $k = 0$, which corresponds to Cautious PCSG [26]. Going back to our running example where $v(\{Ali\}) = 20, v(\{Chan\}) = 10, v(\{Ali, Chan\}) = 60$ and $f(Ali) = 0.9, f(Chan) = 0.3$, we see that v is superadditive on the domain consisting just of *Ali* and *Chan*. However, for $k = 0$, we can easily check that $v_{e,0}(\{Ali\}) = 20 \cdot 0.9 = 18, v_{e,0}(\{Chan\}) = 10 \cdot 0.3 = 3$ and

$$v_{e,0}(\{Ali, Chan\}) = 60 \cdot 0.9 \cdot 0.3 = 16.2 < 21 = v_{e,0}(\{Ali\}) + v_{e,0}(\{Chan\}),$$

meaning that superadditivity of v does not transfer to superadditivity of $v_{e,0}$, and the grand coalition is not an optimal coalition structure. We prove a more general statement in Proposition 2.

Proposition 2 *For every $k \in [0, n - 2]$, there exists an instance of the PCSG problem with problem description $PCSG = \langle \mathcal{A}, v, f, k \rangle$ such that v is superadditive and the grand coalition is not an optimal solution.*

Proof Let $n = |\mathcal{A}|$ and $f(a_i) = p > 0, \forall a_i \in \mathcal{A}$. Furthermore, let $v(C) = \frac{|C|}{n}, \forall C \subseteq \mathcal{A}$. Note that v is superadditive. Then,

$$\begin{aligned}
 v_{e,k}(\mathcal{A}) &= \sum_{\substack{C \subseteq \mathcal{A} \\ |C| \leq k}} v(\mathcal{A} \setminus C) \cdot \prod_{c \in \mathcal{A} \setminus C} f(c) \cdot \prod_{c' \in C} (1 - f(c')) \\
 &\leq \sum_{\substack{C \subseteq \mathcal{A} \\ |C| \leq n-2}} v(\mathcal{A} \setminus C) \cdot \prod_{c \in \mathcal{A} \setminus C} f(c) \cdot \prod_{c' \in C} (1 - f(c')) \\
 &= \sum_{i=1}^{n-2} \binom{n}{i} p^{n-i} (1-p)^i \\
 &= p \cdot (1 - (1-p)^{n-1}) \\
 &< p \\
 &= \sum_{a_i \in \mathcal{A}} v_{e,k}(\{a_i\}),
 \end{aligned}$$

where the strict inequality follows from $(1-p)^{n-1} > 0, \forall p > 0$. Since the coalition structure that consists of all singletons has a higher value than the grand coalition, we know that $CS_{e,k}^* \neq \{\mathcal{A}\}, \forall k \in [0, n-2]$. □

The case $k = n - 1$ needs to be treated separately, as it yields a different result, that is aligned with the one for superadditivity in the standard CSG framework.

Proposition 3 *Let $PCSG = \langle A, v, f, n - 1 \rangle$ be a probabilistic coalition structure generation problem description. If v is superadditive, then the grand coalition is the optimal solution. More precisely,*

$$CS_{e,n-1}^* = \{\{a_i\} | a_i \in \mathcal{A}\}.$$

Proof Let $CS = \{C_1, \dots, C_m\}$ be any coalition structure different from the grand coalition. The result follows by observing that for any realisation $A \subseteq \mathcal{A}$ of the set of the available agents, because of superadditivity of v , we know that

$$V(\{A\}) = v(A) \geq v(A_1) + \dots + v(A_m) = V(CS),$$

where $A_i = C_i \cap A$. □

Finally, let us note that an instance of the PCSG problem can be represented as a zero-one integer program in the same way as for the CSG problem, if we know the $v_{e,k}$ values of all coalitions. For simplicity, we show the zero-one integer programming formulation of our running example for PCSG. The objective function and the constraints are formalized as follows:

$$\max (18 \cdot a_1 + 25 \cdot a_2 + 3 \cdot a_3 + 43 \cdot a_{12} + 29.1 \cdot a_{13} + 34 \cdot a_{23} + 46.5 \cdot a_{123}), \tag{7}$$

$$s.t \ a_1 + a_{12} + a_{13} + a_{123} = 1, \tag{8}$$

$$a_2 + a_{12} + a_{23} + a_{123} = 1, \tag{9}$$

$$a_3 + a_{13} + a_{23} + a_{123} = 1, \quad (10)$$

$$a_1, a_2, a_3, a_{12}, a_{13}, a_{23}, a_{123} \in \{0, 1\}. \quad (11)$$

Variables a_1, a_2, \dots, a_{123} represent all possible coalitions, e.g., a_1 is the coalition $\{Ali\}$, and a_{123} represents the grand coalition (i.e., $\{Ali, Bob, Chan\}$) and can take values 0 or 1 [Eq. (11)]. Equation (8) describes that *Ali* belongs to one of the coalitions $\{Ali\}, \{Ali, Bob\}, \{Ali, Chan\}, \{Ali, Bob, Chan\}$, and he cannot belong to more than one coalition simultaneously. Similarly, Eqs. (9) and (10) show the constraints for *Bob* and *Chan*. Equation (7) represents the objective function which maximizes the sum of the expected values of all coalitions, and each coefficient shows the expected value obtained by the corresponding coalition, e.g., 18 is the expected value of the coalition $\{Ali\}$ and 46.5 is the expected value of the grand coalition.

4 Bounded approximation algorithms

In this section, we present two approximation algorithms for solving the PCSG problem called Bounded Approximation Algorithm based on Attendance Types (BAAAT) and Involved Bounded Approximation Algorithm based of Attendance Types (IBAAAT). Even though our algorithms can be seen as simple heuristics, we term them algorithms as they offer a theoretical bound on the quality of the returned solutions.

Let us for a moment assume that we can get the $v_{e,k}$ function as input, instead of just the characteristic function v and parameter k . Even though this assumption saves a lot of computation that would be necessary to arrive at $v_{e,k}$, still the input, i.e., the representation size, is exponential in the number of agents. Thus, it is important to consider approximation algorithms, i.e., to consider a trade-off between the quality of the returned solution and tractability. To this end, we prove an a priori upper bound on the error of the solution returned by both BAAAT and IBAAAT, i.e., the error bound is obtained before actually running the algorithm.

4.1 Approximation algorithm BAAAT

BAAAT has the following two phases:

Phase 1: For a given parameter $\tilde{p} \in \mathbb{R}$ and the attendance rates of all agents, form singleton coalitions for every agent a whose attendance rate p is such that $p \leq \tilde{p}$.

Phase 2: Find an optimal coalition structure of the relaxed PCSG problem with the remaining agents.

The basic idea of BAAAT is that for a given parameter \tilde{p} , the singleton coalition is formed for an agent who chooses the attendance type where the given probability/attendance rate is less than or equal to the parameter \tilde{p} . Then, an optimal coalition structure of the relaxed problem with the remaining agents is computed. We denote the coalition structure on the subset of agents that remain after singletons are formed in Phase 1, which is obtained in Phase 2, by $CS_{e,k}^-$ and the coalition structure provided by BAAAT, which is a coalition structure on the whole set of agents, by $CS_{e,k}^+$.

