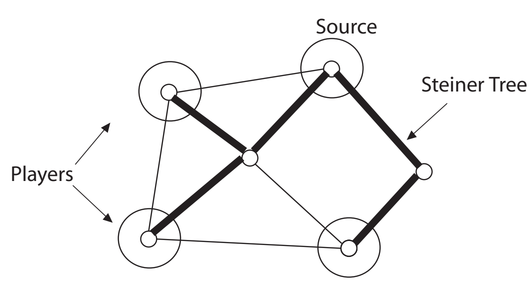
CS 573: Algorithmic Game Theory	Lecture date: April 16, 2008
Instructor: Chandra Chekuri	Scribe: Ali Farhadi

Contents

1 Cooperative Game Theory

Strategic games model intuitions which individual agents selfishly maximize their utility and there is generally no cooperation. In cooperative/coaliteral games the goal is to model situations where the players either benefit by working together or sharing some cost, but they are still selfish in that they will cooperate only if it benefits them. Goal of cooperative game theory is to understand mechanisms/allocations in which players cooperate.

Example: Multicast Game



Graph $G = (V, E)$ represents the network with a source vertex $s \in V$. N is the set of players interested in sharing some common video from the source. Each player $i \in N$ is at some vertex $v_i \in V$. Goal of service provider is to build a multicast tree to connect s to N. Service provider pays to create this tree and charges payments to players. Multicasting is sending information that is shared by several people. c_e is the cost of purchasing bandwidth on the link e.

Requirements:

- Payments of players should pay for the cost of tree.
- For every $S \subseteq N$ (coalition) there is no tree that S can build by itself such that cost of this tree is less than the total payments of the players in S. (otherwise S will threaten to boycott and build its own tree).

Question: Can the above two goals be achieved?
More general setting in which each player may have utility for receiving service and we need a mechanism to not only decide the payments but also which ones to connect/offer service to, such that the players have no incentive to lie even by colluding "group strategy proof" mechanisms.

1.1 Cooperative Games

Cooperative games are most generally defined as non-transferable utility (NTU) games.

Definition 1.1 (NTU Games) *A cooperative NTU game consists of a finite set of players N and a function v that assigns to each coalition $S \subseteq N$ a set of outcomes $v(S) \subseteq R^{|N|}$.*

The interpretation of the above definition is that if $\bar{x} \in v(S)$ then utility of i when S breaks off as a coalition is x_i. The definition is fairly broad. A special case which is of more interest to us will be transferable utility (TU) games. In this setting, we do not consider the relation between coalitions. This setting models these two facts: (1) Is there any coalition which is unhappy? (2) Can we come to some point that no coalition want to deviate.

Definition 1.2 (TU Games) *A cooperative TU games consists of a (1) a finite set, N, of n players (2) a value function $v : 2^n \rightarrow R$, i.e. $\forall S \subseteq N$, $v(S)$ is a number that assigns the total utility of S if it forms a coalition and breaks off.*

The team transferable utility refers to the fact that only the total utility of S is specified which means that the breakaway coalition can transfer the gained utility amongst themselves. Inside each coalition we do not consider which player gets benefit and which one not. For example this would be true if money is the measure of utility.

Note that $v(S)$ can be positive or negative. We will assume for the most part that $v(\emptyset) = 0$. When all $v(S) \leq 0$ the game models "costs". Thus, $v(S)$ indicates cost of S to obtain a service.

Definition 1.3 (Cost-Sharing Game) *A cost-sharing game consists of a finite set, N, of n players and a cost function $c : 2^n \rightarrow R^+$ to denote the positive cost to S for obtaining service.*

This means that players are interested in minimizing cost as opposed to maximizing utility.
We will implicitly assume in TU games that:

$$v(N) \geq v(N_1) + v(N_2) + \ldots + v(N_k) \text{ for any partition of } N \text{ into } N_1, \ldots, N_k$$

In the cost case this implies that:

$$c(N) \leq c(N_1) + c(N_2) + \ldots + c(N_k)$$

Otherwise, no reason for N to cooperate as a whole in the first place.

