Computing Least Cores of Supermodular Cooperative Games

Daisuke Hatano
National Institute of Informatics
JST, ERATO, Kawarabayashi Large Graph Project
hatano@nii.ac.jp

Yuichi Yoshida
National Institute of Informatics, *and*
Preferred Infrastructure, Inc.
yyoshida@nii.ac.jp

Abstract

One of the goals of a cooperative game is to compute a value division to the players from which they have no incentive to deviate. This concept is formalized as the notion of the core. To obtain a value division that motivates players to cooperate to a greater extent or that is more robust under noise, the notions of the strong least core and the weak least core have been considered. In this paper, we characterize the strong and the weak least cores of supermodular cooperative games using the theory of minimizing crossing submodular functions. We then apply our characterizations to two representative supermodular cooperative games, namely, the induced subgraph game generalized to hypergraphs and the airport game. For these games, we derive explicit forms of the strong and weak least core values, and provide polynomial-time algorithms that compute value divisions in the strong and weak least cores.

Introduction

Cooperation and the sharing of resources are crucial factors in achieving a sustainable and resilient economy. Based on these concepts, services such as vehicle sharing, office sharing, house sharing, and skill sharing (Reisch and Thgersen 2015) can be realized. Individuals can reduce their own costs by dividing the cost of services with other customers, instead of using these services alone. A crucial challenge in such scenarios is to divide the cost so that all customers are satisfied with the division and are motivated to cooperate.

In this paper, we tackle this problem from the game-theoretic perspective of *cooperative games*. We explain the motivation underlying cooperative games using the example of sharing taxis. The total fare for a group of customers taking a taxi is typically determined by the customer who travels farthest in the group. Our task is to divide the total fare among these customers. However, if some customers can pay less by forming a smaller group and dividing the total fare of the subgroup appropriately, they will leave the larger group. Hence, we need to find a stable division that avoids such a deviation. This concept have been formalized as the notion of the *core* (Gillies 1959).

It is known that the core exists for the taxi example. More generally, there is a core within *supermodular cooperative games* (sometimes called *convex cooperative games*), that is, cooperative games with supermodular utility functions (Shapley 1971). Indeed, a value division called the *Shapley value* (Shapley 1967) is always in the core.

It is natural to consider what is the "stablest" value division, rather than simply any stable value division. One reason is to motivate the players of the game to cooperate to the greatest extent. From the perspective of a manager of the game, this extent can be regarded as the tax that can be imposed on players while still ensuring they play the game. Of course, the manager aims to maximize the tax. A second reason to identify the stablest value division is to enhance robustness to noise. In practice, we cannot accurately measure the valuation of each group because of external noise. In such a case, the loss incurred when a group deviates (with respect to the observed valuations) should be regarded as the safety margin for the group not to deviate (Li and Conitzer 2015). To address these issues, several extensions of the core have been considered. Two well-known examples are the *strong least core* and the *weak least core* (Shapley and Shubik 1966). Indeed, it is known that value divisions in these cores are most robust against a certain type of noise (Li and Conitzer 2015).

In this paper, we consider computational aspects of the strong and weak least cores. First, we provide theoretical characterizations of those cores of supermodular cooperative games using the theory of minimizing crossing submodular functions. To see that our characterizations are useful to interpret those cores and design efficient algorithms for computing a value division in them, we analyze two representative supermodular cooperative games, namely, the induced subgraph game (Deng and Papadimitriou 1994) generalized to hypergraphs and the airport game (Littlechild and Owen 1973). The induced subgraph game models the profit among countries from forming alliances, whereas the airport game can model the sharing of a runway at an airport and the sharing of taxis. To the best of our knowledge, there has been no computational analysis of the weak least core, and the difference between the strong and weak least cores has not been discussed. We emphasize that we obtain the strong and weak least core values in a uniform manner, which helps us to analyze the difference between them.

Copyright © 2017, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Related Work

Supermodular games are a well-studied class of cooperative games. Examples include the induced subgraph game (Deng and Papadimitriou 1994), airport game (Littlechild and Owen 1973), public good game (Oishi and Nakayama 2009), bidder collusion game (Graham, Marshall, and Richard 1990), multicast tree game (Feigenbaum, Papadimitriou, and Shenker 2001), and bankruptcy game (O'Neill 1982). Our characterizations of the strong and weak least core values can be applied to each of these.

Algorithms for computing the strong least core value have already been developed (Kuipers 1996; Faigle, Kern, and Kuipers 2001). However, as these algorithms solve optimization problems on the extended polymatroid associated with a crossing submodular function, we suspect that they are essentially impractical. In contrast, we have characterized the strong and weak least cores, and used these characterizations to obtain efficient algorithms that compute the strong and weak least core values for specific games.

When defining the strong and weak least cores, we strengthened the stability constraint $\boldsymbol{x}(S) \geq \nu(S)$ in the definition of the core. An alternative approach is to replace the constraint $\boldsymbol{x}(V) = \nu(V)$ by $\boldsymbol{x}(V) = \nu(V) + \epsilon$ (see the formal definition of the core in the next section). The minimum ϵ that renders the program feasible is called the *cost of stability* (Bachrach et al. 2009). This corresponds to having a benevolent external party that wishes to stabilize the game by offering subsidies to players if they remain in the grand coalition. The cost of stability focuses on the case where the core is empty, and the cost of stability of a supermodular game is always zero. Hence, the cost of stability is inappropriate for defining the stablest value division.

Cooperative games represented by marginal contribution nets (Ieong and Shoham 2005) can be seen as a generalization of the induced subgraph game for which negative weights are allowed. Although (Hirayama et al. 2014) considered computing the least cores in this general setting, the time complexity of their algorithm is exponential.