Let us further explain how BAAAT computes an approximate solution by using our running example. We are given the attendance type (that includes the attendance rate) of each agent, i.e., $f(Ali) = 0.9$ (Type1: available), $f(Bob) = 0.5$ (Type 3: unsure), $f(Chan) = 0.3$ (Type 4: probably not available), the parameter $k = 1$, and the following expected values of coalitions:

$$\begin{aligned} v_{e,1}(\{Ali\}) &= 18, v_{e,1}(\{Bob\}) = 25, v_{e,1}(\{Chan\}) = 3, \\ v_{e,1}(\{Ali, Bob\}) &= 43, v_{e,1}(\{Ali, Chan\}) = 29.1, v_{e,1}(\{Bob, Chan\}) = 34, \\ v_{e,1}(\{Ali, Bob, Chan\}) &= 46.5. \end{aligned}$$

Let $\tilde{p} = 0.3$, i.e., the manager does not count on agents who chose attendance Types 4 (i.e., probably not available (30%)) and 5 (i.e., not available (10%)). Since Chan reported attendance Type 4, i.e., $f(Chan) = 0.3 = \tilde{p}$, the singleton coalition is formed for Chan in Phase 1 of BAAAT. Then, the relaxed PCSG problem with the remaining agents (i.e., Ali and Bob) is solved in Phase 2 of BAAAT.² That is, the coalitions which include Chan can be ignored in the simplified problem and only the following is still relevant:

$$v_{e,1}(\{Ali\}) = 18, v_{e,1}(\{Bob\}) = 25, v_{e,1}(\{Ali, Bob\}) = 43.$$

Since the expected value of the coalition formed by Ali and Bob is equal to the sum of the expected values of singleton coalitions with Ali and Bob, that is, $v_{e,1}(\{Ali, Bob\}) = 43 = v_{e,1}(\{Ali\}) + v_{e,1}(\{Bob\})$, the optimal coalition structures of the relaxed problem are $\{Ali, Bob\}$ and $\{\{Ali\}, \{Bob\}\}$. The solutions obtained by BAAAT are $CS_{e,1}^{+1} = \{\{Chan\}, \{Ali, Bob\}\}$ and $CS_{e,1}^{+2} = \{\{Ali\}, \{Bob\}, \{Chan\}\}$. On this instance, there are two possible solutions, but their expected value is the same, namely, $V_e(CS_{e,1}^+) = 46$.

4.2 Quality guarantee of BAAAT

We show that we can provide an upper bound on the error of the solution returned by BAAAT a priori, i.e., the error bound is obtained before actually running the algorithm. Let us denote by \mathcal{C}_a the set of all coalitions that contain agent a as a member, and let $\tilde{\mathcal{A}}$ be the set of agents whose probability is lower than or equal to a given parameter \tilde{p} such that they form their own coalition in Phase 1. Furthermore, let $r_a^{\max} = \max\{v_{e,k}(C) \mid C \in \mathcal{C}_a\}$, i.e., r_a^{\max} is the maximal expected value of all coalitions which include agent a .

Lemma 1 *Let $PCSG = \langle \mathcal{A}, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. For an optimal coalition structure $CS_{e,k}^*$ and a coalition structure $CS_{e,k}^-$ obtained by BAAAT in Phase 2, the following inequality holds:*

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^-) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max}. \tag{12}$$

Proof We prove the claim by induction on the size of $\tilde{\mathcal{A}}$. In the base case, $\tilde{\mathcal{A}} = \emptyset$. Then, since no agents are removed in Phase 1 of BAAAT, the whole instance is solved to optimality in Phase 2. This means that

² One can apply any complete algorithm in Phase 2 for finding an optimal coalition structure.

$$V_{e,k}(CS_{e,k}^*) = V_{e,k}(CS_{e,k}^-).$$

On the other hand, $\sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max} = 0$, so

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^-) = 0 \leq 0 = \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max}$$

and inequality (12) holds.

Let us now assume that inequality (12) holds for all $\tilde{\mathcal{A}}$ such that $|\tilde{\mathcal{A}}| = \ell$. More specifically, we assume that

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^-) \leq \sum_{i=1}^{\ell} r_{\tilde{a}_i}^{\max}, \forall \tilde{\mathcal{A}}, |\tilde{\mathcal{A}}| = \ell. \tag{13}$$

Next, we consider the case where $\tilde{\mathcal{A}} = \{\tilde{a}_1, \dots, \tilde{a}_\ell, \tilde{a}_{\ell+1}\}$. We first observe the coalitions that form the expected optimal coalition structure $CS_{e,k}^*$ and denote these by $CS_{e,k}^* = \{C_{e,k}^{*1}, \dots, C_{e,k}^{*m}\}$. We also know that \tilde{a}_1 must be in one of these coalitions and without loss of generality we can assume that $C_{e,k}^{*1} = \{\tilde{a}_1, b_1, \dots, b_q\}$. It might happen that $b_i = \tilde{a}_j$ for some $i \in [q]$, $j \in [\ell + 1] \setminus \{1\}$ but this does not influence the following inequalities:

$$\begin{aligned} V_{e,k}(CS_{e,k}^*) &= v_{e,k}(C_{e,k}^{*1}) + v_{e,k}(C_{e,k}^{*2}) + \dots + v_{e,k}(C_{e,k}^{*m}) \\ &\leq r_{\tilde{a}_1}^{\max} + v_{e,k}(C_{e,k}^{*2}) + \dots + v_{e,k}(C_{e,k}^{*m}) \\ &\leq r_{\tilde{a}_1}^{\max} + v_{e,k}(\{b_1\}) + \dots + v_{e,k}(\{b_m\}) + v_{e,k}(C_{e,k}^{*2}) + \dots + v_{e,k}(C_{e,k}^{*m}) \\ &\leq r_{\tilde{a}_1}^{\max} + V_{e,k}(CS_{e,k}^{-\tilde{a}_1}), \end{aligned}$$

where $V_{e,k}(CS_{e,k}^{-\tilde{a}_1})$ denotes the optimal coalition structure on the set of agents $\mathcal{A} \setminus \{\tilde{a}_1\}$. The last inequality follows from the fact that $\{\{b_1\}, \dots, \{b_m\}, C_{e,k}^{*2}, \dots, C_{e,k}^{*m}\}$ is a coalition structure over the agents in $\mathcal{A} \setminus \{\tilde{a}_1\}$ and $CS_{e,k}^{-\tilde{a}_1}$ is an optimal such coalition structure. Now, we need to still remove agents $\tilde{a}_2, \dots, \tilde{a}_n$ to reach Phase 2 of BAAAT and to be able to compare $V_{e,k}(CS_{e,k}^*)$ with $V_{e,k}(CS_{e,k}^-)$. However, by the induction hypothesis (13) on $CS_{e,k}^{-\tilde{a}_1}$, we have

$$V_{e,k}(CS_{e,k}^{-\tilde{a}_1}) - V_{e,k}(CS_{e,k}^-) \leq \sum_{i=2}^{\ell+1} r_{\tilde{a}_i}^{\max}$$

Thus, it holds that

$$\begin{aligned} V_{e,k}(CS_{e,k}^*) &\leq r_{\tilde{a}_1}^{\max} + V_{e,k}(CS_{e,k}^{-\tilde{a}_1}) \\ &\leq r_{\tilde{a}_1}^{\max} + \sum_{i=2}^{\ell+1} r_{\tilde{a}_i}^{\max} + V_{e,k}(CS_{e,k}^-) \\ &\leq \sum_{i=1}^{\ell+1} r_{\tilde{a}_i}^{\max} + V_{e,k}(CS_{e,k}^-), \end{aligned}$$

which concludes the proof. □

Theorem 1 Let $PCSG = \langle \mathcal{A}, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. For an optimal coalition structure $CS_{e,k}^*$ and any coalition structure $CS_{e,k}^+$ obtained by BAAAT, the following holds:

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^+) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} \left(r_{\tilde{a}_i}^{\max} - v_{e,k}(\tilde{a}_i) \right). \tag{14}$$

Proof By Lemma 1, it holds that

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^-) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max}.$$

Now, by subtracting $\sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} v_{e,k}(\tilde{a}_i)$ from both sides, we get

$$\begin{aligned} & \left(V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^-) \right) - \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} v_{e,k}(\tilde{a}_i) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max} - \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} v_{e,k}(\tilde{a}_i) \\ & V_{e,k}(CS_{e,k}^*) - \left(V_{e,k}(CS_{e,k}^-) + \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} v_{e,k}(\tilde{a}_i) \right) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} \left(r_{\tilde{a}_i}^{\max} - v_{e,k}(\tilde{a}_i) \right) \\ & V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^+) \leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} \left(r_{\tilde{a}_i}^{\max} - v_{e,k}(\tilde{a}_i) \right). \end{aligned}$$

□

Let us now focus on a special case of the PCSG problem in which we can give a better bound on the quality of the solution returned by BAAAT.

