Cooperation Preference Aware Shapley Value: Modeling, Algorithms and Applications

Hong Xie, *Member, IEEE, ACM*, and John C. S. Lui, *Fellow, IEEE, ACM*

***Abstract*— The Shapley value is a cornerstone in cooperative game theory and has been widely applied in networking, data science, etc. The classical Shapley value assumes that each player has an equal preference to cooperate with each other. Since the cooperation preference is an important factor of a variety of networking applications, we first generalize the classical Shapley value to allow general degree of the cooperation preference. In particular, we develop mathematical models to solicit two types of cooperation preferences, i.e., (1) group-wise preferences and (2) pair-wise preferences, and extend the classical Shapley value to capture this feature. Our second contribution is tackling the intrinsic computational challenge because even for the classical Shapley value, it is computationally expensive to evaluate. We design computationally efficient randomized algorithms with theoretical guarantees to fully cover the computational space of our generalized Shapley value. We also extend our models and algorithms to divide payoffs for multiple coalitions with dynamic preferences. We demonstrate the versatility of our framework by applying it to divide the revenue among ISPs in deploying new Internet architectures, as well as to divide the reward among workers in crowdsourcing systems.**

***Index Terms*— Shapley value, cooperation preferences, randomized algorithms, networking applications.**

I. Introduction

THE Shapley value [1] is a cornerstone in cooperative game theory, and it serves as an important tool in networking [2], [3], [4], data science [5], [6], [7], machine learning [8], [9], [10], etc. For example, in networking, it has been applied to allocate resources in peer-to-peer networks [3], divide revenues among Internet service providers (ISPs) [2], [11], divide rewards among workers in crowdsourcing applications [12], and quantify centralities in social networks [13], etc. In data science, it has been applied to measure the value of private data [7] and design fair and stable recommender systems [5], etc. In machine learning, it has been used to design feature selection algorithms [8], [9], design measures to interpret model predictions [14], [15] and individual classifications [10], as well as quantify the transparency of machine learning systems [16], [17], etc.

Fig. 1. Friendship graph among players.

Briefly speaking, the Shapley value is one of the solution concepts for cooperative game [18]. It aims to *"fairly"* divide the payoff among a set of players. For example, the payoff and player can be mapped as the reward and worker respectively for crowdsourcing applications. The following example illustrates the classical Shapley value.

Example 1 (The Classical Shapley Value): Consider three players denoted by $1, 2$ *and* 3 *and they work on a crowdsourcing task. Each worker only has a subset of the skills required by the task, so the task can only be completed if and only if all of them cooperate. If the task is completed, a total reward (or payoff) of* 10 *is generated, and zero otherwise. Thus, when all of them cooperate, the Shapley value distributes a payoff of* $10/3$ *to each player (refer to Sec. II-A for derivation).*

One limitation of the classical Shapley value is that each player is assumed to have an "equal" preference to cooperate with each other player, but this is often not true in real life [19], [20]. Consider the following example:

Example 2 (Limitations of the Classical Shapley Value): Suppose player $1, 2$ *and* 3 *form a line social network as shown in Fig. 1. Each player is more willing to cooperate with friends than non-friends, i.e., player* 1 *is more willing to cooperate with* 2 *than* 3*. This cooperation preference introduces a heterogeneity dividing the payoff among players. The classical Shapley value fails to capture this cooperation preference as it equally distributes the payoff to all players. In this example, player 2 should have higher reward.*

Example 2 illustrates that the classical Shapley value does not take the cooperation preference among players into account in the payoff division. The cooperation preference is an important factor for a variety of applications. For example, it can be mapped as the correlation among attributes in networking traffic classification. The equal sharing of the payoff in Example 1 is due to the *"symmetry property"* of the classical Shapley value (please refer to Section II-A for more

Manuscript received 4 August 2021; revised 23 April 2022, 29 September 2022, and 1 November 2022; accepted 15 November 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor V. Subramanian. Date of publication 19 December 2022; date of current version 19 December 2023. The work of John C. S. Lui was supported in part by the Research Grants Council's (RGC's) General Research Fund (GRF) under Grant 14200321. *(Corresponding author: Hong Xie.)*

Hong Xie was with Chongqing University, Chongqing 400044, China, and also with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong. He is now with the College of Computer Science, Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Beijing 100045, China (e-mail: xiehong2018@foxmail.com).

John C. S. Lui is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: cslui@cse.cuhk.edu.hk).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNET.2022.3228933, provided by the authors.

Digital Object Identifier 10.1109/TNET.2022.3228933

details). In particular, in Example 2, players are "symmetric" with respect to the payoff function:

$$V(\emptyset) = 0, \quad V(\{1,2,3\}) = 10,$$
$$V(\{1\}) = V(\{2\}) = V(\{3\}) = 0,$$
$$V(\{1,2\}) = V(\{2,3\}) = V(\{1,3\}) = 0,$$

where $V(\cdot)$ denotes the payoff function prescribing the payoff for each possible coalition of players.

In the cooperative game theory literature, generalizing Shapley value to capture cooperation preference can be traced back to capture personal affinities among the players in political games [19], [21]. In this research line, several variants of the Shapley value were proposed to take the cooperation preferences into the consideration in payoff division [19], [20], [21], [22], [23] (please refer to Section VI for more details). Most of previous works mainly focused on axiomatic characterization of these variants of Shapley values. Few of them studied the computational aspect of these variants of Shapley values, but with simplified cooperation preference models to enable efficient computation. In this work, we explore better tradeoffs between the complexity or expressiveness of the cooperation preference model and the computational efficiency. In particular, we aim to develop highly expressive cooperation preference models and also provide computational efficient algorithms to calculate the generalized Shapley value. In general, the cooperation preference is influenced by many factors, e.g., trust, conflict of interests, competition, etc. This paper explores such general settings and we aim to answer: *(1) How to model cooperation preferences and decouple them from the payoff function? (2) How to incorporate them into the classical Shapley value? (3) How to design efficient algorithms with theoretical guarantees to evaluate these generalized Shapley values?* These questions are challenging to answer, due to the complicate nature of players' preferences and the underlying computational complexity, i.e., the classical Shapley value is already NP-hard to evaluate in general [24]. We address these challenges and our contributions are the following:

- We develop a *rating model* to solicit *group-wise cooperation preferences*, where each player specifies a cardinal rating to represent the willingness to join a coalition of players. We also develop a *graph model* to solicit *pair-wise cooperation preferences*, where nodes represent players and each directed edge represents the willingness of a player to cooperate with the linked one. We generalize the classical Shapley value to capture these preferences, leading to the cooperation preference aware Shapley value. The exact cooperation preference aware Shapley value is computationally expensive to evaluate.
- We design efficient randomized algorithms to fully cover the computational space of our cooperation preference aware Shapley value and at the same time, provide theoretical guarantees. We first design a *Monte Carlo algorithm*, which is computationally efficient in approximating the Shapley value under the group-wise cooperation preference, but it is computationally expensive for the pair-wise cooperation preference. We extend the Monte Carlo algorithm to allow approximate samples, resulting in the ϵ-*Oracle based algorithm*. The ϵ-*Oracle based algorithm* approximates a sub-class (i.e., the preference sensitivity parameter is small) of the Shapley value under the pair-wise preference. Then we design an *importance sampling based algorithm* to approximate another sub-class (allow a larger preference sensitivity parameter but with extra conditions on payoff functions) of the pair-wise preference aware Shapley value. Finally, we design a *random walk based algorithm* to approximate the remaining sub-class of the pair-wise cooperation preference aware Shapley value.
- We extend our framework to allow multiple coalitions, where we capture the externalities among coalitions (e.g., competition among coalitions). We also extend our model to allow dynamic cooperation preferences. For each extension, we not only extend our framework to divide the payoff, but also extend our proposed algorithms to compute the corresponding Shapley value.
- Finally, we demonstrate the versatility of our framework by applying it to divide the revenue among ISPs in deploying new Internet architectures, as well as divide the reward among workers in crowdsourcing systems. We provide important insights on the impact of cooperation preference.

This paper organizes as follows. Section II presents cooperation preference models their associated Shapley values. Section III presents several sampling algorithms to approximate the cooperation preference aware Shapley value. Section IV presents two extensions of the model. Section V presents networking applications of our work. Section VI presents the related work. Section VII concludes.

II. The Model & Problem Formulation

We first present the classical Shapley value. We then develop two mathematical models to capture two types of cooperation preferences. We show that the exact cooperation preference aware Shapley value is computationally expensive to compute. *This section focuses on that all players forms a grand coalition and we extend it to multiple coalitions in Section IV.*

A. The Classical Shapley Value

Consider a coalition game with a set of $N \in \mathbb{N}_+$ players (or users) denoted by $\mathcal{N} \triangleq \{1, \ldots, N\}$. Denote the payoff (or gain) function as $V : 2^{\mathcal{N}} \to \mathbb{R}$. The $V(\mathcal{C})$, where $\mathcal{C} \subseteq \mathcal{N}$, describes the total gains to the set $\mathcal{C}$ of players under cooperation. We set $V(\emptyset) = 0$ by default. Consider all players form a grand coalition, let $\phi_n(V)$ denote the gain distributed to player $n \in \mathcal{N}$. A canonical form of the classical Shapley value [1] is

$$\phi_n(V) = \sum_{\mathcal{C} \subseteq \mathcal{N} \setminus \{n\}} \frac{|\mathcal{C}|!(|\mathcal{N}| - |\mathcal{C}| - 1)!}{|\mathcal{N}|!} [V(\mathcal{C} \cup \{n\}) - V(\mathcal{C})]. \tag{1}$$

Consider Example 1, we have $\mathcal{N} = \{1, 2, 3\}$ and

$$V(\mathcal{C}) = \begin{cases} 10, & \text{if } \mathcal{C} = \mathcal{N}, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Equation (1) implies $\phi_1(V) = \phi_2(V) = \phi_3(V) = 10/3$. The classical Shapley value has the following four properties.

1) **Efficiency**: all payoffs are distributed to players, i.e., $\sum_{n\in\mathcal{N}} \phi_n(V) = V(\mathcal{N})$.
2) **Zero player**: if a player has no contribution to any coalition, no gain will be distributed to it, i.e.,
$$V(\mathcal{C}\cup\{n\}) - V(\mathcal{C}) = 0, \forall \mathcal{C}\subseteq \mathcal{N}\setminus\{n\} \Rightarrow \phi_n(V) = 0.$$
3) **Linearity**: if we combine two coalition games with the same set of players $\mathcal{N}$ and with payoff functions V and $\widetilde{V}$ respectively, then the gains distributed to player n equals to the sum of the gain distributed from V and V', i.e., $\phi_n(V+V') = \phi_n(V) + \phi_n(V')$.
4) **Symmetry**: if two players have equal contribution in each coalition, they receive the same gain, i.e.,
$$\begin{aligned} V(\mathcal{C}\cup\{n\}) &= V(\mathcal{C}\cup\{m\}), \forall \mathcal{C}\subseteq\mathcal{N}\setminus\{m,n\} \\ &\Rightarrow \phi_n(V) = \phi_m(V). \end{aligned}$$

One limitation of the classical Shapley value is that it does not capture cooperation preferences among players. In particular, the symmetry property implies that two players have equal shares of payoff if they have the same influence on the payoff function, no matter what their impact on the cooperation preference is. One may argue to modify the payoff function $V(\cdot)$ to encode the cooperation preference. However, as shown in Example 2, the payoff function captures the physical meaning that each worker only has a subset of the skills required by the task, and the task can be completed if and only if all of them cooperate. If we modify the payoff function to encode the cooperation preference, the payoff function may lose this important property. Furthermore, it is more natural to decouple cooperation preference from the payoff function $V(\cdot)$. Thus, we aim to capture the cooperation preference decoupling from the payoff function. We will retain the attractive properties of Shapley Value, i.e., property (1), (2) and (3), but remove the constraint on "equal preference" by the classical Shapley value, i.e., property (4).

B. Cooperation Preference Aware Shapley Value

We present a probabilistic representation of the classical Shapley value [1]. We develop two mathematical models to capture two types of cooperation preferences, and generalize the probabilistic representation to derive closed-form formula for the payoff division under these preferences.

