Coreness of Cooperative Games with Truncated Submodular Profit Functions*

Wei Chen[1], Xiaohan Shan[2,3], Xiaoming Sun[2,3], and JialinZhang[2,3]

[1] Microsoft, Beijing, China
[2] CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[3] University of Chinese Academy of Sciences, Beijing, China

Abstract. *Coreness* represents solution concepts related to core in cooperative games, which captures the stability of players. Motivated by the scale effect in social networks, economics and other scenario, we study the coreness of cooperative game with truncated submodular profit functions. Specifically, the profit function $f(\cdot)$ is defined by a truncation of a submodular function $\sigma(\cdot)$: $f(\cdot) = \sigma(\cdot)$ if $\sigma(\cdot) \geq \eta$ and $f(\cdot) = 0$ otherwise, where η is a given threshold. In this paper, we study the core and three core-related concepts of truncated submodular profit cooperative game. We first prove that whether core is empty can be decided in polynomial time and an allocation in core also can be found in polynomial time when core is not empty. When core is empty, we show hardness results and approximation algorithms for computing other core-related concepts including *relative* least-core value, *absolute* least-core value and least *average dissatisfaction* value.

1 Introduction

With the wide popularity of social media and social network sites such as Facebook, Twitter, WeChat, etc., social networks have become a powerful platform for spreading information among individuals. Thus, influential users always play important role in a social network. Motivated by this background, influence diffusion in social networks has been extensively studied [9,15,3]. Most of previous works focus on exploring influential nodes. To the best of our knowledge, there is no study about the "stability" of influential nodes (seed set) when they are treated as a coalition.

Consider the following scenario. A group of influential people in a social network are considering forming a coalition so that they can better serve many advertisers through viral marketing in the social network. To make the coalition stable, we need to design a fair profit allocation scheme among the members of the coalition, such that no individual or a subset of people have incentive to

* This work is supported in part by the National Natural Science Foundation of China Grant 61433014, 61502449, 61602440, the 973 Program of China Grants No. 2016YFB1000201.

deviate from this coalition, thinking that the allocation to them is unfair and they could earn more by the deviation and forming an alliance by themselves. A useful and mature framework of studying such incentives for stable coalition formation is the cooperative game theory, and in particular the coreness (core and its related concepts) of the cooperative games [7,17].

In the above social influence scenario, the typical way of measuring the contribution of any set S of influential people is by its influence spread function $\sigma(S)$, which measures the expected number of people in the social network that could be influenced by S under some stochastic diffusion model. Extensive researches have been done on stochastic diffusion models, and it has been shown that under a large class of models $\sigma(S)$ is both monotone and submodular[4] [15,18,3]. However, the advertisers would only be interested in the coalition as a viral marketing platform when the influence spread reaches certain scale level. In other words, the coalition can only receive profit after the influence spread is above a certain scale threshold η. Therefore, the true profit function for the coalition is $f(S) = \sigma(S)$ when $\sigma(S) \geq \eta$, and $f(S) = 0$ otherwise. We call such f truncated submodular functions. This motivate us to study the coreness of the cooperative games with truncated submodular profit functions.

Both submodularity and scale effect are common in economic behaviors beyond the above example of viral marketing in social networks. Therefore, considering truncated submodular functions as the profit functions is reasonable. In this paper, we study the computational issues related to the coreness of cooperative games with truncated submodular profit functions.

Solution Concepts in Cooperative Games. A cooperative game $\mathit{\Gamma} = (V, \gamma)$ consists of a player set $V = \{1, 2, \cdots, n\}$ and a profit function $\gamma : 2^V \to \mathbb{R}$ with $\gamma(\emptyset) = 0$. A subset of players $S \subseteq V$ is called a *coalition* and V is called the *grand coalition*. For each coalition S, $\gamma(S)$ represents the profit obtained by S without help of other players. An allocation over the players is denoted by a vector $x = (x_1, x_2, \cdots, x_n) \in \mathbb{R}^n$ whose components are one-to-one associated with players in V, where $x_i \in \mathbb{R}$ is the value received by player $i \in V$ under allocation x. For any player set $S \subseteq V$, we use the shorthand notation $x(S) = \sum_{i \in S} x_i$. A set of all allocations satisfying some specific requirements is called a *solution concept*.