Proposition 4 Let $PCSG = \langle \mathcal{A}, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. In case the expected values of all coalitions satisfy subadditivity, i.e., $v_{e,k}$ is a subadditive function, the coalition structure $CS_{e,k}^+$ obtained by BAAAT is optimal, i.e., it holds that

$$CS_{e,k}^* = CS_{e,k}^+ \tag{15}$$

Proof In the case where $v_{e,k}$ is subadditive, the optimal coalition structure $CS_{e,k}^*$ is formed by singletons. So, we just need to show that $CS_{e,k}^+$ has the form $CS_{e,k}^+ = \{\{a_i\} | a_i \in \mathcal{A}\}$. In Phase 1 of BAAAT, only singleton coalitions are formed, independently of $v_{e,k}$. In Phase 2, since $v_{e,k}$ is subadditive, the optimal coalition structure $CS_{e,k}^-$ of the relaxed problem is formed by singleton coalitions. Thus, the solution $CS_{e,k}^+$ obtained by BAAAT is formed by singleton coalitions. □

4.3 Approximation algorithm IBAAAT

Let us now try to give a better approximation algorithm for PCSG by having a more involved approach to agents that have a low attendance rate. Instead of immediately forming singleton coalitions, we will find an optimal coalition structure on the subset of such agents. We will do the same on the remaining set of agents (as in BAAAT) and the final solution will be the union of these two optima.

IBAAAT has the following two phases:

Phase 1: For a given parameter $\tilde{p} \in \mathbb{R}$ and the attendance rates of all agents, split the agents into two sets such that every agent a whose attendance rate p is such that $p \leq \tilde{p}$ is in A_1 and all other agents are in A_2 .

Phase 2: Find optimal coalition structures of the two relaxed PCSG problems on A_1 and A_2 and output their union.

Going back to our running example, let us compare the solutions returned by BAAAT and IBAAAT. Recall that the attendance types are given by $f(Ali) = 0.9$ (Type1: available), $f(Bob) = 0.5$ (Type 3: unsure), and $f(Chan) = 0.3$ (Type 4: probably not available). We still assume $k = 1$, and the expected values of coalitions can be seen in Table 3. If we, as before, choose $\tilde{p} = 0.3$, then only $Chan$ is placed in A_1 . Previously we already found that the optimal solutions of the relaxed PCSG problem on $A_2 = \{Ali, Bob\}$ are $\{Ali, Bob\}$ and $\{\{Ali\}, \{Bob\}\}$. The optimal solution on A_1 has to be $\{Chan\}$, as this is the only possible solution. Thus, the solutions obtained by IBAAAT are identical to those obtained by BAAAT, i.e., $\{\{Chan\}, \{Ali, Bob\}\}$ and $\{\{Ali\}, \{Bob\}, \{Chan\}\}$ with an expected value of 46. Now, let us set $\tilde{p} = 0.5$. In this case $A_1 = \{Bob, Chan\}$ and the solutions returned by IBAAAT and BAAAT do not necessarily coincide. Indeed, $v_{e,1}(\{Bob, Chan\}) = 34 > 28 = v_{e,1}(\{Bob\}) + v_{e,1}(\{Chan\})$. Thus, the solution returned by IBAAAT is $\{\{Ali\}, \{Bob, Chan\}\}$ and has expected value 52.

4.4 Quality guarantee of IBAAAT

Lemma 2 *Let $PCSG = \langle \mathcal{A}, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. For an optimal coalition structure $CS_{e,k}^*$ and a coalition structure $CS_{e,k}^{+-}$ on A_2 obtained by IBAAAT in Phase 2, the following inequality holds:*

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{+-}) \leq \sum_{a \in A_1} r_a^{\max}. \tag{16}$$

Proof Since $A_1 = \tilde{A}$ and $CS_{e,k}^{+-}$ returned by IBAAAT is exactly the same as $CS_{e,k}^-$ returned by BAAAT, Lemma 1 proves the claim. \square

Theorem 2 *Let $PCSG = \langle \mathcal{A}, v, f, k \rangle$ be a probabilistic coalition structure generation problem description. For an optimal coalition structure $CS_{e,k}^*$ and any coalition structure $CS_{e,k}^{++}$ obtained by IBAAAT, the following holds:*

$$V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{++}) \leq \sum_{a \in A_1} (r_a^{\max} - v_{e,k}(a)). \tag{17}$$

Proof By Lemma 2, it holds that $V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{+-}) \leq \sum_{\tilde{a}_i \in \tilde{A}} r_{\tilde{a}_i}^{\max}$. The coalition structure $CS_{e,k}^{A_1}$ obtained by IBAAAT when dealing with the relaxed problem on A_1 is in fact the optimal solution on A_1 , so $V_{e,k}(CS_{e,k}^{A_1}) \geq \sum_{a \in A_1} v_{e,k}(a)$. Then,

$$\begin{aligned}
 V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{++}) &= V_{e,k}(CS_{e,k}^*) - \left(V_{e,k}(CS_{e,k}^{A_1}) - V_{e,k}(CS_{e,k}^{+-}) \right) \\
 &= \left(V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{+-}) \right) - V_{e,k}(CS_{e,k}^{A_1}) \\
 &\leq \left(V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{+-}) \right) - \sum_{a \in A_1} v_{e,k}(a) \\
 &\leq \sum_{a \in A_1} (r_a^{\max} - v_{e,k}(a))
 \end{aligned}$$

□

4.5 Tightness of the quality guarantee

Next, we show that the bounds given by Theorems 1 and 2 are in fact tight, meaning that there is an instance for which (14) and (17) hold with equality.

Proposition 5 *The quality guarantees for BAAAT and IBAAAT given by Theorems 1 and 2, respectively, are tight.*

Proof Let us construct an instance where $\mathcal{A} = \{a_1, \dots, a_n, b_1, \dots, b_n\}$ and the characteristic function is such that all subsets of the agent set have value 0, other than $C \in \{\{a_1, b_1\}, \dots, \{a_n, b_n\}\}$ for which $v(C) = 1$. Furthermore, let the attendance of any agent $a_i, i \in [n]$ be higher than \tilde{p} and the attendance of any agent $b_i, i \in [n]$ be lower than \tilde{p} .

Then, both BAAAT and IBAAAT will return a solution of value 0 for any $k \in [n - 1]$, while the optimum has value n . Additionally, the loss of the algorithms is exactly described by the term $\sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max} = \sum_{a \in A_1} r_a^{\max}$. In summary,

$$n = n - 0 = V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^+) = V_{e,k}(CS_{e,k}^*) - V_{e,k}(CS_{e,k}^{++}) \tag{18}$$

$$\leq \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} (r_{\tilde{a}_i}^{\max} - v_{e,k}(\tilde{a}_i)) = \sum_{a \in A_1} (r_a^{\max} - v_{e,k}(a)) \tag{19}$$

$$= \sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} r_{\tilde{a}_i}^{\max} = \sum_{a \in A_1} r_a^{\max} = n \tag{20}$$

□

5 Experimental evaluation

In the experiments, BAAAT and IBAAAT are evaluated on a number of benchmarks. We note that since this is the very first work that proposes non-complete algorithms for PCSG, a comparison to existing algorithms from the literature is not possible. The only possible comparison is the one to the complete algorithm that solves the problem optimally, and this is, indeed, one of the benchmark used. In our experiments, BAAAT and IBAAAT are

implemented in Python and all experiments are carried out on a 6 core running at 3.3GHz with 32GB of RAM.

As preliminary experiments, we first investigate the influence of the value k , i.e., the number of agents who may be absent from a coalition such that the remaining agents still give a contribution to the expected coalition value, on the runtime of BAAAT. We also look into the number of singleton coalitions in the optimal coalition structure, and also what are the agent types that tend to form singleton coalitions. Then, the quality of the solutions returned by BAAAT and IBAAAT is evaluated with respect to the optimal solutions.