Preliminaries

For an integer k, we denote the set $\{1, 2, \ldots, k\}$ as $[k]$. The set of non-negative real values is denoted by $\mathbb{R}_+$. We use bold symbols such as $\boldsymbol{x}$ to denote vectors. Let V be a finite set. For a vector $\boldsymbol{x} \in \mathbb{R}^V$ and a set $S \subseteq V$, we denote the value $\sum_{v \in S} \boldsymbol{x}(v)$ as $\boldsymbol{x}(S)$.

A function $f : 2^V \to \mathbb{R}$ is called *submodular* (resp., *supermodular*) if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T)$$
$$(resp., f(S) + f(T) \leq f(S \cap T) + f(S \cup T))$$

for all $S, T \subseteq V$.

We say that two sets $S, T \subseteq V$ are *crossing* if $S \cap T \neq \emptyset$, $S \setminus T \neq \emptyset$, $T \setminus S \neq \emptyset$, and $S \cup T \neq V$. A function $f : 2^V \to \mathbb{R}$ is called *crossing submodular* (Fujishige 2005) if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T)$$

holds for any crossing $S, T \subseteq V$.

We say that a family of sets $\mathcal{S}$ is a *copartition* of V if $\{V \setminus S \mid S \in \mathcal{S}\}$ is a partition of V. For an integer k, let $\mathcal{P}_k(V)$ (resp., $\bar{\mathcal{P}}_k(V)$) be the family of all partitions (resp., copartitions) $\mathcal{S}$ of V into at least k sets, such that each set $S \in \mathcal{S}$ is *nontrivial*, that is, $\emptyset \subsetneq S \subsetneq V$.

Cooperative Games

We now briefly describe the framework of cooperative games. For more details, readers are referred to (Chalkiadakis, Elkind, and Wooldridge 2011) and the references therein. A *cooperative game* is a pair (V, ν), where V is a set of players and $\nu : 2^V \to \mathbb{R}_+$ is a function called the *characteristic function*. We can regard $\nu(S)$ as the profit when the players in S form a coalition. We always assume that $\nu(\emptyset) = 0$. We say that $\boldsymbol{x} \in \mathbb{R}^V$ is a *value division* if $\boldsymbol{x}(V) = \nu(V)$, that is, a value division is a distribution of the total profit to the players.

Suppose that a value division $\boldsymbol{x} \in \mathbb{R}^V$ satisfies $\boldsymbol{x}(S) < \nu(S)$ for some $S \subseteq V$. In such a case, the players in S will form a coalition and leave V. We say that a value division is in the *core* if such an S does not exist. More formally, a value division $\boldsymbol{x} \in \mathbb{R}^V$ is in the core if $\boldsymbol{x}(S) \geq \nu(S)$ for all $S \subseteq V$.

We say that a cooperative game (V, ν) is a *supermodular (cooperative) game* if ν is supermodular. It is known that the core of a supermodular game is nonempty; in particular, a value division called the Shapley value is always in the core (Shapley 1967).

If we wish to ensure that players are tightly retained, it is natural to consider stronger requirements. Let us consider the following two extensions of the core.

- For $\epsilon \in \mathbb{R}$, we say that a value division $\boldsymbol{x} \in \mathbb{R}^V$ is in the *strong* $(-\epsilon)$*-core* if the loss of a deviating nontrivial coalition is at least ϵ, that is, $\boldsymbol{x}(S) - \nu(S) \geq \epsilon$ for any $\emptyset \subsetneq S \subsetneq V$.
- For $\epsilon \in \mathbb{R}$, we say that a value division $\boldsymbol{x} \in \mathbb{R}^V$ is in the *weak* $(-\epsilon)$*-core* if the loss of a deviating nontrivial coalition is at least ϵ on average, that is, $\boldsymbol{x}(S) - \nu(S) \geq \epsilon|S|$ for any $\emptyset \subsetneq S \subsetneq V$.

Note that the core is equal to the strong 0-core and the weak 0-core.

The *strong least core value* of a game is the minimum $\epsilon \in \mathbb{R}$ such that the strong ϵ-core is nonempty. The *strong least core* of a game is the strong ϵ-core for the strong least core value ϵ. The *weak least core value* and the *weak least core* are defined analogously using the notion of the weak ϵ-core. Note that the strong and the weak least core values of a supermodular game are always nonpositive because the core is nonempty. Hence, it is convenient to consider the strong and weak $(-\epsilon)$-cores instead of the strong and the weak ϵ-cores.

Let $\delta^{\mathrm{s}}(\epsilon, S) = \epsilon$ and $\delta^{\mathrm{w}}(\epsilon, S) = \epsilon|S|$. The strong and weak least core values of a cooperative game can be computed by solving the following linear program (LP).

$$\begin{array}{lll} \min & -\epsilon & \\ \text{s.t.} & \boldsymbol{x}(S) \geq \nu(S) + \delta^{\text{type}}(\epsilon, S) & \forall \emptyset \subsetneq S \subsetneq V \\ & \boldsymbol{x}(V) = \nu(V) & \\ & \boldsymbol{x}(v) \geq 0 & \forall v \in V, \end{array} \quad (1)$$

where type is 's' in the strong core case and 'w' in the weak core case. Note that, for supermodular games, $\epsilon \geq 0$ holds in the optimal solution.

Characterizations of Least Cores

In this section, we characterize the strong and weak least core values of a supermodular game.