The *core* [11,21] is one of the earliest and most attractive solution concepts that directly addresses the issue of stability. The core of a game is the set of allocations ensuring that no coalition would have an incentive to split from the grand coalition, and do better on its own. More precisely, the core of a game $\mathit{\Gamma}$ (denoted by $\mathcal{C}(\mathit{\Gamma})$), is the following set of allocations: $\mathcal{C}(\mathit{\Gamma})=\{x \in \mathbb{R}^n : x(V) = \gamma(V), x(S) \geq \gamma(S),\ \forall\ S \subseteq V\}$. Intuitively, the requirement of $x(S) \geq \gamma(S)$ means that the coalition S receives profit allocation $x(S)$ that is at least their profit contribution $\gamma(S)$, so they would prefer to stay with the grand coalition. In practice, core is very strict and may be even empty in some cases. When $\mathcal{C}(\mathit{\Gamma})$ is empty, there must be some coalition becoming dissatisfied since they can obtain

[4] A set function f is monotone if $f(S) \leq f(T)$ for all $S \subseteq T$, and is submodular if $f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T)$ for all $S \subseteq T$ and $u \notin T$.

more benefits if they leave the grand coalition and work as a separated team. In this case, we use the dissatisfaction degree (or dissatisfaction value), defined as $dv(S,x) = \max\{\gamma(S) - x(S), 0\}$, to capture the instability of player set S with respect to the allocation x. Then, the overall stability of the game can be measured as either the worst-case or average-case dissatisfaction degree, for which we consider the following three versions.

The first one is the *relative least-core value* ($\mathcal{RLCV}$) [10], which reflects the relative stability, i.e. the minimum value of the maximum proportional difference between the profits and the payoffs among all coalitions.

Definition 1. *Given a cooperative game Γ, the* relative least-core value *of Γ ($\mathcal{RLCV}(\Gamma)$) is* $\min_x \max_S \frac{dv(S,x)}{\gamma(S)}$. *Technically, $\mathcal{RLCV}(\Gamma)$ is the optimal solution of the following linear programming:*

$$\begin{aligned} \min \ & r \\ s.t. \ & \begin{cases} x(V) = \gamma(V) \\ x(S) \geq (1-r)\gamma(S) & \forall\ S \subseteq V \\ x(\{i\}) \geq 0 & \forall\ i \in V \end{cases} \end{aligned} \tag{1}$$

The second one is the *absolute least-core value* ($\mathcal{ALCV}$) [16], which reflects the absolute stability, i.e. the minimum value of the maximum difference between the profits and the payoffs among all coalitions. The formal definition is as following.

Definition 2. *Given a cooperative game Γ, the* absolute least-core value *of Γ ($\mathcal{ALCV}(\Gamma)$) is* $\min_x \max_S dv(S,x)$. *Technically, $\mathcal{ALCV}(\Gamma)$ is the optimal solution of the following linear programming:*

$$\begin{aligned} \min \ & \varepsilon \\ s.t. \ & \begin{cases} x(V) = \gamma(V) \\ x(S) \geq \gamma(S) - \varepsilon & \forall\ S \subseteq V \\ x(\{i\}) \geq 0 & \forall\ i \in V \end{cases} \end{aligned} \tag{2}$$

The above two classical least-core values capture the stability from the perspective of the most dissatisfied coalition i.e. the worst-case stability. Sometimes the worst case is too extreme to reflect the real stability. Thus, we introduce the *least average dissatisfaction value* ($\mathcal{LADV}$), which reflects the minimum value of average dissatisfaction degree among all coalitions.

Definition 3. *Given a cooperative game Γ, the* least average dissatisfaction value *of Γ ($\mathcal{LADV}(\Gamma)$) is* $\min_x \mathbb{E}_S(dv(S,x))$. *Technically, $\mathcal{LADV}(\Gamma)$ is the optimal value of the following linear programming:*

$$\begin{aligned} \min \ & \tfrac{1}{2^n} \textstyle\sum_{S \subseteq V} \max\{\gamma(S) - x(S), 0\} \\ s.t. \ & \begin{cases} x(V) = \gamma(V) \\ x(\{i\}) \geq 0 & \forall\ i \in V \end{cases} \end{aligned} \tag{3}$$

In this paper, we consider the following computational problems in the context of truncated submodular functions: (a) Whether the core of a given cooperative game is empty? (b) How to find an allocation in core if the core is not empty? (c) If the core is empty, how to compute the relative least-core value, the absolute least-core value and the least average dissatisfaction value of a cooperative game?

Contributions. We study the coreness of truncated submodular profit cooperative game Γ_f. We consider computational properties of the core, the relative least-core value, the absolute least-core value and the least average dissatisfaction value of Γ_f, which are denoted by $\mathcal{C}(\Gamma_f)$, $\mathcal{RLCV}(\Gamma_f)$, $\mathcal{ALCV}(\Gamma_f)$ and $\mathcal{LADV}(\Gamma_f)$, respectively.

We first prove that checking the non-emptiness of $\mathcal{C}(\Gamma_f)$ can be done in polynomial time. Moreover, we can find an allocation in the core if the core is not empty. Next, we consider the case when the core is empty. For the problem of computing the relative least-core value ($\mathcal{RLCV}(\Gamma_f)$), we show that it is in general NP-hard, but when truncation threshold $\eta = 0$, there is a polynomial time algorithm. Along the way, we also find an interesting partial result showing that there is no polynomial time separation oracle for the $\mathcal{RLCV}(\Gamma_f)$'s linear program unless P=NP, which is of independent interest since it reveals close connections with a new class of combinatorial problems. For the absolute least-core value problem $\mathcal{ALCV}(\Gamma_f)$, we prove that finding $\mathcal{ALCV}(\Gamma_f)$ is APX-hard even when $\sigma(\cdot)$ is defined as the influence spread under the classical independent cascade (IC) model in social network. We also prove that there exists a polynomial time algorithm which can guarantee an additive term approximation. Finally, for the least average dissatisfaction value problem $\mathcal{LADV}(\Gamma_f)$, we show that we can use the stochastic gradient descent algorithm to compute $\mathcal{LADV}(\Gamma_f)$ to an arbitrary small additive error.