In order to compute optimal coalition structures, we used the CPLEX solver. The attendance type of each agent was randomly chosen from Type 1: {available (90%)} to Type 5: {not available (10%)}. For each setting, 50 problem instances were generated, each based on one of the several probability distributions for the characteristic function v that are commonly used in the literature:

- (1) Uniform: $v(C) = \text{Uniform}(0, |C|)$ [12]
- (2) Normal: $v(C) = \text{Normal}(\mu = 10 * |C|, \sigma^2 = 0.1)$ [21]
- (3) Modified uniform: $v(C) = \text{Uniform}(0, 10 * |C|) + a$, where $a = \text{Uniform}(0, 50)$ with probability 0.2 and $a = 0$ otherwise [28]
- (4) Modified normal: $v(C) = \text{Normal}(10 * |C|, 0.01) + a$ (a is as above) [20]
- (5) Beta: $v(C) = |C| * \text{Beta}(\alpha = \beta = 0.5)$ [14]
- (6) Gamma: $v(C) = |C| * \text{Gamma}(k = \theta = 2)$ [14]

5.1 Preliminary experiments

The influence of the value k is investigated. More precisely, we set $\bar{p} = 0$ and compare the runtime for finding an optimal coalition structure by varying the value k . Figure 1 represents the average runtime of all distributions for different k (i.e., $k = 0, 1, 2, 3, 4, 5$). As one can see, we observed similar results for all distributions (1)–(6). When the number of agents is small, the influence of k on the runtime is not significant. However, the difference quickly becomes larger when the number of agents increases, e.g., in the case when the number of agents is 14 and the distribution is uniform, the average runtime is 35.4 s for $k = 1$ and 809 s for $k = 5$. The experimental results thus indeed confirmed what was expected, that is, the influence of k on the runtime becomes larger when the number of agents increases. This is the case, because it is necessary to consider $\Theta(\binom{|C|}{k})$ (which is equal to $\Theta(\binom{n}{k}) = \Theta(n^k)$ in the worst case) summands in Eq. (3) in order to compute the expected value for each coalition C . For instance, in our running example of a service company dispatching interpreters with three employees, when we set $k = 2$, it is required to consider $\binom{3}{0} + \binom{3}{1} + \binom{3}{2} = 7$ cases to compute the expected value for the coalition {Ali, Bob, Chan}. That is, we consider the case where all employees are available, one of them might be absent, and two of them might be absent. However, in case the number of agents increases from 3 to 4, we need to consider $\binom{4}{0} + \binom{4}{1} + \binom{4}{2} = 11$ possible cases for computing the expected value for $k = 2$.

Next, we look into the number of singleton coalitions in the optimal coalition structure. Table 4 shows the average number of singletons in the optimal coalition structure $CS_{e,k}^*$ where $k \in [0, 5]$ for all considered distributions. We can see that the average number of singleton coalitions differs for different distributions. For the Normal case of Cautious PCSG problems, i.e., for $k = 0$, all of the agents form singleton coalitions in the optimum, while this is not the case with other considered distributions. Additionally,

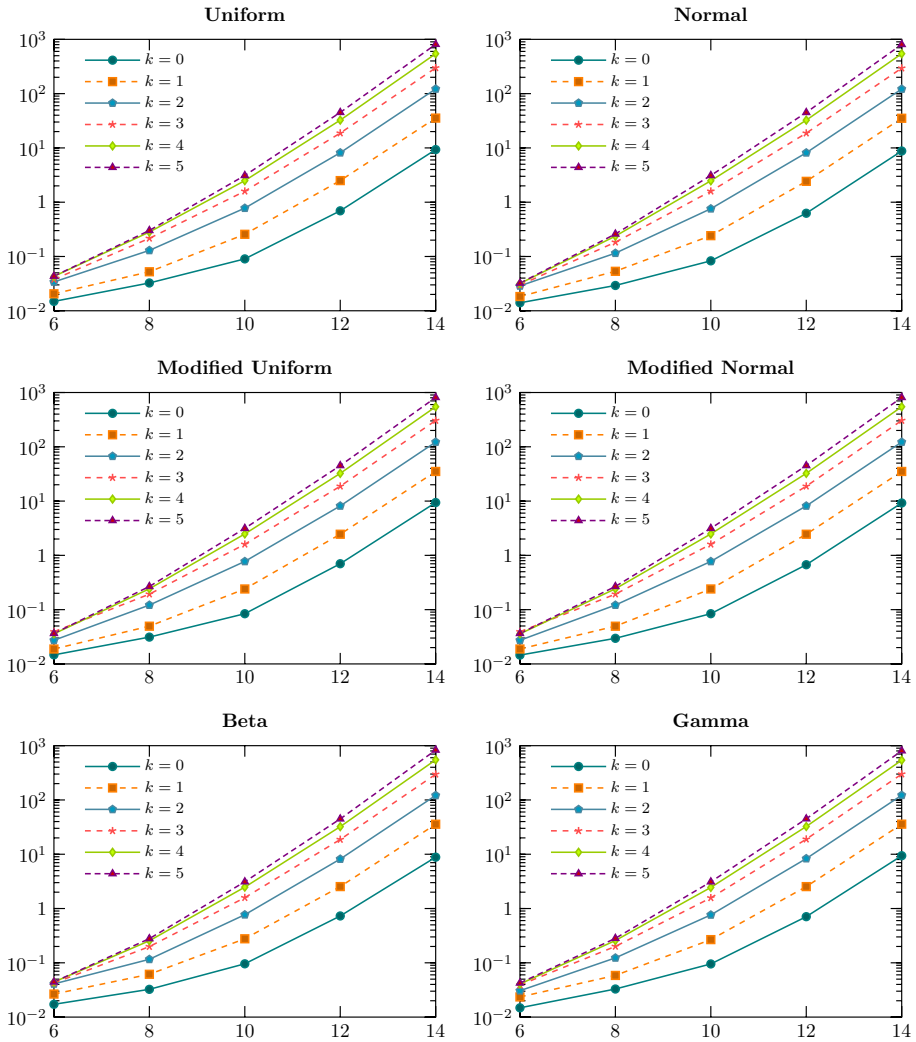


Fig. 1 Runtime of BAAAT for different k . The x-axis shows the number of agents and the y-axis represents the computation time in seconds

the number of singletons formed drastically decreases when the parameter k changes from 0 to 1. For instance, for the Beta distribution it changes from 7.6 to 1.4 when the number of agents is 14. For all distributions we can see that there are few singleton coalitions in the optimal coalition structure when $k \neq 0$.

Lastly, we analyze which agent types tend to form singleton coalitions in the optimal solution. More specifically, building upon the result of Table 4, we count the number of agents of each types that form a singleton coalition in the optimal coalition structure $CS_{e,0}^*$. Figure 2 shows the ratio of each type among the agents that formed singleton coalitions in the optimum, where the number of agents is 14 and $k = 0$. We observed that in most cases the agents with Type5, that is, the agents whose attendance rate is

Table 4 The average number of singletons in the optimal coalition structure $CS_{e,k}^*$