Probabilistic representation [1]. Shapley proposed a bargaining model of coalition formation and showed that the classical Shapley value is an expected outcome of this bargaining model [1]. In the proposed bargaining model, each ordering of players is thought of as successive arrivals of players. Formally, denote an ordering of the players as $\boldsymbol{\sigma}\triangleq(\sigma_1,\ldots,\sigma_N)$, where $\sigma_n\in\mathcal{N}$ denotes the player in the n-th order. For example, if $\mathcal{N}=\{1,2,3\}$, an ordering of the players can be $\boldsymbol{\sigma}=(2,3,1)$, i.e., player 2 is ranked first, while player 3 is ranked second. Denote the set of players ranked before player n in the ordering $\boldsymbol{\sigma}$ as

$$\mathcal{S}_n^{\boldsymbol{\sigma}} \triangleq \{\text{all players ranked before player } n \text{ in } \boldsymbol{\sigma}\}.$$

For example, consider an ordering $\boldsymbol{\sigma}=(2,3,1)$, then $\mathcal{S}_1^{\boldsymbol{\sigma}} = \{2,3\}$ and $\mathcal{S}_2^{\boldsymbol{\sigma}} = \emptyset$. One important interpretation of $\mathcal{S}_n^{\boldsymbol{\sigma}}$ is that the user n and the set of users precede him form a coalition $\mathcal{C}=\mathcal{S}_n^{\boldsymbol{\sigma}}\cup\{n\}$ [1]. Based on this interpretation, the classical Shapley value can be rewritten as

$$\phi_n(V) = \sum_{\boldsymbol{\sigma}\in\Omega} \frac{1}{N!}\left[V(\mathcal{S}_n^{\boldsymbol{\sigma}}\cup\{n\}) - V(\mathcal{S}_n^{\boldsymbol{\sigma}})\right],$$

where $\Omega \triangleq \{\boldsymbol{\sigma}|\boldsymbol{\sigma} \text{ is an ordering of players in } \mathcal{N}\}$ denotes a set of all orderings of players. Note that there are in total $|\Omega| = N!$ number of orderings. Hence, under the classical Shapley value, $1/N!$ can be interpreted as the probability mass of an ordering and different orderings have equal probabilities. This leads to a generalization of the classical Shapley value

$$\phi_n(V) = \sum_{\boldsymbol{\sigma}\in\Omega} \mu(\boldsymbol{\sigma})\left[V(\mathcal{S}_n^{\boldsymbol{\sigma}}\cup\{n\}) - V(\mathcal{S}_n^{\boldsymbol{\sigma}})\right], \tag{3}$$

where μ denotes a general probability distribution over Ω, i.e., $\mu(\boldsymbol{\sigma})\geq 0$ and $\sum_{\boldsymbol{\sigma}\in\Omega}\mu(\boldsymbol{\sigma})=1$. One can easily verify the following proposition.

Proposition 1 [1]: For any probability distribution μ over Ω, $\phi_n(V)$ expressed in Equation (3) satisfies property (1), (2) and (3), i.e., efficiency, zero player and linearity.

Note that $\mu(\boldsymbol{\sigma}) = 1/N!$ corresponds to the classical Shapley value. The uniform distribution captures a *homogeneous cooperation preference*, i.e., each player has the same preference to cooperate with other players. We next consider non-uniform distribution μ to capture heterogeneous cooperation preferences, i.e., a player may prefer to cooperate with some players over others. Under non-uniform distribution μ, only the symmetry property of the classical Shapley value is dropped, as indicated by Proposition 1.

• **Preference Model 1: group-wise cooperation preferences.** The group-wise cooperation preference model generalizes the bargaining model of coalition formation [1]. Essentially, the bargaining model of coalition formation offers an interpretation of how the grand coalition containing all players is formed via successive arrival of players. The cooperation preference is captured in the arrival order of players as we proceed to model. Given coalition $\mathcal{C}$, we use a cardinal rating $R_n(\mathcal{C}) \in [1,M]$ to model the preference (such as trust, conflict of interests, etc.) of player $n\notin\mathcal{C}$ to join the coalition $\mathcal{C}$, where $M\in\mathbb{N}_+$. A larger rating $R_n(\mathcal{C})$ models a higher preference. Denote the cooperation preference profile of player n as $\mathcal{R}_n \triangleq \{R_n(\mathcal{C})|\forall\mathcal{C}\subseteq\mathcal{N}\setminus\{n\}\}$. Consider $\mathcal{N}=\{1,2,3\}$, the preference profile of player 1 can be $\mathcal{R}_1 = \{R_1(\emptyset), R_1(\{2\}), R_1(\{3\}), R_1(\{2,3\})\}$. Denote the preference profile of all players as $\mathscr{R}\triangleq\{\mathcal{R}_n|\forall n\in\mathcal{N}\}$. We use a finite horizon discrete time $t\in\{0,1,\ldots,N\}$ stochastic process to capture the cooperation preference $\mathscr{R}$. Let $\mathcal{C}_t$ denote the coalition at time slot t. Initially, at time slot $t=0$, no one is in the coalition, i.e., $\mathcal{C}_0=\emptyset$. In time slot $t+1$, one player in $\mathcal{N}\setminus\mathcal{C}_t$ joins the coalition. We use the joining probability to capture the cooperation preference. Formally, denote the conditional probability that user $n\in\mathcal{N}\setminus\mathcal{C}_t$ joins $\mathcal{C}_t$ as:

$$p(n|\mathcal{C}_t) = \mathbb{P}[\mathcal{C}_{t+1}=\mathcal{C}_t\cup\{n\}|\mathcal{C}_t;\mathscr{R}], \quad \forall n\in\mathcal{N}\setminus\mathcal{C}_t,$$

where $\sum_{n\in\mathcal{N}\setminus\mathcal{C}_t} p(n|\mathcal{C}_t) = 1$. The following assumption captures that in the coalition formation process, the player with a higher preference rating would more likely to join the coalition.

Assumption 1: The probability $p(n|\mathcal{C}_t)$ is a function of the preference ratings $R_m(\mathcal{C}_t), \forall m \in \mathcal{N}\setminus\mathcal{C}_t$, formally

$$p(n|\mathcal{C}_t) = F(n; R_m(\mathcal{C}_t), \forall m \in \mathcal{N}\setminus\mathcal{C}_t), \quad \forall n \in \mathcal{N}\setminus\mathcal{C}_t, \quad (4)$$

where F denotes a function with range $[0,1]$. Furthermore, the function $F(n; R_m(\mathcal{C}_t), \forall m \in \mathcal{N}\setminus\mathcal{C}_t)$ is increasing in $R_n(\mathcal{C}_t)$ and pairwise-monotone

$$F(n; R_m(\mathcal{C}_t), \forall m \in \mathcal{N}\setminus\mathcal{C}_t) \geq F(n'; R_m(\mathcal{C}_t), \forall m \in \mathcal{N}\setminus\mathcal{C}_t)$$

whenever $R_n(\mathcal{C}_t) \geq R_{n'}(\mathcal{C}_t)$, where $n, n' \in \mathcal{N}\setminus\mathcal{C}_t$.

One possible example of the conditional probability is

$$p(n|\mathcal{C}_t) = \frac{R_n(\mathcal{C}_t)}{\sum_{m\in\mathcal{N}\setminus\mathcal{C}_t} R_m(\mathcal{C}_t)}, \quad \forall n \in \mathcal{N}\setminus\mathcal{C}_t. \quad (5)$$

The probability distribution μ can be derived as:

$$\mu(\boldsymbol{\sigma}) = \prod_{t=1}^{N} p(\sigma_t|\mathcal{C}_{t-1} = \{\sigma_1, \ldots, \sigma_{t-1}\}). \quad (6)$$

For example, consider $\boldsymbol{\sigma} = (2,3,1)$, we have $\mu(2,3,1) = p(2|\emptyset)p(3|\{2\})p(1|\{2,3\})$. It is important to note that the classical Shapley value is a special case of this preference model with $p(n|\mathcal{C}_t) = 1/(N-t), \forall t = 0,1,\ldots,N-1$. Plugging Equation (6) into Equation (3) we obtain the Shapley under the group-wise cooperation preference. Proposition 1 implies that the Shapley value under the group-wise cooperation preference satisfies the attractive properties (1), (2) and (3).

Note that the cooperation preference profile $\mathcal{R}_n$ of user n has 2^{N-1} possibilities. In total, the preference profile $\mathscr{R}$ contains a total of $N2^{N-1}$ elements. This rises a challenge to store the preference profile $\mathscr{R}$, as well as to solicit the preference profile, because each user may not be willing to specify all 2^{N-1} preferences, especially when N is large. We propose the following two methods to address it:

- *Partial preference.* Players specify preferences to a number of coalitions that are of their greatest concern, i.e., $\{R_n(\mathcal{C})|\mathcal{C} \in \mathscr{C}_n$, where $\mathscr{C}_n \subseteq 2^{\mathcal{N}\setminus\{n\}}\}$, while setting a default preference rating for the remaining coalitions, i.e., $R_n(\mathcal{C}) = c, \forall \mathcal{C} \in 2^{\mathcal{N}\setminus\{n\}} \setminus \mathscr{C}_n$, where $c \in [1, M]$. This partial preference model is suitable for the case that each player only differentiates a small number of coalitions and is indifferent to the remaining coalitions. Otherwise, this partial preference model leads to underfitting of cooperation preferences.
- *Low rank representation.* Each player is represented by a vector $\boldsymbol{x}_n$. Let $\boldsymbol{x}_{\mathcal{C}} \triangleq \{\boldsymbol{x}_n : n \in \mathcal{C}\}$ denote the vector profile associated with the coalition $\mathcal{C}$. Then the preference score $R_n(\mathcal{C})$ is modeled as a function of $\boldsymbol{x}$ and $\boldsymbol{x}_{\mathcal{C}}$. For example,

$$R_n(\mathcal{C}) = 1 + \frac{M-1}{1+\exp(\sum_{m\in\mathcal{C}} \|\boldsymbol{x}_n - \boldsymbol{x}_m\|)}.$$

  In this case, we only need to solicit and store $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$. The vector $\boldsymbol{x}_n$ represents the features of players and it is a public information. The distance $\|\boldsymbol{x}_n - \boldsymbol{x}_m\|$ reflects the affinity between two players.

- **Preference Model 2: pair-wise cooperation preferences.** Another alternative is to use a weighted and directed graph $\mathcal{G}$ to model pair-wise cooperation preferences between users. Each player is represented as a node, i.e., the node set is $\mathcal{N}$. Each directed edge from player m to player n is associated with a weight $w_{mn} \in [0,1]$. The weight quantifies the preference of user m to cooperate with user n. A larger weight models a higher preference. We set $w_{nn} = 0$ by default to capture that there is no self-loop in the graph. Denote the pair-wise preference profile as $\boldsymbol{W} \triangleq [w_{mn}|m,n \in \mathcal{N}]$. The graph $\mathcal{G}$ can be represented as $\mathcal{G} = (\mathcal{N}, \boldsymbol{W})$. In practice, there are numerous ways to solicit the weight w_{mn}. For example, it can be directly extracted from an online social network, where $w_{mn} = 1$ indicates a friendship link and $w_{mn} = 0$ indicates no friendship links.

Given a coalition $\mathcal{C}$, denote the pair-wise preference profile restricted to $\mathcal{C}$ as $\boldsymbol{W}_{\mathcal{C}} \triangleq [w_{mn}|m,n \in \mathcal{C}]$. We quantify the aggregate coalition preference of coalition $\mathcal{C}$ as $\Gamma(\mathbf{W}_{\mathcal{C}}) \in \mathbb{R}_{\geq 0}$, where Γ is increasing in $w_{mn}, \forall m,n \in \mathcal{C}$, capturing that increasing the pair-wise preference between any two players increases the aggregate coalition preference. One possible example of $\Gamma(\mathbf{W}_{\mathcal{C}})$ is

$$\Gamma(\mathbf{W}_{\mathcal{C}}) = \exp\left(\sum_{m,n\in\mathcal{C}} w_{mn}\right). \quad (7)$$

The larger the aggregate coalition preference $\Gamma(\mathbf{W}_{\mathcal{C}})$, the more likely the coalition $\mathcal{C}$ will be formed. Note that Equation (7) is just an illustrating example, and our proposed algorithms are not restricted to it.

Recall from the probabilistic representation of the Shapley value that the ordering $\boldsymbol{\sigma}$ induces the following coalitions $\{\sigma_1\}, \{\sigma_1,\sigma_2\}, \ldots, \{\sigma_1,\ldots,\sigma_N\}$. Each coalition $\{\sigma_1,\ldots,\sigma_n\}, \forall n = 1,\ldots,N$, is associated with an aggregate coalition preference $\Gamma(\boldsymbol{W}_{\{\sigma_1,\ldots,\sigma_n\}})$. Let $\Upsilon(\boldsymbol{\sigma}) \in \mathbb{R}_{\geq 0}$ denote the collective cooperation preference associated with the ordering $\boldsymbol{\sigma}$. We model $\Upsilon(\boldsymbol{\sigma})$ as a function of $\Gamma(\boldsymbol{W}_{\{\sigma_1,\ldots,\sigma_n\}}), \forall n = 1,\ldots,N$. One possible example is

$$\Upsilon(\boldsymbol{\sigma}) = \sum_{n=1}^{N} \Gamma(\boldsymbol{W}_{\{\sigma_1,\ldots,\sigma_n\}}). \quad (8)$$

An ordering $\boldsymbol{\sigma}$ is more likely to be formed if it has a larger $\Upsilon(\boldsymbol{\sigma})$. Note that Equation (8) is just an illustrating example, and our proposed algorithms are not restricted to it. Formally, we use the probability distribution μ to quantify the impact of cooperation preference on the likelihood of orderings as

$$\mu(\boldsymbol{\sigma}) = \frac{\exp(\alpha\Upsilon(\boldsymbol{\sigma}))}{\sum_{\widetilde{\boldsymbol{\sigma}}\in\Omega} \exp(\alpha\Upsilon(\widetilde{\boldsymbol{\sigma}}))}, \quad (9)$$

where $\alpha \in \mathbb{R}_{\geq 0}$ models the sensitivity to cooperation preferences. The probability distribution $\mu(\boldsymbol{\sigma})$ is more sensitive to cooperation preferences when we increase the value of α. It is also important to note that the classical Shapley value is a special case of $\alpha = 0$. Equation (9) aims to captures that $\mu(\boldsymbol{\sigma})$ increases when the collective cooperation

preference $\Upsilon(\boldsymbol{\sigma})$ specified by $\boldsymbol{\sigma}$ increases. Equation (9) is the well known softmax function, which is extensively used in many previous works to model probability distributions [25]. Our work presents a simple way to model sensitivity to cooperation preferences and it offers flexibility to allow $\Upsilon(\boldsymbol{\sigma})$ to take negative values. The exponential function is not a restriction. In particular, consider the general form of distribution $\mu(\boldsymbol{\sigma}) = cf(\boldsymbol{\sigma})$, where $f(\boldsymbol{\sigma}) > 0$ and $c = 1/\sum_{\boldsymbol{\sigma}\in\Omega} f(\boldsymbol{\sigma})$ is the normalizing factor. This general form of distribution can be rewritten into an exponential form as $\mu(\boldsymbol{\sigma}) = \exp(g(\boldsymbol{\sigma}))/\sum_{\tilde{\boldsymbol{\sigma}}\in\Omega}\exp(g(\tilde{\boldsymbol{\sigma}}))$, where $g(\boldsymbol{\sigma}) = \ln f(\boldsymbol{\sigma})$. Proposition 1 implies that the Shapley value under the pair-wise cooperation preference satisfies attractive properties (1), (2) and (3).