Related Work. Cooperative game theory is a branch of (micro-)economics that studies the behavior of self-interested agents in strategic settings where binding agreements between agents are possible [2]. Numerous classical studies about cooperative game provide rich mathematical framework to solve issues related to cooperation in multi-agent systems [8,6]. Schulz and Uhan study the approximation of the absolute least core value of supermodular cost cooperative games [19], the results of which can be generalized to submodular profit cooperative games. An important application of our study is to analyze the stability of influential people in social networks. Almost all the existing studies focus on selecting the seed set [5,12,22]. To the best of our knowledge, there is no study considering the stability of the selected seed set. We utilize cooperative game theory to analyse the stability of seed set, and generalize it to a generic cooperative game with truncated submodular functions. The truncated operation represents the "threshold effect" which has been studied widely in literature [13,1].

2 Model and Problems

2.1 Cooperative Games with Truncated Submodular Profit Functions

A truncated submodular profit cooperative game is denoted by $\Gamma_f = (V, f(\cdot))$. In Γ_f, V is the player set and $f(\cdot)$ is the profit function which is defined as follows:

$$f(S) = \begin{cases} \sigma(S), & \text{if } \sigma(S) \geq \eta \\ 0, & \text{if } \sigma(S) < \eta \end{cases}$$

Note that $\sigma(\cdot)$ is a nonnegative monotone increasing submodular function with $\sigma(\emptyset) = 0$ and $0 \leq \eta \leq \sigma(V)$ is a nonnegative threshold. To make it explicit, henceforth, a truncated submodular profit cooperative game is denoted by a triple $(V, \sigma(\cdot), \eta)$. Note that the explicit representation of $\sigma(\cdot)$ might be exponential in the size of V. The standard way to bypass this difficulty is to assume that $\sigma(\cdot)$ is given as a value oracle.

2.2 Computational Problems on the Coreness

Given a truncated submodular profit cooperative game Γ_f, we focus on the following problems:
CORE: Is $\mathcal{C}(\Gamma_f) \neq \emptyset$ and how to find an allocation in $\mathcal{C}(\Gamma_f)$ when $\mathcal{C}(\Gamma_f) \neq \emptyset$?
ALCV: When $\mathcal{C}(\Gamma_f) = \emptyset$, how to compute $\mathcal{ALCV}(\Gamma_f)$?
RLCV: When $\mathcal{C}(\Gamma_f) = \emptyset$, how to compute $\mathcal{RLCV}(\Gamma_f)$?
LADV: When $\mathcal{C}(\Gamma_f) = \emptyset$, how to compute $\mathcal{LADV}(\Gamma_f)$?

Before we analyze the above problems, we introduce a specific instance of truncated submodular profit cooperative game (see Section 2.3).

2.3 Influence Cooperative Game (Γ_{inf})

As the description in our introduction, an important motivation of our model is influence in social networks. In this section, we introduce a specific instance of truncated submodular profit cooperative game, *influence cooperative game*.

Social graph. A social graph is a directed graph $G = (V \cup U, E; P)$, where $V \cup U$ is the vertex set and E is the edge set. $P = \{p_e\}_{e \in E}$ and p_e is the influence probability on each edge $e \in E$. Note that, V and U denote the vertex set of influential people and target people in G, respectively.

Influence diffusion model. The information diffusion process follows the independent cascade (IC) model proposed by [15]. In the IC model, discrete time steps $t = 0, 1, 2, \cdots$ are used to model the diffusion process. Each node in G has two states: inactive or active. At step 0, nodes in seed set S are active and other nodes are inactive. For any step $t \geq 1$, if a node u is newly active at step $t-1$, u has a single chance to influence each of its inactive out-neighbor v with independent probability p_{uv} to make v active. Once a node becomes active, it will never return to the inactive state. The diffusion process stops when there is no new active nodes at a time step. For any $S \subseteq V$, we use $\sigma^{\text{IC}}(S)$ to denote the influence spread of S, the expected number of activated nodes in U from seed set $S \subseteq V$, at the end of an IC diffusion. According to [15], $\sigma^{\text{IC}}(\cdot)$ is a monotone submodular function.

Definition 4. *An* influence cooperative game $\Gamma_{\text{inf}} = (V, \sigma^{\text{IC}}(\cdot), \eta)$ *is a special form of the truncated cooperative game, with V as the player set, and the truncation of influence spread function $\sigma^{\text{IC}}(\cdot)$ as the profit function.*

In the rest of this paper, we analyze problems defined in Section 2.2 one by one. Note that our positive results (properties and algorithms) could apply to all truncated submodular profit cooperative games including influence cooperative game. Our hardness results are established for the influence cooperative games, so it is stronger than the hardness results for general truncated submodular cooperative games.