Uniform							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	Normal							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$													
$n=6$	4	1.4	1.1	1.2	1.2	1.2	$n=6$	6	2.2	1.9	2.1	2	2	$n=6$	6	2.2	1.9	2.1	2	2	$n=8$	8	2	2	1.9	2	2	$n=8$	8	2	2	1.9	2	2				
$n=8$	5.2	1.3	1.3	1.1	1.1	1.1	$n=10$	10	2.7	2.6	2.7	2.6	2.7	$n=10$	10	2.7	2.6	2.7	2.6	2.7	$n=12$	12	3.2	3.3	3.3	3.4	3.4	$n=12$	12	3.2	3.3	3.3	3.4	3.4				
$n=10$	6.2	1.7	1.6	1.7	1.7	1.7	$n=14$	14	3	3.5	3.7	3.7	3.7	$n=14$	14	3	3.5	3.7	3.7	3.7	$n=14$	14	3	3.5	3.7	3.7	3.7	$n=14$	14	3	3.5	3.7	3.7	3.7				
$n=12$	7.3	1.6	1.4	1.3	1.3	1.3	Mod.U							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	Mod.N							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$						
$n=14$	8.1	1.6	1.5	1.3	1.4	1.4	$n=6$	3.7	1.8	1.7	1.6	1.6	1.6	$n=6$	4.4	1.6	1.4	1.4	1.4	1.4	$n=6$	4.4	1.6	1.4	1.4	1.4	$n=6$	4.4	1.6	1.4	1.4	1.4	$n=6$	4.4	1.6	1.4	1.4	1.4
$n=6$	3.7	1.8	1.7	1.6	1.6	1.6	$n=8$	4.5	2	1.9	1.9	1.9	1.9	$n=8$	6.1	2	1.7	1.8	1.8	1.8	$n=8$	6.1	2	1.7	1.8	1.8	$n=8$	6.1	2	1.7	1.8	1.8	$n=8$	6.1	2	1.7	1.8	1.8
$n=8$	4.5	2	1.9	1.9	1.9	1.9	$n=10$	5.3	2.5	2.4	2.3	2.3	2.3	$n=10$	6.6	2.4	2.1	2	2	2	$n=10$	6.6	2.4	2.1	2	2	$n=10$	6.6	2.4	2.1	2	2	$n=10$	6.6	2.4	2.1	2	2
$n=10$	5.3	2.5	2.4	2.3	2.3	2.3	$n=12$	5.9	2.3	2.2	2.2	2.3	2.2	$n=12$	7.8	2.7	2.3	2.2	2.2	2.2	$n=12$	7.8	2.7	2.3	2.2	2.2	$n=12$	7.8	2.7	2.3	2.2	2.2	$n=12$	7.8	2.7	2.3	2.2	2.2
$n=12$	5.9	2.3	2.2	2.2	2.3	2.2	$n=14$	7.2	3.2	3.1	3	3	3	$n=14$	8.6	2.8	2.4	2.4	2.4	2.4	$n=14$	8.6	2.8	2.4	2.4	2.4	$n=14$	8.6	2.8	2.4	2.4	2.4	$n=14$	8.6	2.8	2.4	2.4	2.4
$n=14$	7.2	3.2	3.1	3	3	3	Beta							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	Gamma							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$						
$n=6$	3.7	1.2	1.1	1	1	1	$n=6$	3.7	1.5	1.5	1.5	1.5	1.5	$n=6$	3.7	1.5	1.5	1.5	1.5	1.5	$n=6$	3.7	1.5	1.5	1.5	1.5	$n=6$	3.7	1.5	1.5	1.5	1.5	$n=6$	3.7	1.5	1.5	1.5	1.5
$n=8$	4.9	1.6	1.5	1.5	1.5	1.5	$n=8$	4.9	1.6	1.5	1.5	1.5	1.5	$n=8$	4.8	1.6	1.4	1.5	1.4	1.4	$n=8$	4.8	1.6	1.4	1.5	1.4	$n=8$	4.8	1.6	1.4	1.5	1.4	$n=8$	4.8	1.6	1.4	1.5	1.4
$n=8$	4.9	1.6	1.5	1.5	1.5	1.5	$n=10$	5.3	1.4	1.4	1.3	1.3	1.3	$n=10$	5.3	1.7	1.6	1.4	1.4	1.4	$n=10$	5.3	1.7	1.6	1.4	1.4	$n=10$	5.3	1.7	1.6	1.4	1.4	$n=10$	5.3	1.7	1.6	1.4	1.4
$n=10$	5.3	1.4	1.4	1.3	1.3	1.3	$n=12$	6.7	1.4	1.4	1.4	1.4	1.4	$n=12$	6.2	2	1.8	1.8	1.9	1.9	$n=12$	6.2	2	1.8	1.8	1.9	1.9	$n=12$	6.2	2	1.8	1.8	1.9	1.9				
$n=12$	6.7	1.4	1.4	1.4	1.4	1.4	$n=14$	7.6	1.4	1.2	1.3	1.2	1.2	$n=14$	7.3	2.1	1.7	1.7	1.7	1.7	$n=14$	7.3	2.1	1.7	1.7	1.7	$n=14$	7.3	2.1	1.7	1.7	1.7	$n=14$	7.3	2.1	1.7	1.7	1.7
$n=14$	7.6	1.4	1.2	1.3	1.2	1.2	Gamma							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	Gamma							$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$						

The smallest values in each row are written in bold

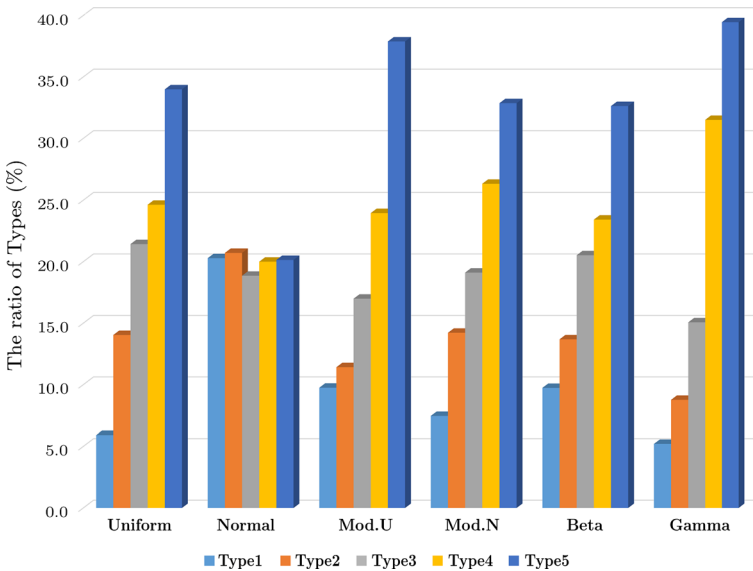


Fig. 2 The ratio of Types of agents of singleton coalitions where the number of agent is 14

lowest, tend to form singletons in the optimal coalition structure. A notable exception is the Normal case. Since the optimal coalition structure completely consists of singleton coalitions in the Normal case (see Table 4), one cannot see a difference among types.

5.2 Performance of the approximation algorithms

We set $k = 1$ and test the two algorithms, while using two different values of the parameter \tilde{p} , 0.3 and 0.7, to see the needed computation time and the quality of the returned solutions with respect to the optima. In Table 5, (a) and (b) show the actual quality of the solution obtained by BAAAT, i.e.,

$$\frac{V_{e,1}(CS_{e,1}^*)}{V_{e,1}(CS_{e,1}^+)}, \tag{21}$$

and IBAAAT, i.e.,

$$\frac{V_{e,1}(CS_{e,1}^*)}{V_{e,1}(CS_{e,1}^{++})} \tag{22}$$

in the experiments, respectively. On the other hand, (c) and (d) represent the theoretical a priori worst-case error bound on the solution quality for BAAAT, i.e.,

$$1 + \frac{\sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} (r_{\tilde{a}_i}^{\max} - v_{e,1}(\tilde{a}_i))}{V_{e,1}(CS_{e,1}^+)} \tag{23}$$

which is an upper bound on (21) by Theorem 1, and for IBAAAT, i.e.,

$$1 + \frac{\sum_{\tilde{a}_i \in \tilde{\mathcal{A}}} (r_{\tilde{a}_i}^{\max} - v_{e,1}(\tilde{a}_i))}{V_{e,1}(CS_{e,1}^{++})} \tag{24}$$

which is an upper bound on (22) by Theorem 2, respectively. All of (a), (b), (c), (d) are always at least 1.0 and a value close to 1 is desirable.