Let (V, ν) be a supermodular game. Instead of solving the optimization problem (1), we fix $\epsilon \geq 0$ and consider its feasibility. We define a function $f_\epsilon^{\text{type}} : 2^V \to \mathbb{R}$ as

$$f_\epsilon^{\text{type}}(S) = \begin{cases} 0 & \text{if } S = \emptyset, \\ -\nu(S) - \delta^{\text{type}}(\epsilon, S) & \text{if } \emptyset \subsetneq S \subsetneq V, \\ -\nu(V) & \text{if } S = V. \end{cases}$$

It is clear that f_ϵ^{type} is crossing submodular (but not necessarily submodular).

The feasibility version of (1) can then be rephrased as follows:

$$\begin{array}{ll} \boldsymbol{x}(S) \leq f_\epsilon^{\text{type}}(S) & \forall \emptyset \subsetneq S \subsetneq V, \\ \boldsymbol{x}(V) = f_\epsilon^{\text{type}}(V), & \\ \boldsymbol{x}(v) \leq 0 & \forall v \in V. \end{array} \tag{2}$$

Note that the variable $\boldsymbol{x}(v)$ is negated for each $v \in V$.

For a function $f : 2^V \to \mathbb{R}$, we can write the *(generalized) extended polymatroid* associated with f as

$$P(f) = \{\boldsymbol{x} \in \mathbb{R}^V \mid \boldsymbol{x}(S) \leq f(S) \text{ for all } \emptyset \subsetneq S \subsetneq V\}.$$

We define $p_\epsilon^{\text{type}} = \max\{\boldsymbol{x}(V) \mid \boldsymbol{x} \in P(f_\epsilon^{\text{type}})\}$. Then, the program (2) is feasible if and only if $p_\epsilon^{\text{type}} \geq f_\epsilon^{\text{type}}(V)$.

It is known that p_ϵ^{type} can be computed in polynomial time (Frank and Tardos 1988; Naitoh and Fujishige 1992). However, as the algorithm is cumbersome to describe and implement, we further simplify the problem using the properties of supermodular cooperative games. We use the following theorem.

Theorem 1 ((Fujishige 1984))**.** *Let $f : 2^V \to \mathbb{R}$ be a crossing submodular function and let $p = \max\{\boldsymbol{x}(V) \mid \boldsymbol{x} \in P(f)\}$. Let $q, r \in \mathbb{R}$ be defined as*

$$q = \min_{\mathcal{S} \in \mathcal{P}_2(V)} \sum_{S \in \mathcal{S}} f(S),$$

$$r = \min_{\mathcal{S} \in \bar{\mathcal{P}}_3(V)} \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} f(S).$$

Then, we have $p = \min\{q, r\}$.

Consider Theorem 1 instantiated with the function f_ϵ^{type}, and set $q = q_\epsilon^{\text{type}}$ and $r = r_\epsilon^{\text{type}}$. From Theorem 1, the program (2) is feasible if and only if

$$\min\{q_\epsilon^{\text{type}}, r_\epsilon^{\text{type}}\} \geq f_\epsilon^{\text{type}}(V). \tag{3}$$

We now derive the strong and weak least core values using (3).

Theorem 2. *The strong least core value of a supermodular game (V, ν) is*

$$-\min\Big\{ \min_{\mathcal{S} \in \mathcal{P}_2(V)} \frac{1}{|\mathcal{S}|} \Big(\nu(V) - \sum_{S \in \mathcal{S}} \nu(S)\Big), \min_{\mathcal{S} \in \bar{\mathcal{P}}_2(V)} \frac{1}{|\mathcal{S}|} \Big((|\mathcal{S}| - 1)\nu(V) - \sum_{S \in \mathcal{S}} \nu(S)\Big)\Big\}.$$

Proof. Note that the strong least core value is $-\epsilon$ for the maximum $\epsilon \geq 0$ such that (3) holds, where the type is 's'.

We first consider the condition $q_\epsilon^{\text{s}} \geq f_\epsilon^{\text{s}}(V)$. Let $\mathcal{S} \in \mathcal{P}_2(V)$ be a partition of V. We then have

$$\begin{aligned} \sum_{S \in \mathcal{S}} f_\epsilon^{\text{s}}(S) - f_\epsilon^{\text{s}}(V) &= \nu(V) - \sum_{S \in \mathcal{S}} (\nu(S) + \delta^{\text{s}}(\epsilon, S)) \\ &= \nu(V) - \sum_{S \in \mathcal{S}} \nu(S) - \epsilon|\mathcal{S}|. \end{aligned} \tag{4}$$

Hence, $q_\epsilon^{\text{s}} \geq f_\epsilon^{\text{s}}(V)$ holds if and only if $\nu(V) - \sum_{S \in \mathcal{S}} \nu(S) - \epsilon|\mathcal{S}| \geq 0$ holds for any partition $\mathcal{S}$ of V.

Next, we consider the condition $r_\epsilon^{\text{s}} \geq f_\epsilon^{\text{s}}(V)$. Let $\mathcal{S} \in \bar{\mathcal{P}}_3(V)$ be a copartition of V. We then have

$$\begin{aligned} &\frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} f_\epsilon^{\text{s}}(S) - f_\epsilon^{\text{s}}(V) \\ &= \nu(V) - \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} (\nu(S) + \delta^{\text{s}}(\epsilon, S)) \\ &= \nu(V) - \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} \nu(S) - \frac{\epsilon|\mathcal{S}|}{|\mathcal{S}| - 1}. \end{aligned}$$

Hence, $r_\epsilon^{\text{s}} \geq f_\epsilon^{\text{s}}(V)$ holds if and only if $\nu(V) - \frac{1}{|\mathcal{S}|-1} \sum_{S \in \mathcal{S}} \nu(S) - \frac{\epsilon|\mathcal{S}|}{|\mathcal{S}|-1} \geq 0$ holds for any copartition $\mathcal{S} \in \bar{\mathcal{P}}_3(V)$.