3 Computing Core

We start by considering the core of Γ_f ($\mathcal{C}(\Gamma_f)$). In Γ_f, we say a player $i \in V$ is a *veto player* if $\sigma(S) < \eta$ for any $S \subseteq V \setminus \{i\}$. That is to say, a successful coalition must include all veto players.

Lemma 1. *$\mathcal{C}(\Gamma_f) \neq \emptyset$ if and only if:*
(i) There exists at least one veto player in Γ_f, or
(ii) $\sigma(S) = \sum_{i \in S} \sigma(\{i\})$, for any $S \subseteq V$.

Proof. Suppose the player set of Γ_f is $V = \{1, 2, \cdots, n\}$. We first prove the sufficiency of Lemma 1. On one hand, suppose i is a veto player of Γ_f, then we can find a trivial allocation x in $\mathcal{C}(\Gamma_f)$: $x(\{i\}) = \sigma(V)$ and $x(\{j\}) = 0$, $\forall\ j \in V \setminus \{i\}$. On the other hand, $x(\{i\}) = \sigma(\{i\})$ ($\forall i \in V$) is an allocation in $\mathcal{C}(\Gamma_f)$ if $\sigma(S) = \sum_{i \in S} \sigma(\{i\})$.

Now we prove the necessity. Suppose $\mathcal{C}(\Gamma_f) \neq \emptyset$ and $x \in \mathcal{C}(\Gamma_f)$. Let $\sigma(V) = \sum_{i=1}^{n} M_i$, where $M_i = \sigma(\{1, 2, \cdots, i\}) - \sigma(\{1, 2, \cdots, i-1\})$ is the marginal increasing of player i. If there is no veto player, then for any $i \in V$, $\sigma(V \setminus \{i\}) \geq \eta$ since $\sigma(S)$ is monotone. Thus, $f(V \setminus \{i\}) = \sigma(V \setminus \{i\})$, $\forall\ i \in V$. Suppose $\sigma(V \setminus \{i\}) = \sum_{j=1}^{i-1} M_j + \sum_{j=i+1}^{n} M'_{ij}$, where $M'_{ij} = \sigma(\{1, 2, \cdots, i-1, i+1, \cdots, j\}) - \sigma(\{1, 2, \cdots, i-1, i+1, \cdots, j-1\})$. Note that $M'_{ij} \geq M_j$ since $\sigma(S)$ is submodular. By the definition of the core, for any $i \in \{1, 2, \cdots, n\}$, we have: $x(V \setminus \{i\}) \geq f(V \setminus \{i\}) = \sigma(\{V \setminus \{i\}\})$. That is, $x(V) - x(\{i\}) \geq \sum_{j=1}^{i-1} M_j + \sum_{j=i+1}^{n} M'_{ij}$, $\forall\ i \in V$.

Summing up these inequalities for all $i \in V$, we have, $(n-1) \sum_{i=1}^{n} x(\{i\}) \geq \sum_{i=1}^{n} (\sum_{j=1}^{i-1} M_j + \sum_{j=i+1}^{n} M'_{ij}) \geq \sum_{i=1}^{n} (\sum_{j=1}^{i-1} M_j + \sum_{j=i+1}^{n} M_j) = \sum_{i=1}^{n} (\sigma(V) - M_i) = (n-1)\sigma(V)$.

We have known that $\sum_{i=1}^{n} x(\{i\}) = \sum_{j=1}^{n} M_j = \sigma(V)$ and then $M_j = M'_{ij}$, $\forall i, j \in V$. Thus, $\sigma(S) = \sum_{i \in S} \sigma(\{i\})$.

An important application of Lemma 1 is Theorem 1.

Theorem 1. *Deciding whether $\mathcal{C}(\Gamma_f)$ is empty can be done in polynomial time and an allocation in $\mathcal{C}(\Gamma_f)$ can be computed in polynomial time if $\mathcal{C}(\Gamma_f)$ is not empty.*

Proof (Sketch). First, it takes polynomial time to check the non-emptiness of $\mathcal{C}(\Gamma_f)$. When $\mathcal{C}(\Gamma_f)$ is not empty, then $(x_j = \sigma(V), \mathbf{0}_{\{i: i \neq j\}}) \in \mathcal{C}(\Gamma_f)$ when j is a veto player and $(\sigma(\{1\}), \cdots, \sigma(\{n\})) \in \mathcal{C}(\Gamma_f)$ when (ii) is satisfied.

The detail proof of Theorem 1 is shown in our full version [4].