1.2 Core of the Game

A key concept in cooperative game theory is the *core* of the game.

Definition 1.4 (core) *Given a cooperative TU game (N, v) the core of the game is the set of all vectors $\alpha \in R^{|N|}$ that satisfy:*

1. $\sum_{i \in N} \alpha_i = v(N)$
2. $\sum_{i \in S} \alpha_i \geq v(S)\ \forall S \subseteq N$

Definition 1.5 *For a cost-sharing game (N, c) the core of the game is the set of all $\alpha \in R^{|N|}$ such that:*

1. **Budget balance:** $\sum_{i \in N} \alpha_i = c(N)$
2. **Core property:** $\sum_{i \in S} \alpha_i \leq c(S)$ *for all* $S \subseteq N$

For cost-sharing the interpretation of a core vector α is that α_i is the payment of i. We want $\alpha(N) = c(N)$ so that cost of providing service to N is paid for by the payments and also that total payments for a set S, $\alpha(S)$ does not exceed $c(S)$, otherwise S will deviate and pay for its service separately. In other words, core is the set of allocations that are *stable* allocations.

Example: Three players majority game

There are 3 players: $N = \{1, 2, 3\}$.

$v(N) = 1$

$v(S) = a$ if $|S| = 2$

$v(S) = 0$ if $|S| = 1$

If $a > 2/3$ core is empty: summing over every 2 elements of vectors in the core should be greater than 2/3, thus the summation of all the elements in the core vectors would be greater than 1.

If $a \leq 2/3$ core is non-empty: $(1/3, 1/3, 1/3)$ is a vector in the core.

Example: Treasure Carryout

An expedition N of n people has found a treasure. Each piece of value 1 can be carried out by 2 people. Then, we can model the pay off as:

$$v(S) = \begin{cases} \frac{|S|}{2} \text{if } |S| \text{ is even} \\ \frac{|S|-1}{2} \text{if } |S| \text{ is odd} \end{cases} \tag{1}$$

If $n \geq 4$ is even then core consists of the single pay off vector $(\frac{1}{2}, \ldots, \frac{1}{2})$.

If $n \geq 3$ is odd, then core is empty. Assume there is a core $\bar{\alpha} = (\alpha_1, \ldots, \alpha_n)$. The elements in the core vector should satisfy these properties:

$$v(N) = \alpha_1 + \ldots + \alpha_n = \frac{n-1}{2}$$

For any $S \subseteq N$, where $|S| = n - 1$ we have the following equations:

$$\alpha_1 + \ldots + \alpha_{n-1} \geq V(S) = \frac{n-1}{2} \Rightarrow \alpha_n \leq 0$$

$$\vdots$$

$$\alpha_2 + \ldots + \alpha_n \geq V(S) = \frac{n-1}{2} \Rightarrow \alpha_1 \leq 0$$

Thus, all the elements in the core should be negative, which is not possible because $\alpha_1 + \ldots + \alpha_n = \frac{n-1}{2} > 0$, where $n \geq 3$.
If $n = 2$ core is not unique and consists of all the vectors (a, b) such that $a + b \leq 1$, $a \geq 0, b \geq 0$.
Example: Submodular Costs/ Submodular benefits
Suppose v is submodular:

$$v(S) + v(T) \leq v(S \cup T) + v(S \cap T) \quad \forall S, T \subseteq N$$

Order players arbitrarily as $1, \ldots, n$. Let $x_i = v(S_{i+1}) - v(S_i)$ where $S_i = \{1, \ldots, i\}$ x is in core, we leave the proof as an exercise.
In the cost-sharing world the above setting translates to:

$$c(S) + c(T) \geq c(S \cup T) + c(S \cap T) \quad \forall S, T \subseteq N$$

Which is the same as:

$$c(S + i) - c(S) \geq c(T + i) - c(T) \quad \forall S \supseteq T$$

One can interpret $c(S + i) - c(S)$ as the cost of adding i to the set S. Then core is non-empty.