Combining these two cases, we obtain the desired value. Note that a partition into two parts can be regarded as a copartition into two parts. Hence, we can replace $\bar{\mathcal{P}}_3$ by $\bar{\mathcal{P}}_2$. □

Theorem 3. *The weak least core value of a supermodular game (V, ν) is*

$$-\frac{1}{|V|} \min_{\mathcal{S} \in \bar{\mathcal{P}}_2(V)} \Big(\nu(V) - \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} \nu(S)\Big).$$

Proof. Note that the weak least core value is $-\epsilon$ for the maximum $\epsilon \geq 0$ such that (3) holds, where the type is 'w'.

We first consider the condition $q_\epsilon^{\text{w}} \geq f_\epsilon^{\text{w}}(V)$. Let $\mathcal{S} \in \mathcal{P}_2(V)$ be a partition of V. We then have

$$\begin{aligned} \sum_{S \in \mathcal{S}} f_\epsilon^{\text{w}}(S) - f_\epsilon^{\text{w}}(V) &= \nu(V) - \sum_{S \in \mathcal{S}} (\nu(S) + \delta^{\text{w}}(\epsilon, S)) \\ &= \nu(V) - \sum_{S \in \mathcal{S}} \nu(S) - \epsilon|V|. \end{aligned} \tag{5}$$

Since ν is supermodular, (5) is minimized when $\mathcal{S}$ is a partition into two parts. Hence, $q_\epsilon^{\text{w}} \geq f_\epsilon^{\text{w}}(V)$ holds if and only if $\nu(V) - (\nu(S_1) + \nu(S_2)) - \epsilon|V| \geq 0$ holds for any partition $\{S_1, S_2\}$ of V.

Next, we consider the condition $r_\epsilon^{\text{w}} \geq f_\epsilon^{\text{w}}(V)$. Let $\mathcal{S} \in \bar{\mathcal{P}}_3(V)$ be a copartition of V. We then have

$$\begin{aligned} &\frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} f_\epsilon^{\text{w}}(S) - f_\epsilon^{\text{w}}(V) \\ &= \nu(V) - \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} (\nu(S) + \delta^{\text{w}}(\epsilon, S)) \\ &= \nu(V) - \frac{1}{|\mathcal{S}| - 1} \sum_{S \in \mathcal{S}} \nu(S) - \epsilon|V|. \end{aligned}$$

Hence, $r^{\mathrm{w}}_\epsilon \geq f^{\mathrm{w}}_\epsilon(V)$ holds if and only if $\nu(V) - \frac{1}{|\mathcal{S}|-1}\sum_{S\in\mathcal{S}}\nu(S) - \epsilon|V| \geq 0$ holds for any copartition $\mathcal{S} \in \bar{\mathcal{P}}_3(V)$.

To summarize, (3) holds if and only if $\nu(V) - \frac{1}{|\mathcal{S}|-1}\sum_{S\in\mathcal{S}}\nu(S) - \epsilon|V| \geq 0$ holds for any copartition $\mathcal{S} \in \bar{\mathcal{P}}_2(V)$, which means that the weak least core value is

$$-\min_{\mathcal{S}\in\bar{\mathcal{P}}_2(V)} \frac{1}{|V|}\Big(\nu(V) - \frac{1}{|\mathcal{S}|-1}\sum_{S\in\mathcal{S}}\nu(S)\Big). \qquad \square$$

Induced Subgraph Game

Let $G = (V, E, w)$ be a weighted hypergraph, where V is a set of vertices, E is a set of hyperedges, and $w : E \to \mathbb{R}_+$ is a weight function on the hyperedges. We often regard a hyperedge $e \in E$ as a subset of V. The *induced subgraph game* associated with G is the cooperative game (V, ν), where $\nu : 2^V \to \mathbb{R}_+$ is the total weight of the hyperedges $e \in E$ with $e \subseteq S$. Note that ν is supermodular.

In this section, we consider induced subgraph games. We derive explicit forms of the strong and weak least core values, and present polynomial-time algorithms that compute the value divisions in the strong and weak least cores.

Definitions

Let $S \subseteq V$ be a set of vertices. We say that a hyperedge $e \in E$ is *cut* by S if $e \not\subseteq S$ and $e \not\subseteq V \setminus S$. The *cut weight* of S, denoted by $c(S)$, is the total weight of the hyperedges cut by S. A vertex set with the minimum cut weight is called a *minimum cut*. Let $c^*(G)$ denote the weight of the minimum cut of G.

Let $\mathcal{S}$ be a partition of V into at least two nontrivial parts. We say that a hyperedge $e \in E$ is *cut* by $\mathcal{S}$ if $e \not\subseteq S$ for any $S \in \mathcal{S}$. The *cut weight* of a partition $\mathcal{S}$ of V, denoted by $c(\mathcal{S})$, is the total weight of hyperedges cut by $\mathcal{S}$. We define $\bar{c}(\mathcal{S})$ as $\sum_{S\in\mathcal{S}} c(S) - c(\mathcal{S})$. That is, we count a hyperedge $e \in E$ exactly ℓ times if it intersects with $\ell + 1$ parts of $\mathcal{S}$.

Characterization of the strong least core value

We first derive the strong least core value of induced subgraph games.