C. Problem Formulation

Note that computing the exact classical Shapley value is NP-hard in general [24]. Computing the exact cooperation preference aware Shapley value would be more challenging, as the classical Shapley value is simply a special case of it when $\alpha = 0$. Henceforth, we aim to design computational efficient algorithms to approximate the cooperation preference aware Shapley value which have theoretical guarantees. Formally, we consider the following problem.

Problem 1: Given the player set $\mathcal{N}$, payoff function V, cooperation preferences $\mathscr{R}$ or $\mathcal{G}$, conditional probability $p(n|\mathcal{C}_t)$ or preference function Υ, and sensitivity parameter α. Design algorithms which have theoretical guarantees to compute the cooperation preference aware Shapley value.

III. Sampling Algorithms

We first design a *Monte Carlo algorithm*, which is computationally efficient in approximating the Shapley value under the group-wise preference model, but it is computationally expensive for the pair-wise preference. We extend the Monte Carlo algorithm to allow approximate samples, resulting in the *ϵ-Oracle based algorithm*. The *ϵ-Oracle based algorithm* approximates a sub-class (i.e., the preference sensitivity parameter is small) of the Shapley value under pair-wise preferences. Then, we design an *importance sampling based algorithm* to approximate another sub-class (allow larger preference sensitivity parameter but with extra conditions on the payoff function) of the pair-wise preference aware Shapley value. Finally, we design a *random walk based algorithm* to approximate the remaining sub-class of the pair-wise preference aware Shapley value.

A. A Monte Carlo Algorithm

Algorithmic framework based on Monte Carlo oracle. Note that the cooperation preference aware Shapley expressed in Equation (3) can be rewritten as

$$\phi_n(V) = \mathbb{E}_{\boldsymbol{\sigma}\sim\mu}\left[\Delta(\boldsymbol{\sigma}, n)\right], \tag{10}$$

where $\Delta(\boldsymbol{\sigma}, n)$ is defined as

$$\Delta(\boldsymbol{\sigma}, n) \triangleq V(\mathcal{S}_n^{\boldsymbol{\sigma}} \cup \{n\}) - V(\mathcal{S}_n^{\boldsymbol{\sigma}}).$$

The $\Delta(\boldsymbol{\sigma}, n)$ quantifies the marginal gain of player n joining a coalition induced by the ordering $\boldsymbol{\sigma}$. Equation (10) shows that the cooperation preference aware Shapley value is the expectation of $\Delta(\boldsymbol{\sigma}, n)$ over the probability measure μ. Based on Equation (10), Algorithm 1 outlines a Monte Carlo algorithm to approximate the cooperation preference aware Shapley value. To run algorithm 1, one needs to input an $\texttt{MC-Oracle}(\mu)$ of the probability distribution μ and the sample size K. It generates K orderings from $\texttt{MC-Oracle}(\mu)$. Then, we can use each ordering to obtain one sample of the Shapley value for user n. Finally, it uses the average of all these samples to estimate $\phi_n(V)$. Note that Step 4 of Algorithm 1 is an incremental update version of the simple average.

Algorithm 1 An $\texttt{MC-Oracle}$ Based Algorithm

Require: An $\texttt{MC-Oracle}(\mu)$ for generating IID samples from the probability distribution μ, sample size K.
Ensure: $\widehat{\phi}_n(V)$
1: $\widehat{\phi}_n(V) \leftarrow 0$;
2: **for** $k = 1$ to K **do**
3: Generate a ordering $\boldsymbol{\sigma}$ from the probability distribution μ:

$$\boldsymbol{\sigma} \leftarrow \texttt{MC-Oracle}(\mu);$$

4: $\widehat{\phi}_n(V) \leftarrow \left(1 - \frac{1}{k}\right)\widehat{\phi}_n(V) + \frac{1}{k}\Delta(\boldsymbol{\sigma}, n)$;
5: **end for**

To illustrate, consider the setting of Example 1. Suppose the ordering generated in the first round, i.e., $k = 1$, is $\boldsymbol{\sigma} = (1, 2, 3)$. It follows that $\Delta(\boldsymbol{\sigma}, 1) = V(\{1\}) - V(\emptyset) = 0 - 0 = 0$ and correspondingly $\widehat{\phi}_1(V)$ is updated as $\widehat{\phi}_1(V) \leftarrow (1 - 1/1) \times 0 + 1 \times 0 = 0$. Suppose the ordering generated in the second round, i.e., $k = 2$, is $\boldsymbol{\sigma} = (3, 2, 1)$. It follows that $\Delta(\boldsymbol{\sigma}, 1) = V(\{3, 2, 1\}) - V(\{3, 2\}) = 10$ and correspondingly $\widehat{\phi}_1(V)$ is updated as $\widehat{\phi}_1(V) \leftarrow (1 - \frac{1}{2}) \times 0 + \frac{1}{2} \times 10 = 5$. Algorithm 1 has the following property.

Theorem 1: The output $\widehat{\phi}_n(V)$ of Algorithm 1 satisfies

$$\left|\widehat{\phi}_n(V) - \phi_n(V)\right| \leq \frac{\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)}{\sqrt{K}}\sqrt{\frac{1}{2}\ln\frac{2}{\delta}},$$

with a probability of at least $1 - \delta$, where $\delta \in (0, 1]$.

Theorem 1 states an upper bound of the approximation error of Algorithm 1. The upper bound of the approximation error is linear in $[\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)]/\sqrt{K}$. This implies that given an estimation error, the number of samples K needed is proportional to $[\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)]^2$. In other words, the larger the gap $|\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)|$ of the marginal gain, the more samples are needed to be generated.

The computational complexity of Algorithm 1 is

$$O(K \times \textit{complexity of generating a sample from } \texttt{MC-Oracle}(\mu)).$$

We next investigate how to design a computationally efficient MC-Oracle(μ).

Implementing the MC-Oracle(μ). Algorithm 2 implements the Monte Carlo oracle MC-Oracle(μ) for the group-wise cooperation preference model with a computational complexity of $O(N)$. However, it is computationally expensive to use the Monte Carlo oracle MC-Oracle(μ) for the pair-wise cooperation preference model. The reason is that the distribution μ (expressed in Equation (9)) under the pair-wise preference has a partition function $\sum_{\widetilde{\boldsymbol{\sigma}}\in\Omega}\exp(\alpha\Upsilon(\widetilde{\boldsymbol{\sigma}}))$, which is computationally expensive to evaluate in general, as the cardinality of the sample space Ω is $|\Omega| = N!$.

Algorithm 2 MC-Oracle(μ) for Group-Wise Preference

Require: Group-wise preference profile $\mathscr{R}$ and function F.
Ensure: $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)$
1: $\mathcal{C}_0 \leftarrow \emptyset$;
2: **for** $t = 1$ to N **do**
3: Evaluate $p(\cdot|\mathcal{C}_{t-1})$ according to Eq. (4) with F and $\mathscr{R}$;
4: Generate a player from the distribution $p(\cdot|\mathcal{C}_{t-1})$:

$$\sigma_t \sim p(\cdot|\mathcal{C}_{t-1});$$

5: Add the generated player to the coalition:

$$\mathcal{C}_t \leftarrow \mathcal{C}_{t-1} \cup \{\sigma_t\};$$

6: **end for**

B. An ϵ-Oracle Based Algorithm

Now, we consider the pair-wise cooperation preference in Section II.

Algorithmic framework based on ϵ-Oracle. In order to precisely define an ϵ-Oracle, we first present a metric to quantify the distance between two probability distributions.

Definition 1: The total variation (TV) distance between two probability measures μ and ν on $(\Omega, 2^{\Omega})$ is defined by

$$\| \mu - \nu \|_{TV} \triangleq \max_{\mathcal{A}\subseteq\Omega} |\mu(A) - \nu(A)|.$$

For example, if two distributions μ and ν are identical, i.e., $\mu = \nu$, then the total variation distance is $\| \mu - \nu \|_{TV} = 0$. Base on definition 1, we define an ϵ-Oracle to generate approximate IID samples from the distribution μ.

Definition 2: An ϵ-Oracle of the probability distribution μ generates IID samples from a probability distribution ν satisfying that $\| \mu - \nu \|_{TV} \leq \epsilon$, where $\epsilon \in \mathbb{R}_{\geq 0}$.

Given an ϵ-Oracle of the probability distribution μ, we present Algorithm 3, which outlines a framework to approximate the Shapley value under the pair-wise cooperation preference. To run Algorithm 3, one needs to input an ϵ-Oracle(μ) of the probability distribution μ and the sample size K. Algorithm 3 first generates K orderings from ϵ-Oracle(μ), then uses each ordering to obtain one sample of the Shapley value for user n. Finally, it averages all these samples to estimate $\phi_n(V)$.

To quantify the accuracy of Algorithm 3, we first decompose the error as $|\widehat{\phi}_n(V) - \phi_n(V)| \leq \epsilon_{app} + \epsilon_{est}$, where ϵ_{app} and ϵ_{est} are defined as the approximation error and estimation error respectively:

$$\epsilon_{app} \triangleq \left| \sum\nolimits_{\boldsymbol{\sigma}\in\Omega} \nu(\boldsymbol{\sigma})\Delta(\boldsymbol{\sigma}, n) - \phi_n(V) \right|,$$
$$\epsilon_{est} \triangleq \left| \widehat{\phi}_n(V) - \sum\nolimits_{\boldsymbol{\sigma}\in\Omega} \nu(\boldsymbol{\sigma})\Delta(\boldsymbol{\sigma}, n) \right|.$$

Algorithm 3 An ϵ-Oracle Based Algorithm

Require: An ϵ-Oracle of probability distribution μ denoted by ϵ-Oracle(μ), sample size K.
Ensure: $\widehat{\phi}_n(V)$
1: $\widehat{\phi}_n(V) \leftarrow 0$;
2: **for** $k = 1$ to K **do**
3: Generate a ordering $\boldsymbol{\sigma}$ from ϵ-Oracle(μ):

$$\boldsymbol{\sigma} \sim \ \epsilon\text{-Oracle}\ (\mu);$$

4: $\widehat{\phi}_n(V) \leftarrow \left(1 - \frac{1}{k}\right)\widehat{\phi}_n(V) + \frac{1}{k}\Delta(\boldsymbol{\sigma}, n)$;
5: **end for**

Theorem 2: The approximation error of Algo. 3 is bounded by

$$\epsilon_{app} \leq 2\epsilon \max_{\boldsymbol{\sigma}\in\Omega} |\Delta(\boldsymbol{\sigma}, n)|,$$

while the estimation error of Algorithm 3 is bounded by

$$\epsilon_{est} \leq \frac{\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)}{\sqrt{K}}\sqrt{\frac{1}{2}\ln\frac{2}{\delta}}$$

with a probability of at least $1 - \delta$, where $\delta \in (0, 1]$.

Theorem 1: Proofs are in our supplementary file.

Theorem 2 reveals that the upper bound of the approximation error is linear in ϵ with a rate of $\max_{\boldsymbol{\sigma}\in\Omega}|\Delta(\boldsymbol{\sigma}, n)|$. This implies that when $\max_{\boldsymbol{\sigma}\in\Omega}|\Delta(\boldsymbol{\sigma}, n)|$ is large, one needs to have a more accurate oracle, i.e., with smaller ϵ, to attain a small approximation error. The estimation error is linear in $[\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)]/\sqrt{K}$. This implies that given an estimation error, the number of samples needed is proportional to $[\max_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n) - \min_{\boldsymbol{\sigma}\in\Omega}\Delta(\boldsymbol{\sigma}, n)]^2$.

The computational complexity of Algorithm 3 is

$$O(K \times \textit{complexity of generating a sample from } \epsilon\text{-Oracle}(\mu)).$$

We next design algorithm to implement the ϵ-Oracle and study its computational complexity.

Implementing the ϵ-Oracle. Recall that the distribution μ under the pair-wise cooperation preference is expressed in Equation (9). We design a Markov Chain Monte Carlo algorithm to implement the ϵ-Oracle for μ, i.e., approximate IID samples. The state space of the Markov Chain is Ω, i.e., each state corresponds to one ordering $\boldsymbol{\sigma}$. We need to design state transition probabilities such that the stationary distribution of the Markov chain is $\mu(\boldsymbol{\sigma})$ as stated in Equation (9). To achieve this, let us first define the neighbor of a state $\boldsymbol{\sigma}$.