In the multicast setting if we set $c(S)$ to be the cost of an optimal Steiner tree connecting S to the root, then c is not necessarily sobmodular. However, if T is a fixed tree on all of N and $c(S)$ is the cost of the subtree of T that spans S, then it is submodular.

1.2.1 Characterizing the Core

We now characterize when the core of a cost-sharing TV game is non-empty. One can see that $\bar{\alpha}$ is in the core if and only if:

$$\begin{aligned} \alpha(N) &= c(N) \\ \alpha(S) &\leq c(S), \ \forall S \subseteq N \end{aligned}$$

Thus, the core is the set of all feasible solutions to the above linear set of inequalities.

Corollary 1.6 *Core is convex.*

We recast the above into a linear program to get further insight.

$$\begin{aligned} \max \quad & \sum_{i \in N} \alpha_i \\ & \sum_{i \in S} \alpha_i \leq c(S) \ \forall S \subseteq N \end{aligned}$$

Core is non-empty iff optimum value of the above linear program is equal to $c(N)$. We write the dual of the above linear program:

$$\begin{aligned} \min \quad & \sum_{S \subseteq N} c(S) \lambda_S \\ & \sum_{S \ni i} \lambda_S = 1 \ \forall i \\ & \lambda_S \geq 0 \ \forall S \subseteq N \end{aligned} \tag{2}$$

Definition 1.7 *A set of weights λ_S, $S \subseteq N$ is balanced iff $\sum_{S \ni i} \lambda_S = 1$ for every $i \in N$.*

Proposition 1.8 *Core is non-empty iff for every balanced set of weights λ_S, $\sum_{S \ni i} \lambda_S c(S) \geq c(N)$.*

This follows from LP-duality.

For many problems of interest the core is empty. For example, if $c(S)$ is the cost of Steiner tree to connect S to root then there are graphs in which core is empty. In addition deciding whether core is empty or not is in general NP-hard, for example Steiner tree problem. Sometimes, the input is not given explicitly. For example, for Steiner tree problem we are just given the graph. In these cases finding out if the core is empty or not is also NP-hard.

1.2.2 Approximate Core

As you see in the previous section, core can be empty in many cases. Therefore, we define the notion of approximate core in cooperative games.

Definition 1.9 *A vertex α is in the γ-core of a game if:*

- $\alpha(S) \leq c(S)\ \forall S \subseteq N$
- $\gamma c(N) \leq \alpha(N) \leq c(N)$ *(called, "Approximately Budget Balanced ")*

where, $\gamma \in [0, 1]$.

For a given combinatorial problem, the goal is to find the largest value γ such that the γ-core is non-empty.

Proposition 1.10 *If the cost function c is subadditive (i.e., $c(S_1 \cup S_2) \leq c(S_1) + c(S_2)$ for two disjoint sets S_1 and S_2) then the largest γ for which the γ-core of a game (N, c) is non-empty is precisely the integrality gap of the dual[1] (Equation 2).*

The above follows from the fact that in the dual (Equation 2), if we allow integer values (i.e., $\lambda_S \in \{0, 1\}$), then because of subadditivity of c the minimum would be $c(N)$.
For multicast problem even though core is empty, 1/2-core is not empty as the integrality gap of the dual (Equation 2) is 2.
Core for NTU games NTU games are much more complicated. There are necessary condition for the existence of a non-empty core but not sufficient conditions. The core can be disconnected (see the text book).

1.3 Group Strategy Proof Mechanism and Cost Sharing Schemes

In cost-sharing TU games we assume that all players N wish to get a service and seek payments in the (approximate) core. However, in general the players may have individual utilities and are price sensitive. They may not wish to participate if price is too high. However, payment may depend on other players. We therefore seek a mechanism that elicits truthful bids from the players to decide who to provide service to and how much to change them. Unlike the strategic game setting, we here also want to protect against coalition formation. This is a difficult requirement and hence we can show the existence of such mechanisms only in a limited setting. The ideas of cost sharing help and need to be generalized.

[1] Integrality gap is the maximum ratio between the solution quality of the integer program and of its relaxation