Theorem 4. *The strong least core value of the induced subgraph game associated with a hypergraph $G = (V, E, w)$ is*

$$-\min_{\mathcal{S}\in\mathcal{P}_2} \frac{c(\mathcal{S})}{|\mathcal{S}|}.$$

Proof. From Theorem 2, the strong least core value is

$$-\min\Big\{\min_{\mathcal{S}\in\mathcal{P}_2}\frac{1}{|\mathcal{S}|}\Big(\nu(V) - \sum_{S\in\mathcal{S}}\nu(S)\Big), \min_{\mathcal{S}\in\bar{\mathcal{P}}_2}\frac{1}{|\mathcal{S}|}\Big((|\mathcal{S}|-1)\nu(V) - \sum_{S\in\mathcal{S}}\nu(S)\Big)\Big\}.$$

For a partition $\mathcal{S} \in \mathcal{P}_2(V)$, we have

$$\nu(V) - \sum_{S\in\mathcal{S}}\nu(S) = c(\mathcal{S}).$$

For a copartition $\mathcal{S} \in \bar{\mathcal{P}}_2(V)$, we have

$$\begin{aligned}
&(|\mathcal{S}|-1)\nu(V) - \sum_{S\in\mathcal{S}}\nu(S) \\
&= (|\mathcal{S}|-1)\nu(V) - \sum_{S\in\mathcal{S}}\Big(\nu(V) - \nu(V\setminus S) - c(V\setminus S)\Big) \\
&= \sum_{S\in\mathcal{S}}\Big(\nu(V\setminus S) + c(V\setminus S)\Big) - \nu(V) = \bar{c}(\bar{\mathcal{S}}),
\end{aligned}$$

where $\bar{\mathcal{S}}$ is the partition consisting of the complements of the sets in $\mathcal{S}$.

Since $c(\mathcal{S}) \leq \bar{c}(\mathcal{S})$ for any partition $\mathcal{S} \in \mathcal{P}_2(V)$ because the former counts each cut hyperedge once whereas the latter counts each cut hyperedge once or more, we have the desired result. $\square$

The value $c(\mathcal{S})/|\mathcal{S}|$ can be understood as how well the hypergraph is clustered by the partition $\mathcal{S}$. Hence, Theorem 4 implies that, the more tightly connected the hypergraph is, the more stable value division there exists.

Suppose that the input hypergraph is a graph, that is, each hyperedge has cardinality two. In this case, we can further simplify the strong least core value in Theorem 4.

Theorem 5. *The strong least core value of the induced subgraph game associated with a graph $G = (V, E, w)$ is*

$$-\frac{c^*(G)}{2}.$$

Proof. First, we show that the strong least core value is at least $-\frac{c^*(G)}{2}$. Suppose that there is a value division $\boldsymbol{x} \in \mathbb{R}^V$ in the strong $(-\epsilon)$-core with $\epsilon > \frac{c^*(G)}{2}$. Let S be the minimum cut of G. Then, $\boldsymbol{x}(S) > \nu(S) + \frac{c^*(G)}{2}$ and $\boldsymbol{x}(V \setminus S) > \nu(V\setminus S) + \frac{c^*(G)}{2}$. However, this implies that $\boldsymbol{x}(V) = \boldsymbol{x}(S) + \boldsymbol{x}(V\setminus S) > \nu(S) + \nu(V\setminus S) + c^*(G) = \nu(V) = \boldsymbol{x}(V)$, which is a contradiction.

We now show that there is a value division in the strong $-\frac{c^*(G)}{2}$-core. Let $\boldsymbol{x} \in \mathbb{R}^E$ be a value division defined as $\boldsymbol{x}(v) = \sum_{e\in E: v\in e} w(e)/2$ for every $v \in V$.

Let $\emptyset \subsetneq S \subsetneq V$ be a set of vertices. Then, $\boldsymbol{x}(S) = \nu(S) + \frac{c(S)}{2} \geq \nu(S) + \frac{c^*(G)}{2}$, which implies that $\boldsymbol{x}$ is in the strong $-\frac{c^*(G)}{2}$-core. $\square$

Characterization of the weak least core value

We now derive the weak least core value of induced subgraph games.

Theorem 6. *The weak least core value of the induced subgraph game associated with a hypergraph $G = (V, E, w)$ is*

$$-\frac{1}{|V|}\min_{\mathcal{S}\in\mathcal{P}_2(V)}\frac{\bar{c}(\mathcal{S})}{|\mathcal{S}|-1}.$$

Proof. From Theorem 3, the weak least core value is

$$-\frac{1}{|V|}\min_{\mathcal{S}\in\bar{\mathcal{P}}_2}\Big(\nu(V) - \frac{1}{|\mathcal{S}|-1}\sum_{S\in\mathcal{S}}\nu(S)\Big).$$

Using the calculation in the proof of Theorem 4 (for the copartition case), we have the desired result. $\square$

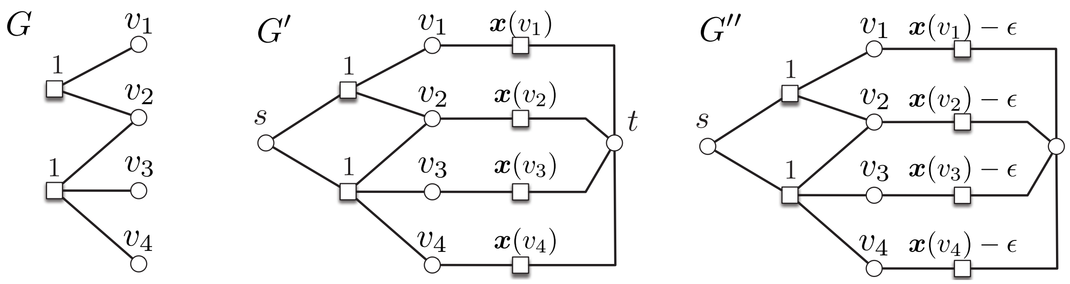

Figure 1: Reduction examples. Circles and boxes represent nodes and a hyperedges, respectively. The value associated with each box is the weight of the corresponding hyperedge. The graphs G' and G'' are obtained from G when computing the strong and weak least core values, respectively.