4 Computing Relative Least-Core Value

From Lemma 1, $\mathcal{C}(\Gamma_f)$ may be empty in many cases. It is obvious that $\mathcal{RLCV}(\Gamma_f) > 0$ if $\mathcal{C}(\Gamma_f) = \emptyset$ and $\mathcal{RLCV}(\Gamma_f) = 0$ otherwise. In this section, we study computational properties of the RLCV problem. The linear programming corresponding to $\mathcal{RLCV}(\Gamma_f)$ (LP-RLCV) is as follows:

$$\begin{aligned} &\min r \\ &\text{s.t.} \begin{cases} x(V) = \sigma(V) \\ x(S) \geq (1-r)\sigma(S) \ \forall\ S \subseteq V,\ \sigma(S) \geq \eta \\ x(\{i\}) \geq 0 \qquad\qquad\ \ \forall\ i \in V \end{cases} \end{aligned} \tag{4}$$

A special case of computing $\mathcal{RLCV}(\Gamma_f)$ is when $\eta = 0$. It captures the scenario that the profit of any coalition exactly equals to its influence spread under influence cooperative game. In Theorem 2 we show that, although there are exponential number of constraints, LP-RLCV can be solved in polynomial time by providing a polynomial time separation oracle when $\eta = 0$. A separation oracle for a linear program is an algorithm that, given a putative feasible solution, checks whether it is indeed feasible, and if not, outputs a violated constraint. It is known that a linear program can be solved in polynomial time by the ellipsoid method as long as it has a polynomial time separation oracle [14].

Theorem 2. *There exists a polynomial time separation oracle of LP-RLCV when $\eta = 0$. Therefore, RLCV can be solved in polynomial time when $\eta = 0$.*

Proof. Given any solution candidate of LP-RLCV (x', r'), we need to either assert (x', r') is a feasible solution or find a constraint in LP-RLCV such that (x', r') violates it. Note that, checking $x'(V) = \sigma(V)$ and $x'(\{i\}) \geq 0$ ($\forall\ i \in V$) can be done in polynomial time. Thus, we only need to check whether $g(S) \triangleq 1 - x'(S)/\sigma(S) \leq r',\ \forall S \subseteq V$.

An important property is that $g(S)$ achieves its maximum value when S contains only one single player. This is because $g(S) = 1 - \frac{x'(S)}{\sigma(S)} \leq 1 - \frac{\sum_{i \in S} x'_i}{\sum_{i \in S} \sigma(\{i\})} \leq 1 - \min_{i:i \in S}\{\frac{x'_i}{\sigma(\{i\})}\} = \max_{i:i \in S}\{g(\{i\})\}$. The first inequality is due to the submodularity of $\sigma(S)$ and the second inequality is due to $\min_{i:i\in[n]}\{\frac{a_i}{b_i}\} \leq \frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i}$, $\forall a_i, b_i \in \mathbb{R}$. Thus, the exponential number of constraints can be simplified to n constraints on all single players. Then, we can find a polynomial time separation oracle of LP-RLCV directly.

When $\eta = 0$, RLCV can be solved in polynomial time is mainly because the most dissatisfaction coalition is a single player. However, when $\eta \neq 0$, it becomes intractable to find the most dissatisfaction coalition.

Theorem 3. *There is no polynomial time separation oracle of LP-RLCV for some $\eta > 0$, unless P=NP.*

Theorem 3 can not imply the NP-hardness of RLCV. However, the proof of Theorem 3 reveals an interesting connection between RLCV problem and a

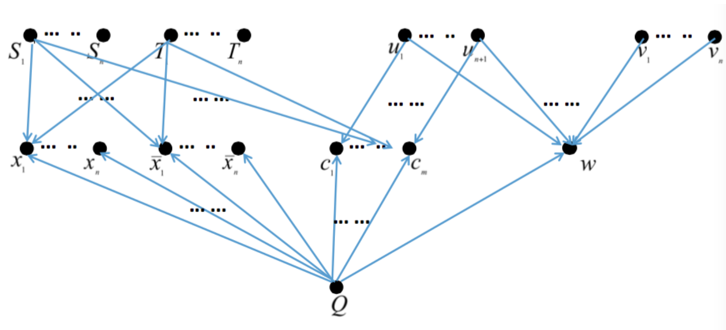

Fig. 1. The reduction from SAT to $\mathcal{RLCV}(\mathit{\Gamma}_f)$

series of well defined combinatorial problems. The proof of Theorem 3 and the generalized combinatorial problem is shown in our full version [4].

In the left of this section, we prove the NP-hardness of RLCV, a stronger hardness result than which in Theorem 3.

Theorem 4. *It is NP-hard to compute* $\mathcal{RLCV}(\mathit{\Gamma}_f)$*, even under influence cooperative game.*

Proof (Sketch). We construct a reduction from the SAT problem. A boolean formula is in conjunctive normal form (CNF) if it is expressed as an AND of clauses, each of which is the OR of one or more literals. The SAT problem is defined as follows: given a CNF formula F, determine whether F has a satisfiable assignment. Let F be a CNF formula with m clauses $C_1, C_2, \cdots, C_m$, over n literals $z_1, z_2, \cdots, z_n$. Without loss of generality, we set $m > 4n$.