From the results for (a) and (b), we can see that when $\tilde{p} = 0.3$, BAAAT and IBAAAT have an observed approximation ratio of less than 1.05 in most cases. For instance, when the number of agents is 16, the result of (a) with the uniform distribution is 1.034, and the result of (b) is 1.019. However, when we set $\tilde{p} = 0.7$, in most cases we can see that the values obtained by BAAAT are more than 1.3, while the ones of IBAAAT are less than 1.2. For instance, when the number of agents is 12, the results of (a) and (b) with modified uniform distribution are 1.444 and 1.156, respectively. Regarding (c) and (d), we can see that the value (i.e., the a priori bound) increases gradually in all cases as the number of agents increases for both algorithms. For instance, with the uniform distribution the result of (c) for the parameter $\tilde{p} = 0.3$ increases from 2.013 to 2.806 when the number of agents increases from 6 to 16, and the result of (d) for the parameter $\tilde{p} = 0.7$ increases from 3.339 to 5.255.

Figure 3 represents the average runtime of BAAAT and IBAAAT for all distributions (1)–(6). The x -axis shows the number of agents and the y -axis represents the average runtime. As one can see, we observed similar results for all distributions.

Table 5 Solution quality of BAAAT ((a) and (c)) and IBAAAT ((b) and (d)) with respect to the optimum (a value close to 1 is desirable)

#agents		Quality	Uniform	Normal	Mod.U	Mod.N	Beta	Gamma
6	$\bar{p} = 0.3$	(a)	1.041	1.0007	1.058	1.051	1.039	1.076
		(b)	1.032	1.0005	1.048	1.045	1.031	1.064
		(c)	2.013	2.2455	2.012	2.041	1.948	2.058
		(d)	2.004	2.2453	2.003	2.035	1.941	2.046
	$\bar{p} = 0.7$	(a)	1.305	1.0029	1.340	1.206	1.360	1.456
		(b)	1.097	1.0014	1.130	1.101	1.093	1.175
		(c)	3.782	3.2347	3.483	3.213	4.075	4.065
		(d)	3.339	3.2312	3.093	3.022	3.472	3.474
8	$\bar{p} = 0.3$	(a)	1.040	1.0006	1.052	1.052	1.046	1.083
		(b)	1.027	1.0005	1.039	1.044	1.030	1.063
		(c)	2.260	2.2968	2.048	2.217	2.252	2.428
		(d)	2.244	2.2966	2.035	2.207	2.233	2.402
	$\bar{p} = 0.7$	(a)	1.243	1.0037	1.428	1.191	1.339	1.440
		(b)	1.083	1.0015	1.177	1.113	1.107	1.179
		(c)	4.155	3.6866	3.803	3.363	4.533	4.729
		(d)	3.750	3.6807	3.310	3.207	3.921	4.053
10	$\bar{p} = 0.3$	(a)	1.045	1.0007	1.042	1.042	1.041	1.047
		(b)	1.032	1.0004	1.031	1.033	1.028	1.036
		(c)	2.400	2.5803	2.415	2.230	2.482	2.482
		(d)	2.382	2.5799	2.400	2.219	2.462	2.467
	$\bar{p} = 0.7$	(a)	1.322	1.0032	1.343	1.312	1.362	1.457
		(b)	1.090	1.0013	1.125	1.135	1.115	1.169
		(c)	4.889	4.0777	4.241	3.993	5.233	5.596
		(d)	4.207	4.0718	3.714	3.588	4.465	4.689
12	$\bar{p} = 0.3$	(a)	1.037	1.0008	1.064	1.049	1.051	1.053
		(b)	1.024	1.0005	1.051	1.039	1.032	1.040
		(c)	2.717	2.5808	2.454	2.370	2.816	2.709
		(d)	2.694	2.5804	2.435	2.356	2.784	2.688
	$\bar{p} = 0.7$	(a)	1.343	1.0035	1.444	1.311	1.359	1.546
		(b)	1.106	1.0011	1.156	1.115	1.099	1.229
		(c)	5.683	4.4160	4.643	4.296	5.707	6.256
		(d)	4.857	4.4077	3.917	3.803	4.808	5.179
14	$\bar{p} = 0.3$	(a)	1.027	1.0006	1.051	1.043	1.045	1.083
		(b)	1.016	1.0004	1.037	1.031	1.028	1.059
		(c)	2.609	2.7148	2.536	2.426	2.682	3.053
		(d)	2.593	2.7144	2.516	2.410	2.655	3.008
	$\bar{p} = 0.7$	(a)	1.308	1.0035	1.364	1.315	1.370	1.652
		(b)	1.091	1.0012	1.126	1.115	1.080	1.219
		(c)	5.727	4.7264	4.666	4.504	6.117	7.328
		(d)	4.944	4.7177	4.026	3.972	5.035	5.671

Table 5 (continued)

#agents		Quality	Uniform	Normal	Mod.U	Mod.N	Beta	Gamma
16	$\tilde{p} = 0.3$	(a)	1.034	1.0005	1.053	1.043	1.051	1.054
		(b)	1.019	1.0003	1.038	1.033	1.028	1.035
		(c)	2.806	2.6974	2.461	2.421	3.103	3.158
		(d)	2.779	2.6970	2.441	2.406	3.057	3.119
	$\tilde{p} = 0.7$	(a)	1.320	1.0037	1.484	1.306	1.336	1.564
		(b)	1.077	1.0010	1.145	1.109	1.073	1.192
		(c)	6.215	4.9510	5.064	4.460	6.408	7.689
		(d)	5.255	4.9404	4.137	3.938	5.342	6.097

The a posteriori approximation bound is shown in (a) and (b), while (c) and (d) show the a priori worst-case error bound

When the parameter \tilde{p} is 0.3 and the number of agents is small, the complete algorithm, BAAAT and IBAAAT can solve the problems very quickly. However, as the number of agents increases, the difference between the complete algorithm and the approximation algorithms becomes significant. For instance, when the number of agents is 16 and the distribution is uniform, the average runtime of BAAAT and IBAAAT is 1.8 s and 2.1 s, respectively, while it is 593.3 s for the complete algorithm. The average runtime of BAAAT and IBAAAT are very similar.

However, when the parameter \tilde{p} is 0.7, we can see that the average runtime of IBAAAT increases and the one of BAAAT decreases. For example, with uniform distribution the average runtime of IBAAAT increases from 2.1 to 74.8 s when the parameter \tilde{p} increases from 0.3 to 0.7, while the average runtime of BAAAT decreases from 1.8 to 0.1 s.

In total, the experimental results are not very surprising but instead confirm and make precise the natural intuition of what kind of impact on the performance should changing some of the parameters of our model and the algorithms' parameter \tilde{p} have. In summary, they reveal that (1) when the parameter \tilde{p} is set to 0.3, the approximation ratio of BAAAT and IBAAAT is less than 1.05 in most cases, and (2) when the parameter \tilde{p} increases, BAAAT can solve the problems faster than IBAAAT, while IBAAAT has a better solution quality than BAAAT. Therefore, the user can decide how to set the parameter \tilde{p} according to his preference and if most of the agents have participation rates that are higher than \tilde{p} , both algorithms have a very similar computation time, but the solution returned by IBAAAT is always at least as good as the solution returned by BAAAT. Furthermore, if many agents have participation probabilities smaller or equal to \tilde{p} , the user chooses which algorithm to use according to whether his priority is to get an approximately optimal solution fast or he is more interested in the solution quality being as good as possible.

6 Related work

Our work is most closely related to the probabilistic CSG model introduced in [26]. Our framework includes the two variants of probabilistic CSG from [26], namely Cautious and Flexible PCSG, as special cases for values of parameter $k = 0$ and $k = n - 1$, respectively.