Suppose that the input hypergraph is a graph. Then, the value of $\bar{c}(\mathcal{S})$ is equal to $\frac{c(\mathcal{S})}{2}$. The value $\min_{\mathcal{S} \in \mathcal{P}_2(V)} \frac{c(\mathcal{S})}{|\mathcal{S}|-1}$ is called the *strength*, which is another notion representing how the graph is well clustered, and it can be computed in polynomial time (Cunningham 1985).

Algorithm for the strong least core

In this subsection, we provide a polynomial-time algorithm that computes a value division in the strong least core of the induced subgraph game associated with a hypergraph $G = (V, E, w)$. To this end, we construct a separation oracle for the LP (1), which returns a constraint (if any) that is not satisfied in the current LP solution $(\epsilon, \boldsymbol{x})$. Using the ellipsoid method (Khachiyan 1980), we can then solve LP (1) by calling the separation oracle polynomially many times in n.

The separation oracle is constructed as follows. As the last two constraints of the LP (1) are easy to check, we assume that the current solution $(\epsilon, \boldsymbol{x})$ satisfies these constraints.

We now wish to check $\boldsymbol{x}(S) \geq \nu(S) + \epsilon$ for any $\emptyset \subsetneq S \subsetneq V$. This problem can be reduced to the minimum cut problem. We first construct a hypergraph $G' = (V', E', w')$ from G as follows: define $V' = V \cup \{s, t\}$, where s and t are new vertices. For each hyperedge $e \in E$, add a hyperedge $e \cup \{s\}$ of the same weight to E'. Finally, for each $v \in V$, add a hyperedge $\{v, t\}$ of weight $\boldsymbol{x}(v)$. An example of this reduction is shown in Figure 1. For a set of vertices $S \subseteq V$, the cut weight of $S \cup \{s\}$ in G' is

$$\nu(V) + \boldsymbol{x}(S) - \nu(S).$$

Hence, it suffices to check that the cut weight of $S \cup \{s\}$ is at least $\nu(V) + \epsilon$ for all $\emptyset \subsetneq S \subsetneq V$.

This problem is almost equivalent to finding a minimum s-t cut in G', except that we need to exclude trivial cuts $\{s\}$ and $V \cup \{s\}$. To this end, for each pair of vertices (u, v), we construct a hypergraph G'_{uv} by contracting u to s and v to t. Note that for any $S \subseteq V \setminus \{u, v\}$, the cut weight of $S \cup \{s\}$ in G'_{uv} is equal to the cut weight of $S \cup \{u, s\}$ in G'. Hence, by taking the minimum cut weight of G'_{uv} over all possible pairs (u, v) and checking that it is at least $\nu(V) + \epsilon$, we can find the constraint violated by the current LP solution $(\epsilon, \boldsymbol{x})$.

The algorithm proposed by (Pistorius and Minoux 2003) can be used to compute the minimum cut of a hypergraph.

Algorithm 1

Input: A hypergraph $G = (V, E, w)$.
Output: The strong least core value of G.

```
Solve LP (1) using the ellipsoid method with SEPARATION-ORACLE, and return −ε.
procedure SEPARATION-ORACLE(ε, x)
    if x(V) ≠ ν(V) or x(v) < 0 for some v ∈ V then
        return the violated constraint.
    Construct a hypergraph G' from G.
    for each u, v ∈ V do
        S ← the minimum cut of G'_uv.
        if c(S) < ν(V) + ε then
            return the violated constraint.
    return by saying no constraint is violated.
```

The overall algorithm is given in Algorithm 1. In summary, we have the following:

Theorem 7. *We can compute a value division in the strong and weak least cores of a hypergraph $G = (V, E, w)$ in $O(|V|^2 f(|V|, O(\sum_{e \in E} |e|))L)$ time, where $f(n, m)$ is the time complexity of solving the max flow problem on a graph consisting of n vertices and m edges, and L is the number of times that the separation oracle is called by the ellipsoid method, which is polynomial in $|V|$ and $|E|$.*

Algorithm for the weak least core

In this subsection, we briefly discuss a polynomial-time algorithm that computes a value division in the weak least core of the induced subgraph game associated with a hypergraph $G = (V, E, w)$.

The algorithm is almost identical to that for the strong least core. The only difference is the implementation of the separation oracle, which can be summarized as follows: (i) Check that the current solution satisfies $\boldsymbol{x}(v) \geq \epsilon$ for all $v \in V$ instead of checking $\boldsymbol{x}(v) \geq 0$. (ii) When constructing an auxiliary graph the weight of the hyperedge $\{v, t\}$ is $\boldsymbol{x}(v) - \epsilon$ instead of $\boldsymbol{x}(v)$. See Figure 1 for an example of the reduction. (iii) The cut weight of $S \cup \{s\}$ is then

$$\nu(V) + \boldsymbol{x}(S) - \epsilon|S| - \nu(S).$$

Hence, it suffices to check that the cut weight of $S \cup \{s\}$ is at least $\nu(V)$ for all $\emptyset \subsetneq S \subsetneq V$.

Airport Game

In this section, we consider the airport game. An instance of the airport game is a tuple $I = (n, c_1, \dots, c_n)$, where n is an integer and $c_1, \dots, c_n \in \mathbb{R}_+$ with $c_1 \le c_2 \le \dots \le c_n$. Let $V = [n]$. We define

$$\nu(S) = -\max_{i \in S} c_i$$

for each set $S \subseteq V$. Then, the cooperative game associated with I is (V, ν). In the example of sharing taxis, c_i denotes the fare of the i-th customer, and the fare for a group $S \subseteq V$ is $\max_{i \in S} c_i$, that is, the fare for the customer whose destination is farthest away.