Definition 3: The state $\tilde{\boldsymbol{\sigma}}$ and $\boldsymbol{\sigma}$ are mutual neighbors if and only if they are different on at most two coordinates. We define a set of all neighbors of $\boldsymbol{\sigma}$ as

$$\mathcal{F}(\boldsymbol{\sigma}) \triangleq \{\tilde{\boldsymbol{\sigma}} | \tilde{\boldsymbol{\sigma}} \textit{ and } \boldsymbol{\sigma} \textit{ are mutual neighbors}\}.$$

For example, $\tilde{\boldsymbol{\sigma}} = (1,3,2)$ and $\boldsymbol{\sigma} = (1,2,3)$ are mutual neighbors, while $\tilde{\boldsymbol{\sigma}} = (3,1,2)$ and $\boldsymbol{\sigma} = (1,2,3)$ are not mutual neighbors. Furthermore, the neighbor set of $\boldsymbol{\sigma} = (1,2,3)$ is $\mathcal{F}(\boldsymbol{\sigma}) = \{(1,2,3),(1,3,2),(3,2,1),(2,1,3)\}$. A state $\boldsymbol{\sigma}$ can only transit to one of its neighbors. The number of neighbors for each state is $|\mathcal{F}(\boldsymbol{\sigma})| = \binom{N}{2} + 1$. This implies a sparse transition matrix as the number of states is $N!$. A sparse transition matrix can make the simulation of transitions of the Markov chain computationally inexpensive. TWe apply the glauber dynamics [26] framework to design the transition probabilities. We outline the details in Algorithm 4 and denote the transition probability as $\boldsymbol{P}$. Thus, the Markov chain is $(\Omega, \boldsymbol{P})$.

Algorithm 4 ϵ-Oracle for Pair-Wise Preference

Require: Pair-wise preference $\mathcal{G}$, the function $\Upsilon(\cdot)$, the number of rounds T and $\boldsymbol{\sigma}^{(0)}$.

Ensure: $\boldsymbol{\sigma}^{(T)}$

1: **for** $t = 0$ to $T-1$ **do**

2: Generate n uniformly at random from $\mathcal{N}$, Generate m uniformly at random from $\mathcal{N} \setminus \{n\}$;

3: $\tilde{\boldsymbol{\sigma}} \leftarrow \boldsymbol{\sigma}^{(t)}$ swapping $\sigma_m^{(t)}$ and $\sigma_n^{(t)}$;

4: $\boldsymbol{\sigma}^{(t+1)} \leftarrow \boldsymbol{\sigma}^{(t)}$ with probability

$$\frac{\exp(\alpha\Upsilon(\boldsymbol{\sigma}^{(t)}))}{\exp(\alpha\Upsilon(\boldsymbol{\sigma}^{(t)})) + \exp(\alpha\Upsilon(\tilde{\boldsymbol{\sigma}}))},$$

and $\boldsymbol{\sigma}^{(t+1)} \leftarrow \tilde{\boldsymbol{\sigma}}$ with probability

$$\frac{\exp(\alpha\Upsilon(\tilde{\boldsymbol{\sigma}}))}{\exp(\alpha\Upsilon(\boldsymbol{\sigma}^{(t)})) + \exp(\alpha\Upsilon(\tilde{\boldsymbol{\sigma}}))};$$

5: **end for**

To illustrate, consider the setting of Example 1. For simplicity, suppose $\alpha = 0$. Suppose the initialized ordering is $\boldsymbol{\sigma}^{(0)} = (1,2,3)$. In the first round, i.e., $t = 0$, suppose the generated n and m are $n = 1$ and $m = 2$ respectively. Note that $\sigma_1^{(0)} = 1$ and $\sigma_2^{(0)} = 2$. Hence, swapping $\sigma_1^{(0)}$ and $\sigma_2^{(0)}$ leads to $\tilde{\boldsymbol{\sigma}} = (2,1,3)$. Finally, with probability $1/2$ the $\boldsymbol{\sigma}^{(1)}$ is set as $\boldsymbol{\sigma}^{(1)} \leftarrow (1,2,3)$, and with probability $1/2$ the $\boldsymbol{\sigma}^{(1)}$ is set as $\boldsymbol{\sigma}^{(1)} \leftarrow (2,1,3)$.

The computational complexity of Algorithm 4 is

$$O(T \times \textit{complexity of simulating one transition}),$$

where T denotes the simulation rounds. The cost of simulating one transition has two key components: (1) drawing a uniform sample from $\mathcal{N}$ or from $\mathcal{N} \setminus \{n\}$; (2) drawing a sample from a Bernoulli distribution. In other words, simulating one transition is computationally inexpensive. The remaining question is: how many rounds of simulation do we need? The number of simulation rounds needed is closely related to the local variation of $\mu(\boldsymbol{\sigma})$ defined as follows.

Definition 4: We define the local variation of $\mu(\boldsymbol{\sigma})$ as

$$\|\mu\|_{LV} \triangleq \max_{\boldsymbol{\sigma}\in\Omega} \max_{\tilde{\boldsymbol{\sigma}}\in\mathcal{F}(\boldsymbol{\sigma})} \frac{\mu(\boldsymbol{\sigma})}{\mu(\tilde{\boldsymbol{\sigma}})}.$$

For example, consider the $\mu(\boldsymbol{\sigma})$ derived in Equation (9), the local variation of μ can be derived as

$$\|\mu\|_{LV} = \exp\left[\alpha \max_{\boldsymbol{\sigma}\in\Omega} \max_{\tilde{\boldsymbol{\sigma}}\in\mathcal{F}(\boldsymbol{\sigma})} (\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}}))\right].$$

Note that the local variation $\|\mu\|_{LV} \geq 1$, because $\boldsymbol{\sigma} \in \mathcal{F}(\boldsymbol{\sigma})$ and $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}$ gives a lower bound of 1.

Theorem 3: Consider the $\mu(\boldsymbol{\sigma})$ derived in Equation (9). Suppose $\|\mu\|_{LV} \leq \binom{N}{2}/(\binom{N}{2} - 2)$, i.e., α satisfies

$$\alpha \leq \frac{\ln[\binom{N}{2}/(\binom{N}{2} - 2)]}{\max_{\boldsymbol{\sigma}\in\Omega} \max_{\tilde{\boldsymbol{\sigma}}\in\mathcal{F}(\boldsymbol{\sigma})} (\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}}))}. \tag{11}$$

If the number of rounds T satisfies

$$T \geq \left(\ln \frac{\epsilon}{N}\right) / \ln\left(\frac{\binom{N}{2} - 1}{\binom{N}{2}} \frac{2\|\mu\|_{LV}}{\|\mu\|_{LV} + 1}\right),$$

Algorithm 4 implements an ϵ-Oracle of μ.

Theorem 3 derives an upper bound on the local variation of μ, such that Algorithm 4 can implement an ϵ-Oracle of μ. For each μ that satisfies the local variation upper bound, Theorem 3 also derives the number of simulation rounds needed to implement the ϵ-Oracle. The local variation upper bound is equivalent to an upper bound of the preference sensitivity α, which implies a small α.

C. An Importance Sampling Based Algorithm

We consider the pair-wise preference, in particular, the case that the preference parameter α does not satisfy (11).

Transformation via importance sampling. Recall the pair-wise preference aware Shapley value, where the distribution μ is derived in Equation (9). One challenge is that the denominator of Equation (9) is computationally expensive to compute. One way to address this challenge is via estimating the ratio $\phi_n(V)/\phi_1(V), \forall n \in \mathcal{N}$. And then reconstruct $\phi_n(V), \forall n \in \mathcal{N}$ via the efficiency property, i.e., $\sum_{n\in\mathcal{N}} \phi_n(V) = V(\mathcal{N})$. We apply the importance sampling framework to estimate the ratio $\phi_n(V)/\phi_1(V), \forall n \in \mathcal{N}$. We show the key idea in the following derivation:

$$\begin{aligned}
\frac{\phi_n(V)}{\phi_1(V)} &= \frac{\mathbb{E}_{\boldsymbol{\sigma}\sim\mu}[\Delta(\boldsymbol{\sigma}, n)]}{\mathbb{E}_{\boldsymbol{\sigma}\sim\mu}[\Delta(\boldsymbol{\sigma}, 1)]} \\
&= \frac{\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[N!\mu(\boldsymbol{\sigma})\Delta(\boldsymbol{\sigma}, n)]}{\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[N!\mu(\boldsymbol{\sigma})\Delta(\boldsymbol{\sigma}, 1)]} \qquad (12) \\
&= \frac{\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_n(\boldsymbol{\sigma})]}{\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_1(\boldsymbol{\sigma})]}, \qquad (13)
\end{aligned}$$

where $Uniform(\Omega)$ denotes a uniform distribution over Ω and

$$X_n(\boldsymbol{\sigma}) \triangleq \exp(\alpha\Upsilon(\boldsymbol{\sigma}))\Delta(\boldsymbol{\sigma}, n).$$

In the above derivation, (12) follows the importance sampling framework (or change of measure), and (13) follows the linear scaling property of expectation. Combining (13) with the efficiency property $\sum_{n\in\mathcal{N}} \phi_n(V) = V(\mathcal{N})$, we have

$$\phi_n(V) = \frac{\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_n(\boldsymbol{\sigma})]}{\sum_{m\in\mathcal{N}} \mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_m(\boldsymbol{\sigma})]} V(\mathcal{N}). \tag{14}$$

Now we only need to deal with a uniform distribution over Ω and random variable $X_n(\boldsymbol{\sigma})$.

Design and analysis of the importance sampling based algorithm. Based on the above derivation, we propose a sampling algorithm, i.e., Algorithm 5, which approximates the Shapley value under pair-wise cooperation preference. Algorithm 5 generates K orderings from $Uniform(\Omega)$, i.e., the uniform distribution over Ω. Then, it uses each generated ordering to compute one sample of $X_n(\boldsymbol{\sigma})$. The average of these samples is used to estimate $\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_n(\boldsymbol{\sigma})]$. Finally, the estimated $\mathbb{E}_{\boldsymbol{\sigma}\sim Uniform(\Omega)}[X_n(\boldsymbol{\sigma})]$ is used to estimate the Shapley value according to Equation (14).

Algorithm 5 Importance Sampling Based Algorithm

Require: Pair-wise preference $\mathcal{G}$, the function $\Upsilon(\cdot)$, the sample complexity K.
Ensure: $\widehat{\phi}_n(V), \forall n \in \mathcal{N}$
1: $x_n \leftarrow 0$;
2: **for** $k = 1$ to K **do**
3: Generate an ordering $\boldsymbol{\sigma} \sim Uniform(\Omega)$;
4: $x_n \leftarrow \left(1 - \frac{1}{k}\right) x_n + \frac{1}{k} X_n(\boldsymbol{\sigma}), \forall n \in \mathcal{N}$;
5: **end for**
6: $\widehat{\phi}_n(V) \leftarrow \frac{x_n}{\sum_{m\in\mathcal{N}} x_m} V(\mathcal{N}), \forall n \in \mathcal{N}$

To illustrate, consider the setting of Example 1. For simplicity, suppose $K = 2$ and $\alpha = 0$. In the first round, i.e., $k = 1$, suppose the generated ordering is $\boldsymbol{\sigma} = (1, 2, 3)$. It follows that $\Delta(\boldsymbol{\sigma}, 1) = 0$ and correspondingly $X_1(\boldsymbol{\sigma}) = \exp(\alpha\Upsilon(\boldsymbol{\sigma}))\Delta(\boldsymbol{\sigma}, 1) = 0$. The weight x_1 is then updated as $x_1 \leftarrow (1 - 1/1) \times 0 + 0 = 0$. Similarly, we have $x_2 \leftarrow 0$ and $x_3 \leftarrow 10$. In the second round, i.e., $k = 2$, suppose the generated ordering is $\boldsymbol{\sigma} = (3, 2, 1)$. It follows that $\Delta(\boldsymbol{\sigma}, 1) = 10$ and correspondingly the weight x_1 is updated as $x_1 \leftarrow (1-\frac{1}{2})\times 0 + \frac{1}{2} \times 10 = 5$. Similarly, we have $x_2 \leftarrow 0$ and $x_3 \leftarrow 5$. Finally, the estimated Shapley value for player 1 can be calculated as $\widehat{\phi}_1(V) \leftarrow 10 \times 5/(5 + 0 + 5) = 5$. Similarly, we have $\widehat{\phi}_2(V) \leftarrow 0$ and $\widehat{\phi}_3(V) \leftarrow 5$.

The computational complexity of Algorithm 5 is

$$O(K \times \textit{complexity of generating } \boldsymbol{\sigma} \sim Uniform(\Omega)).$$

One method of generating $\boldsymbol{\sigma} \sim Uniform(\Omega)$ is: generate an index n uniformly at random from $\mathcal{N}$ and set $\sigma_1 = n$; generate an index m uniformly at random from $\mathcal{N} \setminus \{n\}$ and set $\sigma_2 = m$; repeat this process until we obtain σ_N. Thus, the cost of generating $\boldsymbol{\sigma} \sim Uniform(\Omega)$ is small. The remaining question is how many samples do we need (i.e., the sample complexity) such that Algorithm 5 produces an accurate estimation. This sample complexity is closely related to the global variation of $\mu(\boldsymbol{\sigma})$ and $\Delta(\boldsymbol{\sigma}, n)$, which is defined as follows.