We construct a social graph G as follows: $G = (V_1 \cup V_2 \cup V_3, E)$ is a tripartite graph (see the sketch graph in Figure 1). In the first layer (V_1), there are two nodes S_i and T_i corresponding to each $i \in \{1, 2, \cdots, n\}$, $n+1$ dummy nodes labeled as $u_1, u_2, \cdots, u_{n+1}$ and n dummy nodes labeled as $v_1, v_2, \cdots, v_n$. In the second layer (V_2), there are two nodes x_i and $\overline{x}_i$ corresponding to each $i \in \{1, 2, \cdots, n\}$, one node c_j for each $j \in \{1, 2, \cdots, m\}$ and a dummy node w. The third layer (V_3) contains only node Q. Edges exist only between the adjacent layers. For each $i \in \{1, 2, \cdots, n\}$, S_i sends an edge to every node in $\{x_i, \overline{x}_i\} \cup \{c_j$: clause C_j contains literal $z_i, j \in \{1, 2, \cdots, m\}\}$. Similarly, for each $i \in \{1, 2, \cdots, n\}$, T_i sends an edge to every node in $\{x_i, \overline{x}_i\} \cup \{c_j$: clause C_j contains literal $\overline{z}_i, j \in \{1, 2, \cdots, m\}\}$. The probabilities on edges sent from S_i and T_i are 1. There is an edge with influence probability 1 from u_i to c_i for any $i \in \{1, 2, \cdots, n\}$ and $m - n$ edges form u_{n+1} to $c_{n+1}, c_{n+2}, \cdots, c_m$. There is an edge from u_i to w with influence probability $1 - \sqrt[n+1]{1/2}$ for any $i \in \{1, 2, \cdots, n+1\}$. There is also exists an edge from v_i to w with influence probability $1 - \sqrt[n]{1/2}$ for any $i \in \{1, 2, \cdots, n\}$. The left edges are from Q to all nodes in the second layer. The influence probability on edge (Q, w) is $1/2$ and all other probabilities on edges sent from Q is 1. The influence cooperative game defined on G is $\Gamma(G) = (V = V_1 \cup V_3, \sigma^{\mathrm{IC}}(\cdot), \eta = 2n + m + 1/2)$. For convenient, we set $N = 2n + m$.

Suppose r^* is the optimal solution of the relative least-core value of $\Gamma(G)$. We can prove that $r^* \geq 1 - \frac{1}{3}(N + \frac{7}{8})/(N + \frac{1}{2})$ if F is satisfiable and $r < 1 - \frac{1}{3}(N + \frac{7}{8})/(N + \frac{1}{2})$ if F is un-satisfiable. The proof of this part is shown in the full version [4].

5 Computing Absolute Least-Core Value

5.1 Hardness of ALCV

Theorem 5. *ALCV problem of influence cooperative game cannot be approximated within 1.139 under the unique games conjecture.*

Proof (Sketch). We construct a reduction from MAX-CUT problem. Under our construction, for any instance of MAX-CUT problem, we can construct an instance of ALCV problem such that the optimal solution of these two instances are equal. The detail proof is shown in our full version [4].

5.2 Approximating $\mathcal{ALCV}(\Gamma_f)$

In this section, we approximate $\mathcal{ALCV}(\Gamma_f)$ by approximating the following linear programming (LP-PRIME):

$$\begin{aligned} &\min \ \varepsilon \\ &\text{s.t.} \ \begin{cases} x(V) = \sigma(V) \\ x(\{S\}) \geq \sigma(\{S\}) - \varepsilon & \forall S \subseteq V, \sigma(S) \geq \eta \\ x(\{u\}) \geq 0 & \forall u \in V \end{cases} \end{aligned}$$

The intractability of LP-PRIME lies on the exponential number of constraints and the hardness of identifying all successful coalitions. We use a relaxed version LP-RE and a strengthen version LP-STR of LP-PRIME to design an approximation algorithm of $\mathcal{ALCV}(\Gamma_f)$. (5) and (6) are formal definitions of LP-RE and LP-STR, respectively.

$$\begin{aligned} &\min \ \varepsilon \\ &\text{s.t.} \ \begin{cases} x(V) = \sigma(V) \\ x(S) \geq \eta - \varepsilon & \forall \ S \subseteq V, \sigma(S) \geq \eta \\ x(\{u\}) \geq 0 & \forall \ u \in V \end{cases} \end{aligned} \tag{5}$$

$$\begin{aligned} &\min \ \varepsilon \\ &\text{s.t.} \ \begin{cases} x(V) = \sigma(V) \\ x(S) \geq \sigma(S) - \varepsilon & \forall \ S \subseteq V \\ x(\{u\}) \geq 0 & \forall \ u \in V \end{cases} \end{aligned} \tag{6}$$

Intuitively, LP-RE and LP-STR denote absolute least-core values of two cooperative games with new profit functions. Specifically, LP-RE relaxes the constraints in LP-PRIME by reducing the profits of all successful coalitions excepting V to η. Formally, the profit function in LP-RE is $g(S)$: $g(V)=\sigma(V)$, $\forall\ S \subset V$, $g(S)=\eta$ if $\sigma(S)\geq\eta$ and $g(S)=0$ otherwise. The profit function in LP-STR is $h(S)=\sigma(S)$, $\forall S\subseteq V$. Clearly, LP-STR strengthens LP-PRIME by increasing the profits of all unsuccessful coalitions.