Characterization of the strong least core

We derive a closed formula for the strong least core value of the airport game.

Theorem 8. *The strong least core value of the airport game associated with an instance $I = (n, c_1, \dots, c_n)$ is*

$$-\min_{k \in [n-1]} \frac{c_k}{k+1}.$$

Proof. From Theorem 2, the strong least core value is

$$-\min\Big\{ \min_{\mathcal{S} \in \mathcal{P}_2} \frac{1}{|\mathcal{S}|}\Big(\nu(V) - \sum_{S \in \mathcal{S}} \nu(S)\Big),$$
$$\min_{\mathcal{S} \in \bar{\mathcal{P}}_2} \frac{1}{|\mathcal{S}|}\Big((|\mathcal{S}|-1)\nu(V) - \sum_{S \in \mathcal{S}} \nu(S)\Big)\Big\}.$$

The first term is

$$\min_{\mathcal{S} \in \mathcal{P}_2} \frac{1}{|\mathcal{S}|}\Big(\sum_{S \in \mathcal{S}} \max_{i \in S} c_i - c_n\Big) = \min_{2 \le k \le n} \frac{1}{k} \sum_{i \in [k-1]} c_i.$$

The second term is

$$\min_{\mathcal{S} \in \bar{\mathcal{P}}_2} \frac{1}{|\mathcal{S}|}\Big(\sum_{S \in \mathcal{S}} \max_{i \in S} c_i - (|\mathcal{S}|-1)c_n\Big)$$
$$= \min_{2 \le k \le n} \frac{1}{k}\Big((k-1)c_n + c_{k-1} - (k-1)c_n\Big) = \min_{2 \le k \le n} \frac{c_{k-1}}{k}.$$

Since $c_{k-1} \le \sum_{i \in [k-1]} c_i$, we have the desired result. □

Theorem 9. *The strong least core value $-\epsilon$, where $\epsilon := \min_{k \in [n-1]} \frac{c_k}{k+1}$ of the airport game associated with an instance $I = (n, c_1, \dots, c_n)$ is achieved by the following value division $\boldsymbol{x} \in \mathbb{R}^V$:*

$$\boldsymbol{x}(i) = -\epsilon \ (i \in [n-1]) \text{ and } \boldsymbol{x}(n) = -c_n + \epsilon(n-1).$$

Proof. First, we have $\boldsymbol{x}([n]) = -c_n = \nu([n])$. In the following, we check that $\boldsymbol{x}(S) \ge \nu(S) + \epsilon$ holds for any $\emptyset \subsetneq S \subsetneq V$.

Note that $\boldsymbol{x}(i) = -\epsilon \le 0$ for all $i \in [n-1]$ and $\boldsymbol{x}(n) = -c_n + \epsilon(n-1) \le c_{n-2} - c_n \le 0$. Hence, we need only consider sets of the form $[k]$ for each $k \in [n-1]$ and $[n] \setminus \{k\}$ for each $k \in [n-1]$. Specifically, we check the following conditions: (i) $\boldsymbol{x}([k]) \ge \nu([k]) + \epsilon$ for each $k \in [n-1]$, and (ii) $\boldsymbol{x}([n]) - \boldsymbol{x}(k) \ge \nu([n]) + \epsilon$ for each $k \in [n-1]$. Since $\boldsymbol{x}([k]) = -\epsilon k$ and $\nu([k]) = -c_k$, the former condition is equivalent to $\epsilon(k+1) \le c_k$ for each $k \in [n-1]$, which is true. Since $x([n]) = \nu([n])$ and $\boldsymbol{x}(k) = -\epsilon$, the latter condition is trivially true. □

Characterization of the weak least core

We derive a closed formula for the weak least core value of the airport game.

Theorem 10. *The weak least core value of the airport game associated with an instance $I = (n, c_1, \dots, c_n)$ is*

$$-\frac{1}{n} \min_{k \in [n-1]} \frac{c_k}{k}.$$

Proof. From Theorem 3, the weak least core value is

$$-\frac{1}{n} \min_{\mathcal{S} \in \bar{\mathcal{P}}_2} \frac{1}{|\mathcal{S}|-1}\Big((|\mathcal{S}|-1)\nu(V) - \sum_{S \in \mathcal{S}} \nu(S)\Big),$$

which is

$$-\frac{1}{n} \min_{\mathcal{S} \in \bar{\mathcal{P}}_2} \frac{1}{|\mathcal{S}|-1}\Big(\sum_{S \in \mathcal{S}} \max_{i \in S} c_i - (|\mathcal{S}|-1)c_n\Big)$$
$$= -\frac{1}{n} \min_{2 \le k \le n} \frac{1}{k-1}\Big((k-1)c_n + c_{k-1} - (k-1)c_n\Big)$$
$$= -\frac{1}{n} \min_{2 \le k \le n} \frac{c_{k-1}}{k-1} = -\frac{1}{n} \min_{k \in [n-1]} \frac{c_k}{k}.$$ □

Theorem 11. *The weak least core value $-\epsilon$, where $\epsilon := \frac{1}{n}\min_{k \in [n-1]} \frac{c_k}{k}$ of an instance $I = (n, c_1, \dots, c_n)$ of the airport game is achieved by the following value division $\boldsymbol{x} \in \mathbb{R}^V$:*

$$\boldsymbol{x}(i) = -\epsilon(n-1) \quad \forall i \in [n-1],$$
$$\boldsymbol{x}(n) = -c_n + \epsilon(n-1)^2.$$

Proof. Note that $\boldsymbol{x}([n]) = -c_n = \nu([n])$. In the following, we check that $\boldsymbol{x}(S) \ge \nu(S) + \epsilon|S|$ holds for any $\emptyset \subsetneq S \subsetneq V$.