Definition 5: We define the global variation of $\mu(\boldsymbol{\sigma})$ and $\Delta(\boldsymbol{\sigma}, n)$ as

$$\|\mu\|_{GV} \triangleq \max_{\boldsymbol{\sigma},\tilde{\boldsymbol{\sigma}}\in\Omega} \frac{\mu(\boldsymbol{\sigma})}{\mu(\tilde{\boldsymbol{\sigma}})}, \quad \|\Delta\|_{GV} \triangleq \max_{n\in\mathcal{N}} \max_{\boldsymbol{\sigma},\tilde{\boldsymbol{\sigma}}\in\Omega} \frac{\Delta(\boldsymbol{\sigma}, n)}{\Delta(\tilde{\boldsymbol{\sigma}}, n)}.$$

For example, consider the $\mu(\boldsymbol{\sigma})$ derived in Equation (9), the global variation of μ can be derived as

$$\|\mu\|_{GV} = \exp\left[\alpha \max_{\boldsymbol{\sigma},\tilde{\boldsymbol{\sigma}}\in\Omega} (\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}}))\right].$$

Based on the notion of global variation, we next derive the sample complexity of Algorithm 5.

Theorem 4: Suppose $\Delta(\boldsymbol{\sigma}, n)$ satisfy $\Delta(\boldsymbol{\sigma}, n) > 0, \forall \boldsymbol{\sigma}, n$. If the number of samples K satisfies

$$K \geq \frac{1}{\varepsilon^2} \|\mu\|_{GV}^2 \|\Delta\|_{GV}^2 \ln \frac{2N}{\delta},$$

then Algorithm 5 has the following accuracy guarantee

$$\mathbb{P}\left[\frac{1-\varepsilon}{1+\varepsilon}\phi_n(V) \leq \widehat{\phi}_n(V) \leq \frac{1+\varepsilon}{1-\varepsilon}\phi_n(V), \forall n\right] \geq 1 - \delta.$$

where $\varepsilon \in (0, 1]$.

Theorem 4 derives the number of samples such that Algorithm 5 produces an accurate estimation. Note that it only holds for a class of payoff functions where there is a positive externality of increasing the coalition size. It states that the sample complexity is quadratic in the global variation of μ and $\Delta(\boldsymbol{\sigma}, n)$ respectively. Namely, to make the sample size polynomial, we need $\|\Delta\|_{GV}^2 \leq \mathtt{poly}(N)$ and

$$\alpha \leq \frac{\ln \mathtt{poly}(N)}{\max_{\boldsymbol{\sigma},\tilde{\boldsymbol{\sigma}}\in\Omega}(\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}}))}. \tag{15}$$

One can easily show

$$\max_{\boldsymbol{\sigma},\tilde{\boldsymbol{\sigma}}\in\Omega} (\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}})) \leq N \max_{\boldsymbol{\sigma}\in\Omega} \max_{\tilde{\boldsymbol{\sigma}}\in\mathcal{F}(\boldsymbol{\sigma})} (\Upsilon(\boldsymbol{\sigma}) - \Upsilon(\tilde{\boldsymbol{\sigma}})),$$

and $\mathtt{poly}(N) \geq [\binom{N}{2}/(\binom{N}{2} - 2)]^N$. Thus, the upper bound in (15) is larger than that in (11). This implies that Algorithm 5 can handle larger range of α than Algorithm 4. But this is achieved when extra conditions on $\Delta(\boldsymbol{\sigma}, n)$ hold.

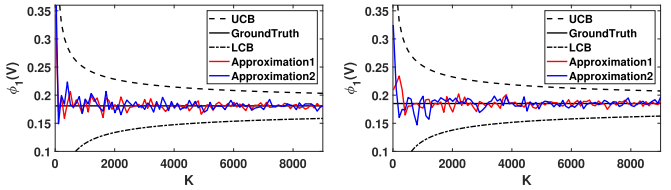
D. A Random Walk Based Algorithm

We still consider the pair-wise cooperation preference, i.e., $\mu(\boldsymbol{\sigma})$ satisfies Equation (9). Here we focus on general sensitivity parameter α and payoff function V, i.e., not restricted to the conditions in Theorem 3 and 4. We design an algorithm based on random walk to approximate the Shapley value. In particular, the random walk is associated with a Markov Chain in Section III-B, i.e., $(\Omega, \boldsymbol{P})$. In other words, the walker walks on the states in Ω and jumps to each neighbor according to the transition probability matrix $\boldsymbol{P}$. Recall that the Markov chain $(\Omega, \boldsymbol{P})$ is ergodic and has a stationary distribution μ. We estimate the Shapley value using the samples $\boldsymbol{\sigma}$ generated by the random walker. We outline the details in Algorithm 6.

To illustrate, consider the setting of Example 1. For simplicity, suppose $T = 2$. In the first round, i.e., $t = 0$, suppose the generated ordering is $\boldsymbol{\sigma}^{(1)} = (1, 2, 3)$. It follows that $\Delta(\boldsymbol{\sigma}^{(1)}, 1) = 0$. The estimated Shapley value of player 1 can be updated as $\widehat{\phi}_1(V) \leftarrow (1 - 1/1) \times 0 + \Delta(\boldsymbol{\sigma}^{(1)}, 1) = 0$. Similarly, we have $\widehat{\phi}_2(V) \leftarrow 0$ and $\widehat{\phi}_3(V) \leftarrow 10$. In the second round, i.e., $t = 1$, suppose the generated ordering is $\boldsymbol{\sigma}^{(2)} = (3, 2, 1)$. It follows that $\Delta(\boldsymbol{\sigma}^{(2)}, 1) = 10$. The estimated Shapley value of player 1 in this round can be

Algorithm 6 Random Walk Based Algorithm

Require: Pair-wise preference $\mathcal{G}$, the function $\Upsilon(\cdot)$, the number of walks T and $\boldsymbol{\sigma}^{(0)}$.
Ensure: $\widehat{\phi}_n(V), \forall n \in \mathcal{N}$
1: $\widehat{\phi}_n(V) \leftarrow 0, \forall n \in \mathcal{N}$
2: **for** $t = 0$ to $T-1$ **do**
3: Step 2-4 of Algorithm 4.
4: $\widehat{\phi}_n(V) \leftarrow (1-\frac{1}{t+1})\widehat{\phi}_n(V) + \frac{1}{t+1}\Delta(\boldsymbol{\sigma}^{(t+1)}, n), \forall n \in \mathcal{N}$
5: **end for**



(a) $\theta = 0.4$ (b) $\theta = 1.6$

Fig. 2. Impact of number of samples on the tightness of Theorem 1 with $\delta = 0.1$.

updated as $\widehat{\phi}_1(V) \leftarrow (1-\frac{1}{2}) \times 0 + \frac{1}{2} \times 10 = 5$. Similarly, we have $\widehat{\phi}_2(V) \leftarrow 0$ and $\widehat{\phi}_3(V) \leftarrow 5$.

The computational complexity of Algorithm 6 is

$$O(T \times \textit{complexity of simulating one transition}),$$

where T denotes the simulation rounds. The complexity of simulating one transition is the same as that of Algorithm 4. The $\widehat{\phi}_n(V)$ converges to $\phi_n(V)$ asymptotically, because the Markov chain is ergodic. One can easily verify that the Markov chain $(\Omega, \boldsymbol{P})$ is reversible. This means that we can use the spectral gap of the Markov chain $(\Omega, \boldsymbol{P})$ to quantify the convergence rate of Algorithm 6. This convergence rate implies the number of walks that (i.e., sample complexity) we need. Formally, let β denote the spectral gap of the Markov chain $(\Omega, \boldsymbol{P})$. Let $\mu^{(0)}$ denote the distribution of the initial state $\boldsymbol{\sigma}^{(0)}$. We define the distance between $\mu^{(0)}$ and μ as

$$D\left(\mu^{(0)}, \mu\right) \triangleq \sqrt{\sum_{\boldsymbol{\sigma} \in \Omega} \left(\frac{\mu^{(0)}(\boldsymbol{\sigma})}{\mu(\boldsymbol{\sigma})}\right)^2}.$$

Based on these notations, we derive the sample complexity of Algorithm 6 in the following theorem.

Theorem 5: Suppose the number of walks satisfies

$$T \geq \frac{1}{\beta \tilde{\varepsilon}^2} \frac{16\|\Delta\|_\infty^2 b}{1 - 1.25\tilde{\varepsilon}/(\|\Delta\|_\infty b)} \ln\left[D(\mu^{(0)}, \mu)\frac{2N}{\delta}\exp\frac{\beta}{5}\right]. \quad (16)$$

Algorithm 6 has the following accuracy guarantee

$$\mathbb{P}\left[\left|\widehat{\phi}_n(V) - \phi_n(V)\right| < \tilde{\varepsilon}, \forall n\right] \geq 1 - \delta.$$

where $\|\Delta\|_\infty \triangleq \max_{n \in \mathcal{N}} \max_{\boldsymbol{\sigma} \in \Omega} |\Delta(\boldsymbol{\sigma}, n)|$, $b = \sum_{\boldsymbol{\sigma} \in \Omega} \left(\frac{\Delta(\boldsymbol{\sigma}, n) - \phi_n(V)}{2\|\Delta\|_\infty}\right)^2$ *and* $\tilde{\varepsilon} \leq 0.8\, b\|\Delta\|_\infty$.

Theorem 5 states that the sample complexity (or the number of walks needed) of Algorithm 6 is roughly proportional to $1/\epsilon^2$, inverse of spectral gap β and $\|\Delta\|_\infty^2$. Namely, it is critical to estimation accuracy ϵ, inverse of spectral gap β and $\|\Delta\|_\infty$. The sample complexity is not critical to the initial distribution, because it is proportional to the logarithmic of the distance, i.e., $\ln D(\mu^{(0)}, \mu)$.

It is important to note that Algorithm 6 works on general preference sensitivity α and payoff functions V. This is an advantage over Algorithm 3 and Algorithm 5. However, one shortcoming of algorithm 6 is that the spectral gap is computational expensive to evaluate, i.e., a complexity of higher than $|\Omega| = N!$. This makes it challenging to determine the sample complexity in practice. In practice, one can address this challenge by applying some convergence diagnosing methods [27]. We omit the details as it has been studied extensively in the literature of MCMC.

E. Discussions

Tightness of bound. All the error bounds are built on distribution-independent concentration inequalities. They are tight in terms of the order with respect to sample size K as indicated by the anti-concentration inequalities [28]. But the bound is not tight with respect to the multiplying factor as illustrated in Figure 2. Figure 2 illustrates the ground truth of Shapley value, the upper confidence bound (UCB) and lower confidence bound (LCB) produced by Theorem 1, and two approximation paths produced by combining Algo. 1 and 2. The setting of Figure 2 is the same as Figure 3, whose detail is deferred to Section V. One can observe that the estimated Shapley value produced by combining Algo. 1 and 2 is much closer to the ground truth than the UCB and LCB produced by Theorem 1. This implies that the error bound is not tight with respect to the multiplying factor. We are also aware of other concentration inequalities that capture more properties of the distribution such as variance [29] and they can improve the multiplying factor of the error bound a little bit. We do not choose them because they also make the error bound more complicated, making factors that affect the order of bound less clear.

Novelty. Similar with works in the research line of studying computational aspect of Shapley value [19], [20], [30], [31], our work employs techniques from randomized algorithms and concentration inequalities to design algorithms and derive error bounds that reveal the impact of various factors such local variation and global variation on the approximation accuracy of cooperation preference aware Shapley value.

IV. Extensions

In this section, we present two extensions of our model: *(1) multiple coalitions; (2) dynamic cooperation preferences.* We focus on extending the cooperation preference aware Shapley value to divide payoff, and extending the algorithmic framework in Section III to address the computational issues arose from these two extensions.

A. Multiple Coalitions

We first present models on the formation of multiple coalitions. Then, we extend our cooperation preference aware

Shapley value to divide the payoff as well as extend the algorithms to compute the resulting Shapley value.

Multiple coalition formation. In real-world applications, multiple coalitions usually arise, instead of just one grand coalition as presented in Section II. Both the payoff function and the cooperation preference may result in multiple coalitions.

Let us first consider the case of multiple coalitions induced by the payoff function. To illustrate, consider the following example of three players, i.e., $\mathcal{N} = \{1, 2, 3\}$ with a payoff function V as follows:

$$\begin{aligned} V(\emptyset) &= 0, \\ V(\{1\}) = 5, \quad V(\{2\}) &= 5, \ V(\{3\}) = 5, \\ V(\{1,2\}) = 100, \quad V(\{1,3\}) &= 1, \ V(\{2,3\}) = 1, \\ V(\{1,2,3\}) &= 0.5. \end{aligned}$$

Suppose that all players have equal preference to cooperate with other players. Then, the coalitions that maximizes each user's payoff are $\{1, 2\}$ and $\{3\}$, where both players 1 and 2 get 50 and player 3 gets 5. This implies that the final coalition outcomes will most likely be $\{1, 2\}$ and $\{3\}$. Many previous works studied the stability of coalitions [32] or formation of coalitions [33], [34], [35] from the perspective of payoff functions.

Besides the above multiple coalition formation models or methods, the cooperation preference in Section II can also be extended to model multiple coalitions.