Our main result in this section is shown in Theorem 6.

Theorem 6. $\forall\ \delta>0$*, there exists an approximate algorithm* $\mathcal{A}$ *of the* $\mathcal{ALCV}(\Gamma_f)$ *problem with running time in poly*$(n, 1/\delta, \log\sigma(V))$*,* $\mathcal{A}$ *outputs* ε'_p *such that* $\varepsilon^*_p \leq \varepsilon'_p \leq \min\{\varepsilon^*_p+\sigma(V)-\eta+2\delta,\ \ \max\{3\varepsilon^*_p,\eta\}\}$.

We prove Theorem 6 by show Lemma 2, Lemma 3 and Lemma 4 in order.

Lemma 2. *Suppose the optimal value of LP-PRIME, LP-RE and LP-STR are* ε^*_p*,* ε^*_r *and* ε^*_s*, respectively. Then, we have*

$$\varepsilon^*_p \leq \varepsilon^*_r + (\sigma(V)-\eta) \leq \varepsilon^*_p + (\sigma(V)-\eta), \tag{7}$$

$$\varepsilon^*_p \leq \varepsilon^*_s \leq \max\{\varepsilon^*_p, \eta\}. \tag{8}$$

Lemma 3. *There exists a polynomial time approximate algorithm of LP-STR outputting* ε'_s *such that* $\varepsilon^*_s \leq \varepsilon'_s \leq 3\varepsilon^*_s$.

Lemma 4. $\forall\ \delta>0$*, there exists an algorithm of LP-RE outputting* ε'_r *such that* $\varepsilon^*_r \leq \varepsilon'_r \leq \varepsilon^*_r + 2\delta$*, with runs time in poly*$(n, 1/\delta, \log\sigma(V))$.

The proofs of Lemma 2 - Lemma 4 rely heavily on mathematical computation and we report them in our full version [4].

6 Computing Least Average Dissatisfaction Value

Based on Definition 3, $\mathcal{LADV}(\Gamma_f)$ equals the optimal value of the following linear program:

$$\begin{aligned} &\min F(x) = \tfrac{1}{2^n}\textstyle\sum_{S\subseteq V}\max\{f(S)-x(S),0\} \\ &\text{s.t.}\ \begin{cases} x(V)=\sigma(V) \\ x(\{i\})\geq 0 \qquad \forall\ i\in V \end{cases} \end{aligned} \tag{9}$$

Where $f(S)=\sigma(S)$ if $\sigma(S)\geq\eta$ and $f(S)=0$ otherwise. There are exponential terms in $F(x)$, however, we can utilize stochastic gradient algorithm to approximate the optimal solution of (9). This is because the object function $F(x)$ is a convex function (Lemma 5) and the feasible solution area in (9) is a convex set.

Lemma 5. $F(x)$ *is a convex function.*

The proof of Lemma 5 is shown in our full version [4]. The stochastic gradient descent algorithm (SGD, cf. [20]) can be used to compute $\mathcal{LADV}(\Gamma_f)$ (see Algorithm 1).

Let F^* be the optimal solution of $\mathcal{LADV}(\Gamma_f)$, $\hat{F}$ be the output of Algorithm 1 and the profit of grand coalition $\sigma(V)=V$. Then, the performance of Algorithm 1 can be formalized in the following theorem.

Algorithm 1 Stochastic gradient descent for LADV

1: **Parameters:** Scaler $\alpha > 0$, integer $T > 0$
2: **Initialize:** $\mathbf{X}^1 = \mathbf{0}$, $t = 0$.
3: Set $D = \{\mathbf{X} : \mathbf{X}_i \geq 0(\forall\ i \in V),\ \sum_{i \in V} \mathbf{X}_i = \sigma(V)\}$.
4: **for** $t = 1$ to T **do**
5:     /*choose a random $\mathbf{Y}^t$ such that $\mathbb{E}[\mathbf{Y}^t|\mathbf{X}^t]$ is a subgradient of F.*/
6:     Uniformly at random choose a set $S \in 2^V$.
7:     **if** $f(S) \geq \mathbf{X}^t(S)$ **then**
8:         Set $\mathbf{Y}^t = (-\mathbf{1}_S, \mathbf{0}_{V \setminus S})$.
9:     **else**
10:         Set $\mathbf{Y}^t = \mathbf{0}$.
11:     **end if**
12:     update $\mathbf{X}^{t+\frac{1}{2}} = \mathbf{X}^t - \alpha \mathbf{Y}^t$.
13:     /*Project $\mathbf{X}^{t+\frac{1}{2}}$ to D*/
14:     $\mathbf{X}^{t+1} = \arg\min_{\mathbf{X} \in D} \|\mathbf{X} - \mathbf{X}^{t+\frac{1}{2}}\|^2$.
15: **end for**
16: **return** $\hat{F} = \min\{F(\mathbf{X}^t)\}_{t \in \{1,2,\cdots,T\}}$.