Let $\emptyset \subsetneq S \subsetneq V$ be a set. If $n \notin S$, then we have that $\boldsymbol{x}(S) - \nu(S) = c_j - \epsilon(n-1)|S| \ge c_j - \frac{c_j}{j}|S| + \epsilon|S| \ge \epsilon|S|$ for $j = \max\{i \in S\}$. If $n \in S$, then we have $\boldsymbol{x}(S) - \nu(S) = \epsilon(n-1)\big((n-1) - (|S|-1)\big) = \epsilon(n-1)(n-|S|) \ge \epsilon|S|$. □

Conclusion

The contributions reported in this paper can be summarized as follows: First, we provided theoretical characterizations of the strong and weak least core values of a supermodular game. We then derived explicit and concise formulations for the strong and weak least core values of the induced subgraph game and the airport game, and presented polynomial-time algorithms for computing the value divisions in the strong and weak least cores. It should be noted that, although space limitations mean that we have only provided formulations for two supermodular games, we could easily analyze other supermodular games using a similar reasoning.

Acknowledgments D. H. is supported by JSPS Grant-in-Aid for Young Scientists (B) (No. 15K16056). Y. Y. is supported by JSPS Grant-in-Aid for Young Scientists (B) (No. 26730009), MEXT Grant-in-Aid for Scientific Research on Innovative Areas (No. 24106003), and JST, ERATO, Kawarabayashi Large Graph Project.

References

Bachrach, Y.; Elkind, E.; Meir, R.; Pasechnik, D.; Zuckerman, M.; Rothe, J.; and Rosenschein, J. S. 2009. The cost of stability in coalitional games. In *Algorithmic Game Theory*. Berlin, Heidelberg: Springer Berlin Heidelberg. 122–134.

Chalkiadakis, G.; Elkind, E.; and Wooldridge, M. 2011. Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 5.

Cunningham, W. H. 1985. Optimal attack and reinforcement of a network. *Journal of the ACM* 32(3):549–561.

Deng, X., and Papadimitriou, C. H. 1994. On the complexity of cooperative solution concepts. *Mathematics of Operations Research* 19(2):257–266.

Faigle, U.; Kern, W.; and Kuipers, J. 2001. On the computation of the nucleolus of a cooperative game. *International Journal of Game Theory* 30(1):79–98.

Feigenbaum, J.; Papadimitriou, C. H.; and Shenker, S. 2001. Sharing the cost of multicast transmissions. *Journal of computer and system sciences* 63(1):21–41.

Frank, A., and Tardos, É. 1988. Generalized polymatroids and submodular flows. *Mathematical Programming* 42(1-3):489–563.

Fujishige, S. 1984. Structures of polyhedra determined by submodular functions on crossing families. *Mathematical Programming* 29(2):125–141.

Fujishige, S. 2005. *Submodular Functions and Optimization*, volume 58 of *Annals of Discrete Mathematics*. Elsevier, 2nd edition.

Gillies, D. B. 1959. Solutions to general non-zero-sum games. *Contributions to the Theory of Games* 4:47–85.

Graham, D. A.; Marshall, R. C.; and Richard, J.-F. 1990. Differential payments within a bidder coalition and the Shapley value. *The American Economic Review* 80:493–510.

Hirayama, K.; Hanada, K.; Ueda, S.; Yokoo, M.; and Iwasaki, A. 2014. Computing a payoff division in the least core for mc-nets coalitional games. In *PRIMA*, 319–332.

Ieong, S., and Shoham, Y. 2005. Marginal contribution nets: a compact representation scheme for coalitional games. In *EC*, 193–202. ACM.

Khachiyan, L. G. 1980. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics* 20(1):53–72.

Kuipers, J. 1996. *A Polynomial Time Algorithm for Computing the Nucleolus of Convex Games*. Reports in operations research and systems theory. Maastricht University.

Li, Y., and Conitzer, V. 2015. Cooperative game solution concepts that maximize stability under noise. *AAAI* 979–985.

Littlechild, S. C., and Owen, G. 1973. A simple expression for the shapley value in a special case. *Management Science* 20(3):370–372.

Naitoh, T., and Fujishige, S. 1992. A note on the Frank—Tardos bi-truncation algorithm for crossing-submodular functions. *Mathematical Programming* 53(1-3):361–363.

Oishi, T., and Nakayama, M. 2009. Anti-dual of economic coalitional tu games. *The Japanese Economic Review* 60(4):560–566.

O'Neill, B. 1982. A problem of rights arbitration from the talmud. *Mathematical Social Sciences* 2(4):345–371.

Pistorius, J., and Minoux, M. 2003. An improved direct labeling method for the max–flow min–cut computation in large hypergraphs and applications. *International Transactions in Operational Research* 10(1):1–11.

Reisch, L. A., and Thgersen, J. 2015. *Handbook of Research on Sustainable Consumption*. Cheltenham, UK: Edward Elgar Publishing, Inc.

Shapley, L. S., and Shubik, M. 1966. Quasi-cores in a monetary economy with nonconvex preferences. *Econometrica* 34(4):805.

Shapley, L. S. 1967. On balanced sets and cores. *Naval research logistics quarterly* 14(4):453–460.

Shapley, L. S. 1971. Cores of convex games. *International Journal of Game Theory* 1(1):11–26.