• ***Multiple coalitions via group-wise preference.*** Our group-wise cooperative preference model can be extended to model the formation of multiple coalitions by providing an option of not joining any existing coalitions, but instead forming a new singleton coalition by itself. Recall the group-wise preference model that there is a coalition $\mathcal{C}_t$ at time slot t and a player joins the coalition $\mathcal{C}_t$ (with probability $p(n|\mathcal{C}_t)$ the joining user is $n \in \mathcal{N} \setminus \mathcal{C}_t$). Let $\ell_t \in \mathbb{N}_+$ denote the number of nonempty coalitions formed up to the time slot t. Denote these coalitions by $\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_t}$, where $\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_t}$ are disjoint and $\widetilde{\mathcal{C}}_1 \cup \ldots \cup \widetilde{\mathcal{C}}_{\ell_t} = \mathcal{C}_t$. The joining user $n \in \mathcal{N} \setminus \mathcal{C}_t$ can join coalition $\widetilde{\mathcal{C}}_i$, where $i \in \{1, \ldots, \ell_t\}$ with probability $p(i|\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_t})$ or forming a new singleton coalition by itself with probability $p(0|\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_t})$, where $\sum_{i=0}^{\ell_t} p(i|\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_t}) = 1$. In the end, a number of ℓ_N coalitions will be formed, i.e., $\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_{\ell_N}$.

• ***Multiple coalitions via pair-wise preference.*** Our pair-wise cooperation preference model can also be extended to model the formation of multiple coalitions via subgraph components. Suppose that the pair-wise preference graph $\mathcal{G} = (\mathcal{N}, \boldsymbol{W})$ can be partitioned into $\ell \in \mathbb{N}_+$ isolated components $\widetilde{\mathcal{N}}_1, \ldots, \widetilde{\mathcal{N}}_\ell$ such that there is no edge between any two components $\widetilde{\mathcal{N}}_i$ and $\widetilde{\mathcal{N}}_j$, and each component $\widetilde{\mathcal{N}}_i$ can not be partitioned into smaller isolated components. Each isolated component corresponds to one coalition. Thus, the pair-wise cooperation preference graph with ℓ isolated components induces ℓ coalitions denoted by $\widetilde{\mathcal{N}}_1, \ldots, \widetilde{\mathcal{N}}_\ell$.

To summarize, the above models separately use either the payoff function or the cooperation preference to model multiple coalitions. One can also combine the payoff function and the cooperation preference model to model multiple coalitions. In the following, we focus on how to divide the payoff when multiple coalitions have been formed. Without loss of generality, we consider that ℓ nonempty coalitions has been formed denoted by

$$\mathcal{N}_1, \ldots, \mathcal{N}_\ell,$$

where $\ell \in \{1, \ldots, N\}$, the coalitions $\mathcal{N}_1, \ldots, \mathcal{N}_\ell$ are disjoint and satisfy $\mathcal{N}_1 \cup \ldots \cup \mathcal{N}_\ell = \mathcal{N}$.

Shapley value under multiple coalitions. We consider two extensions of the cooperation preference aware Shapley value to divide the payoff under multiple coalitions. The first extension considers the case that the payoff among different coalitions are independent. The second extension considers the case that the payoff among different coalitions mutually influence each other, capturing externalities such as competition among coalitions. Let $\mathcal{N}_{-i}$ denote a set of all the coalitions except the coalition $\mathcal{N}_i$, i.e.,

$$\mathcal{N}_{-i} \triangleq \{\mathcal{N}_1, \ldots, \mathcal{N}_{i-1}, \mathcal{N}_{i+1}, \ldots, \mathcal{N}_\ell\}.$$

• ***Independent payoff across coalitions.*** The payoff across different coalitions are independent, i.e., for any $\mathcal{C} \subseteq \mathcal{N}_i$, the payoff for $\mathcal{C}$ is uniquely determined by $\mathcal{C}$ itself and it is not influenced by the other coalitions $\mathcal{N}_{-i}$. Without loss of generality, denote the payoff function associated with coalition $\mathcal{N}_i$ as V_i. Thus, the payoff for $\mathcal{C} \subseteq \mathcal{N}_i$ is $V_i(\mathcal{C})$. Let $\phi_{i,n}(V_i)$ denote the payoff share or Shapley value of user $n \in \mathcal{N}_i$. Then, $\phi_{i,n}(V_i)$ can be expressed as

$$\phi_{i,n}(V_i) = \sum_{\boldsymbol{\sigma}_i \in \Omega_i} \mu_i(\boldsymbol{\sigma}_i) \left[V_i(\mathcal{S}_n^{\boldsymbol{\sigma}_i} \cup \{n\}) - V_i(\mathcal{S}_n^{\boldsymbol{\sigma}_i}) \right],$$

where Ω_i denotes a set of all of the orderings of the players in coalition $\mathcal{N}_i$, i.e., $\Omega_i \triangleq \{\boldsymbol{\sigma} | \boldsymbol{\sigma}$ is an ordering of players in $\mathcal{N}_i\}$, and μ_i denotes a probability distribution over Ω_i. Applying the cooperation preference model in Section II, we can obtain closed form expressions for the distribution μ_i under the group-wise preference or pair-wise preference. One can apply the algorithms in Section III to compute the Shapley value $\phi_{i,n}(V_i)$ for each coalition $\mathcal{N}_i$.

• ***Coalition externalities aware payoff.*** The payoff of a coalition $\mathcal{N}_i$ is influenced by other coalitions $\mathcal{N}_{-i}$. Formally, let $\widetilde{V}_i(\cdot; \mathcal{N}_{-i})$ denote the payoff function associated with the coalition $\mathcal{N}_i$. Namely, the payoff for the set of players $\mathcal{C} \in \mathcal{N}_i$ is $\widetilde{V}_i(\mathcal{C}; \mathcal{N}_{-i})$. The parameter $\mathcal{N}_{-i}$ in the payoff function $\widetilde{V}_i(\mathcal{C}; \mathcal{N}_{-i})$ can be used to capture externalities among coalitions such as competition or complementarity among coalitions. Let $\phi_{i,n}(\widetilde{V}_i)$ denote the payoff share or Shapley value of user $n \in \mathcal{N}_i$. Then, $\phi_{i,n}(\widetilde{V}_i)$ can be expressed as

$$\phi_{i,n}\left(\widetilde{V}_i\right) = \sum_{\boldsymbol{\sigma}_i \in \Omega_i} \mu_i(\boldsymbol{\sigma}_i) \Big[\widetilde{V}_i(\mathcal{S}_n^{\boldsymbol{\sigma}_i} \cup \{n\}; \mathcal{N}_{-i}) - \widetilde{V}_i(\mathcal{S}_n^{\boldsymbol{\sigma}_i}; \mathcal{N}_{-i}) \Big].$$

Applying the cooperation preference model in Section II, we can obtain closed form expressions for the distribution μ_i under the group-wise preference or pair-wise preference. Then one can apply algorithms in Section III to compute the Shapley value $\phi_{i,n}(\widetilde{V}_i)$ for each coalition $\mathcal{N}_i$.

B. Dynamic Cooperation Preferences

We extend our model to capture dynamic cooperation preferences evolving over time. This paper only focuses on dividing the payoff under the evolving dynamics of cooperation preferences. For the analysis, control, etc., of the dynamic cooperation preferences, they are interesting research directions for future research.

Dynamic cooperation preferences. We use a discrete time system indexed by $\tau \in \mathbb{N}_+$ to model the dynamic cooperation preference. With a little abuse of notations, let $\ell_\tau \in \mathcal{N}_+$ denote the number of nonempty coalitions in time slot τ. Denote these $\ell_\tau \in \mathbb{N}_+$ nonempty coalitions as $\mathcal{N}_1^{(\tau)}, \ldots, \mathcal{N}_{\ell_\tau}^{(\tau)}$, where the coalitions $\mathcal{N}_1^{(\tau)}, \ldots, \mathcal{N}_{\ell_\tau}^{(\tau)}$ are disjoint and satisfy

$$\mathcal{N}_1^{(\tau)} \cup \ldots \cup \mathcal{N}_{\ell_\tau}^{(\tau)} = \mathcal{N}.$$

Let $\mathcal{N}_{-i}^{(\tau)}$ denote a set of all the coalitions except $\mathcal{N}_i^{(\tau)}$, i.e.,

$$\mathcal{N}_{-i}^{(\tau)} \triangleq \left\{ \mathcal{N}_1^{(\tau)}, \ldots, \mathcal{N}_{i-1}^{(\tau)}, \mathcal{N}_{i+1}^{(\tau)}, \ldots, \mathcal{N}_{\ell_\tau}^{(\tau)} \right\}.$$

Let $\Omega_i^{(\tau)}$ denote a set of all orderings of players in the coalition $\mathcal{N}_i^{(\tau)}$, i.e.,

$$\Omega_i^{(\tau)} \triangleq \left\{ \boldsymbol{\sigma} \middle| \boldsymbol{\sigma} \text{ is an ordering of players in } \mathcal{N}_i^{(\tau)} \right\}.$$

For brevity, we only consider the case that the payoff function across different coalitions mutually influence each other. Let $\widetilde{V}_i^{(\tau)}(\cdot; \mathcal{N}_{-i}^{(\tau)})$ denote the payoff function associated with the coalition $\mathcal{N}_i^{(\tau)}$ in time slot τ. Namely, in time slot τ, the payoff for the set of players $\mathcal{C} \in \mathcal{N}_i^{(\tau)}$ is $\widetilde{V}_i^{(\tau)}(\mathcal{C}; \mathcal{N}_{-i}^{(\tau)})$. Let $\phi_{i,n}^{(\tau)}(\widetilde{V}_i^{(\tau)})$ denote the payoff share or Shapley value of player $n \in \mathcal{N}_i^{(\tau)}$ in time slot τ. In general, the cooperation preference across different time slots are mutually dependent. We do not model such dependence, because it does not affect the application of our algorithms to divide the payoff in different time slots.

Shapley value under dynamic cooperation preferences. Consider the group-wise preference. Denote the group-wise cooperation preference profile for all players in time slot $\tau \in \mathbb{N}_+$ as

$$\mathscr{R}^{(\tau)} \triangleq \left\{ \mathcal{R}_n^{(\tau)} \middle| \forall n \in \mathcal{N} \right\}.$$

Let $\mu_i^{(\tau)}(\cdot|\mathscr{R}^{(\tau)})$ denote the probability distribution over $\Omega_i^{(\tau)}$ under the cooperation preference profile $\mathscr{R}^{(\tau)}$. Then, $\phi_{i,n}^{(\tau)}(\widetilde{V}_i^{(\tau)})$ can be expressed as

$$\begin{aligned}\phi_{i,n}^{(\tau)}&\left(\widetilde{V}_i^{(\tau)}\right) \\ &= \sum_{\boldsymbol{\sigma}_i \in \Omega_i^{(\tau)}} \mu_i^{(\tau)}\left(\boldsymbol{\sigma}_i \middle| \mathscr{R}^{(\tau)}\right)\Big[\widetilde{V}_i^{(\tau)}\left(\mathcal{S}_n^{\boldsymbol{\sigma}_i} \cup \{n\}; \mathcal{N}_{-i}^{(\tau)}\right) \\ &\quad -\widetilde{V}_i^{(\tau)}\left(\mathcal{S}_n^{\boldsymbol{\sigma}_i}; \mathcal{N}_{-i}^{(\tau)}\right)\Big].\end{aligned}$$

Now, one can apply the algorithms in Section III to compute the Shapley value $\phi_{i,n}^{(\tau)}\left(\widetilde{V}_i^{(\tau)}\right)$ for each coalition $\mathcal{N}_i^{(\tau)}$.

Denote $\boldsymbol{W}^{(\tau)} \triangleq [w_{mn}^{(\tau)} | m, n \in \mathcal{N}]$ as the pair-wise cooperation preference profile in time slot τ. Let $\mu_i^{(\tau)}(\cdot|\boldsymbol{W}^{(\tau)})$ denote the probability distribution over $\Omega_i^{(\tau)}$ associated with the pari-wise cooperation preference profile $\boldsymbol{W}^{(\tau)}$. Then, $\phi_{i,n}^{(\tau)}(\widetilde{V}_i^{(\tau)})$ can be expressed as

$$\begin{aligned}\phi_{i,n}^{(\tau)}\left(\widetilde{V}_i^{(\tau)}\right) = \sum_{\boldsymbol{\sigma}_i \in \Omega_i^{(\tau)}} \mu_i^{(\tau)}\left(\boldsymbol{\sigma}_i \middle| \boldsymbol{W}^{(\tau)}\right)\Big[\widetilde{V}_i^{(\tau)}\left(\mathcal{S}_n^{\boldsymbol{\sigma}_i} \cup \{n\}; \mathcal{N}_{-i}^{(\tau)}\right) \\ -\widetilde{V}_i^{(\tau)}\left(\mathcal{S}_n^{\boldsymbol{\sigma}_i}; \mathcal{N}_{-i}^{(\tau)}\right)\Big].\end{aligned}$$

Again, one can apply the algorithms in Section III to compute the Shapley value $\phi_{i,n}^{(\tau)}\left(\widetilde{V}_i^{(\tau)}\right)$ for each coalition $\mathcal{N}_i^{(\tau)}$.

V. Applications

To show the versatility of our models and algorithms, we apply them to: (1) divide the revenue among ISPs in deploying new Internet architectures and, (2) divide the reward among workers in crowdsourcing applications.