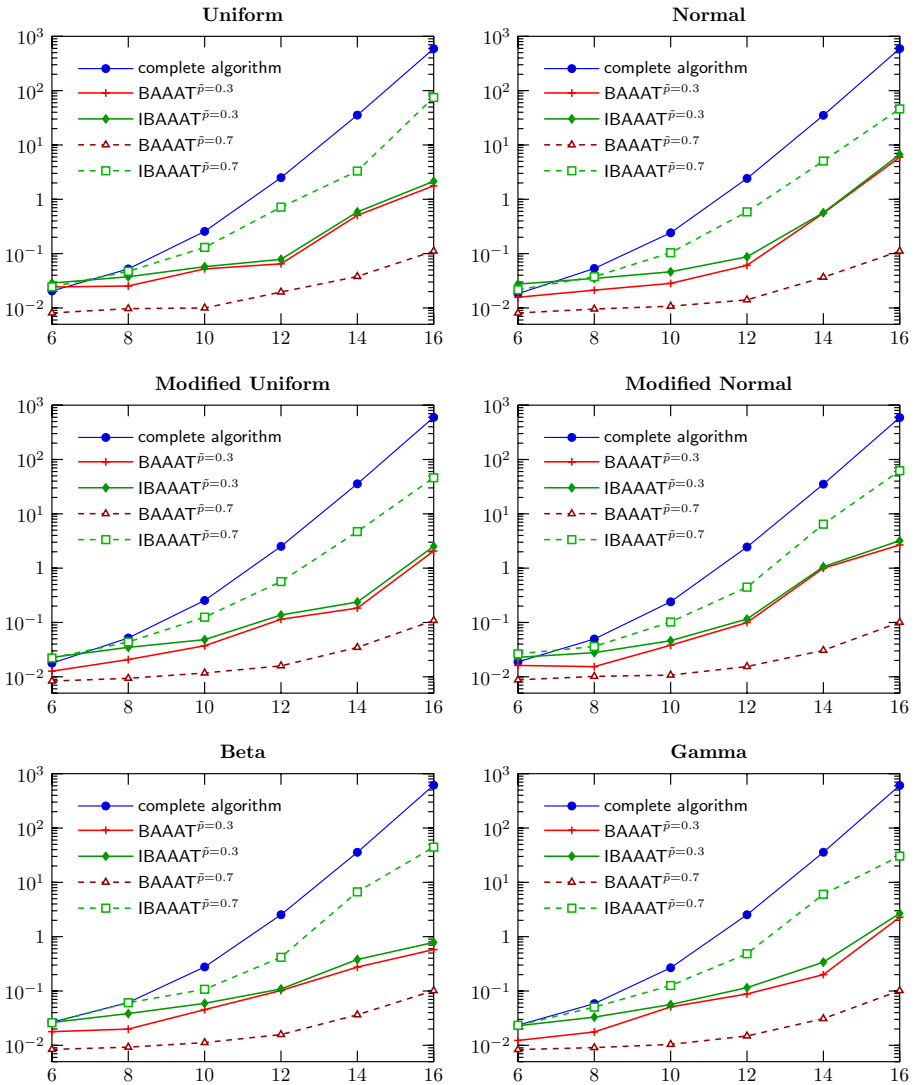


Fig. 3 Runtime of BAAAT and IBAAAT for $\bar{p} = 0.3$ and $\bar{p} = 0.7$, while $k = 1$. The x-axis shows the number of agents and the y-axis represents the computation time in seconds

For the experiments, however, we restrict our attention to a limited number of attendance types, while in [26] there is no limitation placed on the number of types.

In coalition formation, which also includes CSG, many works have been devoted to the uncertainty of forming a coalition. Chalkiadakis and Boutilier [3] focused on the uncertainty of the types (capabilities) of the agents, and proposed a Bayesian reinforcement learning framework for repeated coalition formation under type uncertainty. In this framework, the agents maintain and update beliefs about the types of others through the experience gained by repeated interaction, and through this process improve their ability to form useful coalitions. Compared to this work, we rather focus on the attendance type

(i.e., the uncertainty of agents' attendances), which is different from capability uncertainty. Also, the coalition values depend on the capabilities and the actions in the aforementioned work, while we compute them with the payoffs given by the characteristic function and the attendance rate of the attendance type.

Kraus et al. [11] worked on coalition formation under coalitional value uncertainty. In this framework, a set of tasks is given, and each task is performed by a different agent. The agents do not know the value of a task of another agent or the cost of performing it, but they know the overall payoff associated with performing a set of tasks and the capabilities of other agents. Faye et al. [8] worked on dynamic coalition formation in dynamic uncertain environments. This work investigates dynamic, uncertain environments in which tasks may evolve during execution, and agents and resource availability may vary rapidly and unpredictably. None of those works actually considers the attendance rate of each agent. Also, how the coalition values are computed in PCSG makes our model quite different from previous work.

Moreover, related to our work is the Team Formation Problem (TF) [15, 32]. TF is the problem of forming the best possible team to perform some tasks of interest, given limited resources. Nair and Tambe [15] worked on forming a team with the maximum expected value, under the constraint that it has all the required skills to accomplish the tasks of interest. To give a comparison between the TF problem and the CSG problem, it is useful to keep in mind that CSG (and PCSG) is similar to the *complete set partition problem* [33], while TF is equivalent to the *set cover problem* [10]. Okimoto et al. [16, 17] worked on the robustness issue in team formation problems. In these papers, a set of agents and a set of tasks are given, and the aim is to form a team which is robust, i.e., which can achieve the given tasks even if some agents break down. The parameter k in PCSG (i.e., the number of agents who may be absent from a coalition) has a somewhat similar role as the robustness considered in team formation.

In traditional CSG, since the value of each coalition is assumed to be given by a characteristic function, the representation size is exponential in the number of agents [24, 25]. In order to solve this problem, several compact representation schemes for characteristic functions have been proposed [4, 9, 29, 31], e.g., concise representation scheme based on agent types [31], marginal contribution nets (MC-nets) [9] and synergy coalition groups (SCGs) [4]. The attendance type used in this paper is based on the idea introduced in [31], where the authors consider a situation where multiple agents have similar capabilities/skills called *recognizable types*, and the number of possible types is small. In our work, we use this idea for the attendance rate of each agent.

7 Conclusion

How to form a coalition is a major issue for many applications related to multi-agent cooperation. Coalition Structure Generation (CSG) involves partitioning a set of agents into coalitions such that the social surplus is maximized. Probabilistic Coalition Structure Generation (PCSG) is an extension of CSG where the aim is to find an optimal coalition structure that maximizes the sum of the expected values of all coalitions. The contributions of this paper are as follows:

- A formal framework for the Probabilistic Coalition Structure Generation (PCSG) is introduced where the attendance type of each agent is considered. This framework is a

generalization of the framework introduced in [26], namely it includes the Cautious and Flexible PCSG treated in [26] as special cases.

- Approximation algorithms for solving the PCSG problem called Bounded Approximation Algorithm based on Attendance Types (BAAAT) and Involved Bounded Approximation Algorithm based on Attendance Types (IBAAAT) are presented. The characteristics of these algorithms are as follows: (1) there is an upper bound on the error of the returned solution and this bound can be obtained a priori, and (2) one can use any complete algorithm for solving to optimality the relaxed problems in Phase 2 of both algorithms.
- The performances of BAAAT and IBAAAT are evaluated on a number of benchmarks. Our experimental results revealed that (1) both algorithms can solve the PCSG problems very quickly and provide high solution quality when the parameter \tilde{p} is low, and (2) we can see a trade-off between these algorithms by varying the parameter \tilde{p} , i.e., BAAAT can solve the problems faster than IBAAAT for larger \tilde{p} , while IBAAAT has the better solution quality.

Regarding future work, a potential direction is investigating concise representations of the characteristic function in PCSG. Since the number of coalitions is exponential in the number of agents, it is reasonable to try to reduce the representation size of the characteristic function which provides the value of each coalition. One could try to apply the existing concise representations of the characteristic function from [4, 9, 29, 31] in our framework, and then try to solve large-scale problem instances.

Furthermore, what could be of interest is applying our PCSG framework to real-world problems such as the nurse scheduling problem (NSP) [2] and the distributed vehicle routing problem [25]. Forming effective working groups by considering the nurses' attendances, amounts to solving the PCSG problem. Also, drivers' attendances in the vehicle routing problem can be represented in the PCSG framework.