Theorem 7. *$\forall\ \varepsilon > 0$, $\mathbb{E}[\hat{F}] - F^* \leq \varepsilon$ if $T \geq \frac{\sigma(V)^4 n^4}{\varepsilon^2}$ and $\alpha = \sqrt{\frac{\sigma(V)^4}{T n^4}}$ in Algorithm 1.*

Following the standard analysis of SGD (e.g. in Chapter 14 of [20]), Theorem 7 holds since it is easy to check that $\mathbb{E}[\mathbf{Y}^t|\mathbf{X}^t]$ is a subgradient of $F(\mathbf{X})$ at node $\mathbf{X}^t$, for any $t \in [T]$ (lines 6–11 in Algorithm 1).

7 Conclusion and future work

In this paper, we study the core related solution concepts of truncated submodular profit cooperative game. One possible future work is to change the way of truncating a function. For example, we can set $f(S) = \sigma(S)$ if $|S| \geq k$ and $f(S) = 0$ otherwise. This setting is a special case of the setting in our paper and thus it may allow efficient algorithms. In this paper, we prove that computing the relative least-core value is NP-hard. We also prove that the relative least-core value can be solved in polynomial time in a special case. A directly future work is to design an approximate algorithm of RLCV under general case.

References

1. Bemmaor, A.C.: Testing alternative econometric models on the existence of advertising threshold effect. Journal of Marketing Research **21**(3), 298–308 (1984)
2. Chalkiadakis, G., Elkind, E., Wooldridge, M.: Computational aspects of cooperative game theory. Synthesis Lectures on Artificial Intelligence and Machine Learning **5**(6), 1–168 (2011)
3. Chen, W., Lakshmanan, L.V.S., Castillo, C.: Information and influence propagation in social networks. Morgan & Claypool Publishers (2013)

4. Chen, W., Shan, X., Sun, X., Zhang, J.: Coreness of cooperative games with truncated submodular profit functions. In: arXiv:1806.10833 (2018)
5. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: KDD. pp. 199–208. ACM (2009)
6. Conitzer, V., Sandholm, T.: Complexity of constructing solutions in the core based on synergies among coalitions. Artificial Intelligence **170**(6-7), 607–619 (2006)
7. Demange, G.: On group stability in hierarchies and networks. Journal of Political Economy **112**, 754–778 (2004)
8. Deng, X., Papadimitriou, C.H.: On the complexity of cooperative solution concepts. Mathematics of Operations Research **19**(2), 257–266 (1994)
9. Domingos, P., Richardson, M.: Mining the network value of customers. In: KDD. pp. 57–66. ACM (2001)
10. Faigle, U., Kern, W.: On some approximately balanced combinatorial cooperative games. Math. Methods Oper. Res. **38**, 141–152 (1993)
11. Gillies, D.: Some Theorems on n-Person Games. Ph.D. thesis, Princeton University (1953)
12. Goyal, A., Bonchi, F., Lakshmanan, L.V., Venkatasubramanian, S.: On minimizing budget and time in influence propagation over social networks. Social Network Analysis and Mining pp. 1–14 (2012)
13. Granovetter, M.: Threshold models of collective behavior. American Journal of Sociology pp. 489–515 (1978)
14. Grötschel, M., Lovász, L., Schrijver, A.: Geometric algorithms and combinatorial optimization, vol. 2. Springer Science & Business Media (2012)
15. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: KDD. pp. 137–146. ACM (2003)
16. Maschler, M., Peleg, B., Shapley, L.S.: Geometric properties of the kernel, nucleolus, and related solution concepts. Math. Oper. Res. **4**(4), 303–338 (1979)
17. Meir, R., Rosenschein, J.S., Malizia, E.: Subsidies, stability, and restricted cooperation in coalitional games. In: IJCAI. pp. 301–306 (2011)
18. Mossel, E., Roch, S.: Submodularity of influence in social networks: from local to global. SIAM Journal on Computing **39**(6), 2176–2188 (2010)
19. Schulz, A.S., Uhan, N.A.: Approximating the least core value and least core of cooperative games with supermodular costs. Discrete Optimization **10**(2), 163–180 (2013)
20. Shalev-Shwartz, S., Ben-David, S.: Understanding machine learning: From theory to algorithms. Cambridge university press (2014)
21. Shapley, L.S.: Markets as cooperative games. In: IJCAIRand Corporation Memorandum (1955)
22. Tang, Y., Shi, Y., Xiao, X.: Influence maximization in near-linear time: A martingale approach. In: SIGMOD. pp. 1539–1554. ACM (2015)