A. ISP Settlement

Problem description. Revenue devision is critical for the deployability of new Internet architectures [11]. Consider dividing the revenue gain of deploying a new Internet architecture, e.g., IPv6, named data networking (NDN), differential service (DiffServ), etc., among a set $\mathcal{N}$ of ISPs. We consider a class of new architectures that require full participation of all ISPs. Such architectures include named data networking (NDN) and differential service (DiffServ). Namely, the functionality of the architecture on the routing path (composed of ISPs $\mathcal{N}$) can be enabled if and only if all ISPs in $\mathcal{N}$ participate. Formally, we use the following payoff function to capture this full participation property:

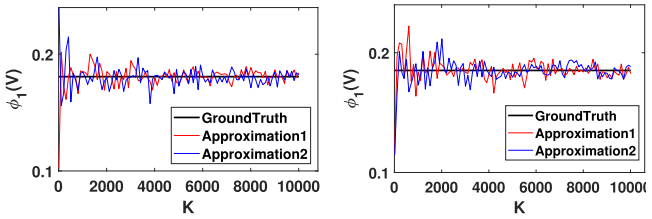
$$V(\mathcal{C}) = \begin{cases} 1, & \text{if } \mathcal{C} = \mathcal{N}, \\ 0, & \text{otherwise,} \end{cases}$$

where we normalize the revenue gain to be 1. The competition or complementarity among ISPs' businesses influence ISPs' cooperation preferences. Namely, two ISPs are more (or less) willing to cooperate if their businesses are mutually complement (or compete). *How to divide the revenue gain taking this preference into account?*

Apply our group-wise preference model. For simplicity, we consider two business types denoted by $\{-1, 1\}$. Let $b_n \in \{-1, 1\}$ denote the business type of ISP $n \in \mathcal{N}$. Two ISPs $m, n \in \mathcal{N}$ are mutually competitive if and only if they operate the same type of business, $b_m = b_n$, otherwise, they are mutually complement, i.e, $b_m \neq b_n$. We use our group-wise preference model to capture the competition or complementarity among the ISPs $\mathcal{N}$. We use the following rating model to capture that an ISP is more preferable to join a coalition whose businesses are complement to hers

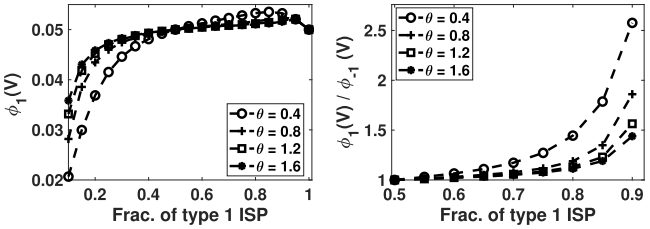
$$R_n(\mathcal{C}) = 1 + \frac{M-1}{1 + \exp(-\theta \sum_{m \in \mathcal{C}} |b_n - b_m|)},$$

where θ models the sensitivity to the business type. A larger θ models that the cooperation preference is more sensitive to the business type. Then, we apply the stochastic process developed in Section II-B and consider the conditional joining probability derived in Equation (5).



(a) $\theta = 0.4$ (b) $\theta = 1.6$

Fig. 3. Impact of number of samples on the accuracy of Algo. 1 and 2.



(a) Revenue share (b) Revenue share ratio

Fig. 4. Revenue share and revenue share ratio of deploying a new Internet architecture.

Numerical studies and implications: We combine Algorithm 1 and 2 to compute the Shapley value (i.e., revenue share) for each ISP, setting $K = 10^9$. Due to symmetry, each ISP of the same type has the same revenue share or Shapley value. Thus, we denote the revenue share for a type 1 and -1 ISP as $\phi_1(V)$ and $\phi_{-1}(V)$ respectively.

To evaluate the performance of our proposed algorithms, we first consider a small coalition formed by $N = 5$ ISPs, where the exact Shapley value can be calculated via exhaustive search. Figure 3 shows the impact of number of samples on the accuracy of the estimation produced by combining Algo. 1 and 2. The horizontal axis show the number of samples K and the vertical axis shows the estimated Shapley value $\phi_1(V)$ produced by combining Algo. 1 and 2. The red curve and blue curve correspond to two approximation paths. From Figure 3, one can observe that the estimated $\phi_1(V)$ converges to the ground truth. When the number of samples is more than 2000, the estimated $\phi_1(V)$ is very close to the ground truth. This shows that the combination of Algo. 1 and 2 has superior empirical performance.

We consider a larger coalition of $N = 20$ ISPs for illustration purpose. There can be many ISPs and they can form many coalitions, where our method can be applied to handle each coalition individually. Note that $20! \approx 2 \times 10^{18}$, implying a high computational complexity for computing the exact Shapley value. Note that under the classical Shapley value, each ISP equally shares the revenue, i.e., each one gets $\phi_1(V) = \phi_{-1}(V) = 1/20 = 0.05$. Figure 6(a) shows the revenue share of a type 1 ISP $\phi_1(V)$, where the fraction of type 1 ISP varies from 10% to 100%. Figure 6(c) shows the revenue ratio, i.e., the revenue share of a type 1 ISP divided by that of a type -1 ISP $\phi_1(V)/\phi_{-1}(V)$, where the fraction of type 1 ISPs varies from 50% to 90%. One can observe that when the fraction of type 1 ISP is less than 50%, $\phi_1(V)$ is smaller than $\phi_{-1}(V)$, otherwise $\phi_1(V)$ is larger of equal to $\phi_{-1}(V)$. This implies that each minority ISP (i.e., fraction of type 1 ISPs is less than 50%) shares less revenue than each majority ISP (i.e., fraction of type 1 ISPs is more than 50%). The reason is that for the majority ISPs, they are mutually competitive to each other, and thus more revenue share is needed to be distributed to them. As the fraction of type 1 ISP increases from 50% to 100%, the revenue share for each type 1 ISP $\phi_1(V)$ first increases and then drops to 0.05 (when all ISPs are of type 1, they equally share the revenue gain $1/20 = 0.05$). As the fraction of type 1 ISP increases from 50% to 90%, the revenue ratio $\phi_1(V)/\phi_{-1}(V)$ increases from 1 to as high as 2.5. Thus, when there is a small fraction, e.g., 10%, of type -1 ISPs, the revenue share for a type -1 ISP (i.e., minority ISP) is significantly smaller than that of a type 1 ISP (i.e., majority ISP). The insight is that when the minority player can obtain such a small revenue share as compare to the majority player, the minority player will be less willing to participate, and this makes the deployment of the new protocol (requiring full participation of all ISPs) extraordinarily difficult. When half of the ISPs are types 1 and the other half are type -1, each ISP equally share the revenue, making it less difficult for all the ISPs to deploy the new architecture. Note that these observations complement the results of [11] on the deployability of new architectures, which does not consider cooperation preferences among ISPs.

B. Crowdsourcing

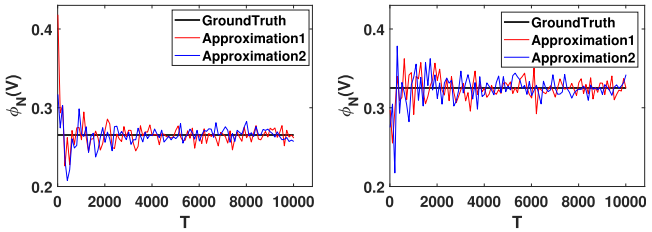
Problem description. Dividing reward is a crucial problem in crowdsourcing applications [12], [36]. Consider dividing the reward among a set $\mathcal{N}$ of workers, who work on a crowdsourcing task. We use an unweighted undirected graph to represent the social relationships among workers. In particular, workers are more willing to cooperate with friends. The cooperation of more workers lead to higher quality solutions, resulting in larger reward. We use the following reward functions to capture this positive externality:

$$V(\mathcal{C}) = (|\mathcal{C}|/|\mathcal{N}|)^{1-\eta}, \quad \forall \mathcal{C} \subseteq \mathcal{N},$$

where $\eta \in [0, 1]$ captures the diminishing return effect. A larger η models a larger strength of diminishing return effect. The question is: *How to divide the reward taking the social network into account?*

Apply our pair-wise preference model. We apply our pair-wise preference model developed in Section II-B to incorporate the social network. In our pair-wise preference model, we set $w_{mn} = 0$ if there is no link between m and n in the social network and $w_{mn} = w_{nm} = 1$ if there is a link. We also use Equation (7) and (8) to incorporate the social network into the Shapley value.

Numerical studies and implications. To illustrate the impact of the social network on reward division, we consider a "star network". In particular, we consider N workers and the worker N lies in the center, i.e., $w_{Nm} = w_{mN} = 1, \forall m \in \{1, \ldots, N-1\}$ and all other weights are 0. The worker N is also known as the social hub. Choosing a star network allow readers to verify the result and to better deliver the connection



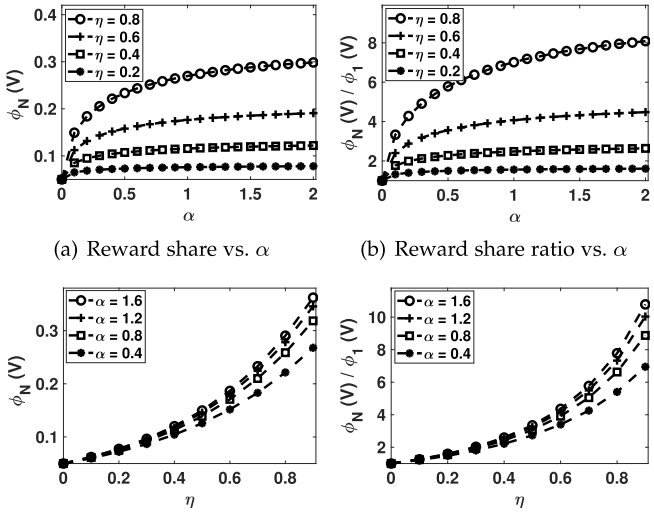
(a) $\alpha = 0.4$ (b) $\alpha = 1.6$

Fig. 5. Impact of number of samples on the accuracy of Algorithm 6.

between network structure (or cooperation preference) and payoff share. Our algorithms can definitely be applied to *general and arbitrary network topologies*. By symmetry, all other $N-1$ workers have the same Shapley value. We thus focus on the Shapley value of $\phi_N(V)$ and $\phi_1(V)$. We apply Algorithm 6 to compute the Shapley value by setting $T = 10^9$.

To evaluate the performance of Algorithm 6, we first consider a small coalition formed by $N = 5$ workers, where the exact Shapley value can be calculated via exhaustive search. Figure 5 shows the impact of number of samples on the accuracy of Algorithm 6, where the horizontal axis shows the number of samples T and the vertical axis shows the estimated Shapley value $\phi_N(V)$ by Algorithm 6. The red curve and blue curve correspond to two approximation paths. From Figure 5, one can observe that the estimated $\phi_N(V)$ converges to the ground truth. When the number of samples is more than 3000, the estimated $\phi_N(V)$ is very close to the ground truth. This shows that Algorithm 6 has superior empirical performance.

We consider a larger coalition formed by $N = 20$ workers in a crowdsourcing system. Note that we do not assume a crowdsourcing system with only $N = 20$ workers. There can be many workers and they can form many coalitions, where our method can be applied to handle each coalition individually. Figure 6(a) shows that $\phi_N(V)$ increases in the preference sensitivity parameter α. This implies that the social hub N shares more reward when workers' cooperation preferences become more sensitive to the social relationship. Figure 6(c) shows that $\phi_N(V)$ increases in η. This implies that the social hub N receives more reward when the strength of diminishing return effect η increases. Figure 6(b) and 6(d) show that the reward share ratio, i.e., $\phi_N(V)/\phi_1(V)$ increases in both the preference sensitivity parameter α and strength of diminishing return effect η. Furthermore, the reward ratio $\phi_N(V)/\phi_1(V)$ is larger than 1. This implies that the social hub receives more reward than each of the other 19 workers and each of the other 19 workers receives less and less reward as α or η increases. This also demonstrates the important role of worker N, who is a "hub" in the preference graph. Note that the reward share for a worker needs to exceed the cost of this worker in solving the task, so that this worker has incentive to participate in the crowdsourcing task. Thus, when the preference sensitivity parameter α or the strength of diminishing return effect η increases, it would be more difficult for all workers to cooperate. Note that these observations complement the results of [12], which does not consider cooperation preferences in reward division.



(a) Reward share vs. α (b) Reward share ratio vs. α

(c) Reward share vs. η (d) Reward share ratio vs. η

Fig. 6. Reward division in crowdsourcing systems.

VI. Related Work

There are several solution concepts for cooperative game [18], for example, Shapley value [1], Banzhaf index [37], etc. The Shapley value has been studied extensively and our work is closely related to two lines of them: (1) cooperation preference modeling; (2) sampling algorithms design for computing Shapley value.

Cooperation preference modeling. A variety of works proposed restricted cooperation models to quantify cooperation preferences [38], [39], [40]. This line of works used a graph among players to filter out some permutations of players (i.e., restrict the coalition to the remaining permutations), and each one of the remaining permutations happens with equal probability. However, for our model, we do not filter out any permutation, and we use a preference graph to adjust the probability mass over each permutation. Stern et al. [41] proposed a hypercube graph to capture the cooperation preference. In their hypercube graph, each node corresponds to a subset of players (i.e., representing a coalition) and each directed link represents a user joining a coalition. The weights of a link can be interpreted as the willingness to join a coalition. The key difference to our work is that in our model, each node corresponds to a player and we use a directed link to capture the willingness of a player to cooperate with the other one. Also our graph is not constrained to be a hypercube. Few works modeled the cooperation preference from a probabilistic perspective, i.e., generalizing the distribution μ over orderings Ω. Von Hohenbalken and Levesque [19] used a simplicial distribution to generate a non-uniform distribution μ over orderings Ω, i.e., mapping each sample of the simplicial distribution to an ordering of players. Kalai and Samet [20] proposed to use the weights of players to generate a non-uniform distribution μ over orderings Ω. Their models associate each player with a weight and the ordering is biased toward the weight. Balkanski et al. [30] and Feldotto et al. [31] considered the data dependent Shapley

value. They considered non-uniform distribution μ over orderings Ω and assumed that IID samples from μ are available. We formulate a group-wise model and a pair-wise model to solicit cooperation preferences and develop mathematical models to characterize how they influence the formation of coalition. Furthermore, we address a number of challenges, i.e., how to store these resource demanding preferences, and obtain computationally expensive IID samples from μ. Lastly, our model is more flexible and extends to multiple coalitions and dynamic cooperation preferences.