Lastly, our framework could potentially be extended to a dynamic setting in which the set of agents \mathcal{A} may change over time. The objective here would be to apply such an extended framework to the distributed robot team reconfiguration problem [6].

Acknowledgements This work was supported by JSPS KAKENHI Grant No. JP17H01790.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Bachrach, Y., Kohli, P., Kolmogorov, V., & Zadimoghaddam, M. (2013). Optimal coalition structure generation in cooperative graph games. In *Proceedings of 27th AAAI conference on artificial intelligence* (pp. 81–87)
2. Burke, E. K., & Curtois, T. (2014). New approaches to nurse rostering benchmark instances. *European Journal of Operational Research*, 237(1), 71–81.
3. Chalkiadakis, G., & Boutilier, C. (2012). Sequentially optimal repeated coalition formation under uncertainty. *Autonomous Agents and Multi-Agent Systems*, 24(3), 441–484.

4. Conitzer, V., & Sandholm, T. W. (2006). Complexity of constructing solutions in the core based on synergies among coalitions. *Artificial Intelligence*, *170*(6–7), 607–619.
5. Dang, V. D., Dash, R. K., Rogers, A., & Jennings, N. R. (2006). Overlapping coalition formation for efficient data fusion in multi-sensor networks. In *Proceedings of 21st national conference on artificial intelligence* (pp. 635–640)
6. Dasgupta, P., & Cheng, K. (2013). Robust multi-robot team formations using weighted voting games. *Distributed Autonomous Robotic Systems*, *83*, 373–387.
7. Dinar, A., Moretti, S., Patrone, F., & Zara, S. (2006). Application of stochastic cooperative games in water resources. In R. U. Goetz & D. Berga (Eds.), *Frontiers in water resource economics. Natural Resource Management and Policy* (pp. 1–20). Boston, MA: Springer.
8. Faye, P. F., Akinne, S., Sene, M., & Shehory, O. (2015). Dynamic coalitions formation in dynamic uncertain environments. *WI-IAT*, *2*, 273–276.
9. Jeong, S., & Shoham, Y. (2005) Marginal contribution nets: A compact representation scheme for coalitional games. In *Proceedings of 6th ACM conference on electronic commerce* (pp. 193–202)
10. Karp, R. M. (1972). Reducibility among combinatorial problems. In R. E. Miller, J. W. Thatcher & J. D. Bohlinger (Eds.), *Complexity of computer computations. The IBM Research Symposia Series* (pp. 85–103). Boston, MA: Springer.
11. Kraus, S., Shehory, O., & Taase, G. (2003). Coalition formation with uncertain heterogeneous information. In *Proceedings of 2nd international joint conference on autonomous agents and multiagent systems* (pp. 1–8)
12. Larson, K. S., & Sandholm, T. W. (2000). Anytime coalition structure generation: An average case study. *Journal of Experimental and Theoretical Artificial Intelligence*, *12*(1), 23–42.
13. Matsumura, K., Okimoto, T., & Hirayama, K. (2018) Bounded approximate algorithm for probabilistic coalition structure generation. In *Proceedings of 21st international conference on principles and practice of multi-agent systems* (pp. 123–139)
14. Michalak, T., Rahwan, T., Elkind, E., Wooldridge, M., & Jennings, N. R. (2016). A hybrid exact algorithm for complete set partitioning. *Artificial Intelligence*, *230*, 14–50.
15. Nair, R., & Tambe, M. (2005). Hybrid BDI-POMDP framework for multiagent teaming. *Journal of Artificial Intelligence Research*, *23*, 367–420.
16. Okimoto, T., Ribeiro, T., Bouchabou, D., & Inoue, K. (2016). Mission oriented robust multi-team formation and its application to robot rescue simulation. In *Proceedings of 25th international joint conference on artificial intelligence* (pp. 454–460)
17. Okimoto, T., Schwind, N., Clement, M., Ribeiro, T., Inoue, K., & Marquis, P. (2015) How to form a task-oriented robust team. In *Proceedings of 14th international conference on autonomous agents and multiagent systems* (pp. 395–403)
18. Rahwan, T., & Jennings, N. R. (2008). Coalition structure generation: Dynamic programming meets anytime optimization. In *Proceedings of 23rd AAAI conference on artificial intelligence* (pp. 156–161)
19. Rahwan, T., & Jennings, N. R. (2008) An improved dynamic programming algorithm for coalition structure generation. In *Proceedings of 7th international joint conference on autonomous agents and multiagent systems* (pp. 1417–1420)
20. Rahwan, T., Michalak, T., & Jennings, N. R. (2012). A hybrid algorithm for coalition structure generation. In *Proceedings of 26th AAAI conference on artificial intelligence* (pp. 1443–1449)
21. Rahwan, T., Ramchurn, S., Dang, V. D., Giovannucci, A., & Jennings, N. R. (2007). Anytime optimal coalition structure generation. In *Proceedings of 2nd national conference on artificial intelligence* (pp. 1184–1190)
22. Rahwan, T., Ramchurn, S. D., Jennings, N. R., & Giovannucci, A. (2009). An anytime algorithm for optimal coalition structure generation. *Journal of Artificial Intelligence Research*, *34*, 521–567.
23. Sandholm, T. W. (1993). An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of 11th national conference on artificial intelligence* (pp. 295–308)
24. Sandholm, T. W., Larson, K., Andersson, M., Shehory, O., & Tohmé, F. (1999). Coalition structure generation with worst case guarantees. *Artificial Intelligence*, *111*(1–2), 209–238.
25. Sandholm, T. W., & Lesser, V. R. (1997). Coalitions among computationally bounded agents. *Artificial Intelligence*, *94*(1–2), 99–137.
26. Schwind, N., Okimoto, T., Inoue, K., Hirayama, K., Lagniez, J. M., & Marquis, P. (2018). Probabilistic coalition structure generation. In *Proceedings of 16th international conference on principles of knowledge representation and reasoning* (pp. 663–664)
27. Service, T. C., & Adams, J. A. (2010). Anytime dynamic programming for coalition structure generation. In *Proceedings of 9th international joint conference on autonomous agents and multiagent systems* (pp. 1411–1412)

28. Service, T. C., & Adams, J. A. (2010). Approximate coalition structure generation. In *Proceedings of 24th AAAI conference on artificial intelligence* (pp. 854–859)
29. Ueda, S., Iwasaki, A., Conitzer, V., Ohta, N., Sakurai, Y., & Yokoo, M. (2018). Coalition structure generation in cooperative games with compact representations. *Autonomous Agents and Multi-Agent Systems*, 32(4), 503–533.
30. Ueda, S., Iwasaki, A., Yokoo, M., Silaghi, M. C., Hirayama, K., & Matsui, T. (2010) Coalition structure generation based on distributed constraint optimization. In *Proceedings of 24th AAAI conference on artificial intelligence* (pp. 197–203)
31. Ueda, S., Kitaki, M., Iwasaki, A., & Yokoo, M. (2011). Concise characteristic function representations in coalitional games based on agent types. In *Proceedings of 22nd international joint conference on artificial intelligence* (pp. 393–399)
32. Vidal, J. M. (2004). The effects of co-operation on multiagent search in task-oriented domains. *Journal of Experimental and Theoretical Artificial Intelligence*, 16(1), 5–18.
33. Yeh, D. Y. (1986). A dynamic programming approach to the complete set partitioning problem. *BIT Computer Science and Numerical Mathematics*, 26(4), 467–474.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Kouki Matsumura¹  · Bojana Kodric² · Tenda Okimoto³ · Katsutoshi Hirayama³

Bojana Kodric
bojana.kodric@gssi.it

Tenda Okimoto
tenda@maritime.kobe-u.ac.jp

Katsutoshi Hirayama
hirayama@maritime.kobe-u.ac.jp

¹ Graduate School of Information Science and Technology, Osaka University, Osaka, Japan

² Gran Sasso Science Institute, L'Aquila, Italy

³ Graduate School of Maritime Sciences, Kobe University, Kobe, Japan