Sampling algorithms design for Shapley value. A number of works used the Monte Carlo sampling to estimate the Shapley value. Mann and Shapley [42] were the first to apply the Monte Carlo sampling to estimate the classical Shapley values, but without any theoretical guarantees. Bachrach et al. [43] derived the upper and lower bound on the sample complexity of estimating the classical Shapley value via simple Monte Carlo sampling. Liben et al. [44], Aziz et al. [45] and Balkanski et al. [30] refined the sample complexity analysis via restricting the classical Shapley value for supermodular coalition games, submodular games with bounded curvature and matching games respectively. To improve the simple Monte Carlo sampling method in estimating the classical Shapley value, Castro et al. [46] proposed an algorithm based on the generalization of order importance sampling, and O'Brien et al. [47], [48] proposed a stratified sampling algorithm. A number of works [19], [20], [30], and [31] also studied Monte Carlo sampling for generalized Shapley value, where μ is a non-uniform distribution over Ω. Their works were built on the assumption that IID samples from μ are inexpensive to obtain. Different from these works, we focus on modeling the cooperation preferences and incorporating them into the Shapley value. Under our preference model, IID samples from μ can be computationally expensive to obtain, which are inexpensive for the classical Shapley value and the previous generalized Shapley value models. More importantly, we also design efficient algorithms with theoretical grantees to compute its values. Lastly, our algorithms are more flexible and extends to multiple coalitions and dynamic cooperation preferences.

VII. Conclusion

This paper generalizes the classical Shapley value to capture cooperation preferences. We develop mathematical models to represent two types of cooperation preferences, i.e., *group-wise preferences* and *pair-wise preferences*. We generalize the classical Shapley value to incorporate theses preferences. We also show that it is computationally expensive to evaluate the cooperation preference aware Shapley value. We design efficient randomized algorithms (with theoretical guarantees) to compute the cooperation preference aware Shapley value. We also extend our models and algorithms to divide payoff for multiple coalitions with dynamic preferences. Finally, we demonstrate the applications of our framework by applying it to divide the revenue among ISPs in deploying new Internet architectures, as well as divide the reward among workers in a crowdsourcing system. We also provide important insights on how cooperation preferences influence the reward division.

One limitation of our work is that the sample complexity of some algorithms are computationally expensive to evaluate, making it difficult to determine the sample size in advance. Another limitation is that it is difficult to apply or extend our model to other concepts in coalitional games, such as cores and voting theory. The reason is that cooperation preferences are captured in the likelihood of ordering of the Shapley value and these concepts do not have such probabilistic aspect.

References

[1] L. S. Shapley, "A value for N-person games," *Contrib. Theory Games*, vol. 2, no. 28, pp. 307–317, Jun. 1953.

[2] R. T. B. Ma, D. M. Chiu, J. Lui, V. Misra, and D. Rubenstein, "Internet economics: The use of Shapley value for ISP settlement," *IEEE/ACM Trans. Netw.*, vol. 18, no. 3, pp. 775–787, Jun. 2010.

[3] V. Misra, S. Ioannidis, A. Chaintreau, and L. Massoulié, "Incentivizing peer-assisted services: A fluid Shapley value approach," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 38, no. 1, pp. 215–226, Jun. 2010.

[4] F. Zafari, J. Li, K. K. Leung, D. Towsley, and A. Swami, "A game-theoretic approach to multi-objective resource sharing and allocation in mobile edge," in *Proc. Technol. Wireless Edge Workshop*, Oct. 2018, pp. 9–13.

[5] O. Ben-Porat and M. Tennenholtz, "A game-theoretic approach to recommendation systems with strategic content providers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2018, pp. 1118–1128.

[6] M. Chessa and P. Loiseau, "A cooperative game-theoretic approach to quantify the value of personal data in networks," in *Proc. 12th workshop Econ. Netw., Syst. Comput.*, Jun. 2017, p. 9.

[7] J. Kleinberg, C. H. Papadimitriou, and P. Raghavan, "On the value of private information," in *Proc. 8th Conf. Theor. Aspects Rationality Knowl.* Burlington, MA, USA: Morgan Kaufmann, 2001, pp. 249–257.

[8] S. Cohen, E. Ruppin, and G. Dror, "Feature selection based on the Shapley value," in *Proc. Int. Joint Conf. Artif. Intell.*, vol. 5, 2005, pp. 665–670.

[9] S. Cohen, G. Dror, and E. Ruppin, "Feature selection via coalitional game theory," *Neural Comput.*, vol. 19, no. 7, pp. 1939–1961, Jul. 2007.

[10] E. Strumbelj and I. Kononenko, "An efficient explanation of individual classifications using game theory," *J. Mach. Learn. Res.*, vol. 11, pp. 1–18, Jan. 2010.

[11] Y. Li, H. Xie, and J. C. S. Lui, "Quantifying deployability evolvability of future internet architectures via economic models," in *Proc. IEEE 26th Int. Conf. Work Protocols (ICNP)*, Sep. 2018, pp. 324–334.

[12] S. Yang, F. Wu, S. Tang, X. Gao, B. Yang, and G. Chen, "On designing data quality-aware truth estimation and surplus sharing method for mobile crowdsensing," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 4, pp. 832–847, Apr. 2017.

[13] M. del Pozo, C. Manuel, E. González-Arangüena, and G. Owen, "Centrality in directed social networks. A game theoretic approach," *Social Netw.*, vol. 33, no. 3, pp. 191–200, Jul. 2011.

[14] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowl. Inf. Syst.*, vol. 41, no. 3, pp. 647–665, Dec. 2014.

[15] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. vol. 30, 2017, pp. 4765–4774.

[16] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2016, pp. 598–617.

[17] J. Sliwinski, M. Strobel, and Y. Zick, "Axiomatic characterization of data-driven influence measures for classification," 2017, *arXiv:1708.02153*.

[18] G. Chalkiadakis, E. Elkind, and M. Wooldridge, "Computational aspects of cooperative game theory," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 5, no. 6, pp. 1–168, Oct. 2011.

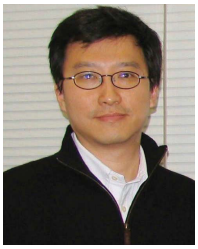
[19] B. Von Hohenbalken and T. Levesque, "Generalized Shapley values by simplicial sampling," Laxenburg, Austria, IIASA Work. Paper WP-78-50, 1978.

[20] E. Kalai and D. Samet, "On weighted Shapley values," *Int. J. Game Theory*, vol. 16, no. 3, pp. 205–222, Sep. 1987.

[21] G. Owen, "Political games," *Nav. Res. Logistics Quart.*, vol. 18, no. 3, pp. 345–355, 1971.

[22] S. S. Hamlen, W. A. Hamlen, Jr., and J. Tschirhart, "The use of the generalized Shapley allocation in joint cost allocation," *Accounting Rev.*, vol. 55, no. 2, pp. 269–287, 1980.
[23] D. M. Kilgour, "A Shapley value for cooperative games with quarrelling," in *Game Theory as a Theory of a Conflict Resolution*. New York, NY, USA: Springer, 1974, pp. 193–206.
[24] X. Deng and C. H. Papadimitriou, "On the complexity of cooperative solution concepts," *Math. Oper. Res.*, vol. 19, no. 2, pp. 257–266, May 1994.
[25] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.
[26] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*. Providence, RI, USA: American Mathematical Society, 2006.
[27] C. Robert and G. Casella, *Monte Carlo Statistical Methods*. Berlin, Germany: Springer Verlag, 2004.
[28] J. Matoušek and J. Vondrák, "The probabilistic method," Lecture Notes, Dept. Appl. Math., Charles Univ., Prague, Czechia, Lect. Notes, 2001.
[29] S. Boucheron, G. Lugosi, and P. Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford, U.K.: Oxford Univ. Press, 2013.
[30] E. Balkanski, U. Syed, and S. Vassilvitskii, "Statistical cost sharing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6221–6230.
[31] M. Feldotto, C.-J. Haake, A. Skopalik, and N. Stroh-Maraun, "Disaggregating user evaluations using the Shapley value," in *Proc. NetEcon*, 2018, pp. 1–6.
[32] R. J. Aumann and S. Hart, *Handbook of Game Theory With Economic Applications*, vol. 1. Amsterdam, The Netherlands: Elsevier, 1992.
[33] S. Hart and M. Kurz, "Endogenous formation of coalitions," *Econometrica, J. Econ. Soc.*, vol. 51, no. 4, pp. 1047–1064, 1983.
[34] J. P. Kahan and A. Rapoport, *Theories of Coalition Formation*. London, U.K.: Psychology Press, 2014.
[35] D. Ray and R. Vohra, "Coalition formation," in *Handbook of Game Theory With Economic Applications*, vol. 4. Amsterdam, The Netherlands: Elsevier, 2015, pp. 239–326.
[36] H. Xie, J. C. S. Lui, and D. Towsley, "Design and analysis of incentive and reputation mechanisms for online crowdsourcing systems," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 1, no. 3, p. 13, May 2016.
[37] J. F. Banzhaf III, "Weighted voting doesn't work: A mathematical analysis," *Rutgers Law Rev.*, vol. 19, no. 1, pp. 317–343, 1964.
[38] J. M. Alonso-Meijide and M. G. Fiestras-Janeiro, "The banzhaf value and communication situations," *Nav. Res. Logistics*, vol. 53, no. 3, pp. 198–203, 2006.
[39] J. M. Alonso-Meijide, M. Álvarez-Mozos, and M. G. Fiestras-Janeiro, "Values of games with graph restricted communication and a priori unions," *Math. Social Sci.*, vol. 58, no. 2, pp. 202–213, Sep. 2009.
[40] R. B. Myerson, "Graphs and cooperation in games," *Math. Oper. Res.*, vol. 2, no. 3, pp. 225–229, 1977. [Online]. Available: http://www.jstor.org/stable/3689511
[41] A. Stern and A. Tettenhorst, "Hodge decomposition and the Shapley value of a cooperative game," *Games Econ. Behav.*, vol. 113, pp. 186–198, Jan. 2019.
[42] I. Mann and L. S. Shapley, "Values of large games. 6: Evaluating the electoral college exactly," RAND Corp., Santa Monica, CA, USA, Tech. Rep. RM-3158-PR, 1962.
[43] Y. Bachrach, E. Markakis, E. Resnick, A. D. Procaccia, J. S. Rosenschein, and A. Saberi, "Approximating power indices: Theoretical and empirical analysis," *Auto. Agents Multi-Agent Syst.*, vol. 20, no. 2, pp. 105–122, Mar. 2010.
[44] D. Liben-Nowell, A. Sharp, T. Wexler, and K. Woods, "Computing Shapley value in supermodular coalitional games," in *Computing and Combinatorics*, J. Gudmundsson, J. Mestre, and T. Viglas, Eds. Berlin, Germany: Springer, 2012, pp. 568–579.
[45] H. Aziz and B. de Keijzer, "Shapley meets Shapley," in *Proc. STACS*, 2014, pp. 1–12.
[46] J. Castro, D. Gömez, and J. Tejada, "Polynomial calculation of the Shapley value based on sampling," *J. Comput. Oper. Res.*, vol. 36, no. 5, pp. 1726–1730, 2009.
[47] G. O'Brien, A. El Gamal, and R. Rajagopal, "Shapley value estimation for compensation of participants in demand response programs," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2837–2844, Nov. 2015.
[48] S. Maleki, "Addressing the computational issues of the Shapley value with applications in the smart grid," Ph.D. dissertation, Phys. Sci. Eng., Univ. Southampton, Southampton, U.K., 2015.

Hong Xie (Member, IEEE) received the B.Eng. degree from the School of Computer Science and Technology, University of Science and Technology of China, in 2010, and the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong (CUHK), in 2015, proudly under the supervision of Prof. John C. S. Lui. He was a Post-Doctoral Research Fellow at the Department of Computing Science and Engineering, CUHK, hosted by Prof. John C. S. Lui, and a Post-Doctoral Research Fellow at the School of Computing, National University of Singapore, hosted by Prof. Richard T. B. Ma. He is a member of CCF and ACM.



John C. S. Lui (Fellow, IEEE) received the Ph.D. degree in computer science from the University of California at Los Angeles, Los Angeles, CA, USA. He is currently the Choh-Ming Li Chair Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. His current research interests include machine learning, online learning (e.g., multi-armed bandit and reinforcement learning), network science, future internet architectures and protocols, network economics, network/system security, and large scale storage systems. He is an Elected Member of the IFIP WG 7.3, a Fellow of ACM, and a Senior Research Fellow of the Croucher Foundation. From 2011 to 2015, he was the Past Chair of the ACM SIGMETRICS. He received various departmental teaching awards and the CUHK Vice-Chancellor's Exemplary Teaching Award. He was a co-recipient of the Best Paper Award in the IFIP WG 7.3 Performance 2005, IEEE/IFIP NOMS 2006, SIMPLEX 2013, and ACM RecSys 2017.