Cooperative Games
Lecture 9: Representation and Complexity issues
Stéphane Airiau

BLC - University of Amsterdam

Representation by enumeration

- Let us assume we want to write a program for computing a solution concept.
- How do we represent the input of a TU game?
- Straightforward representation by enumeration requires exponential space.
- Brute force approach may appear good as complexity is measured in term of the input size.
- We need compact or succinct representation of coalitional games.
- E.g., a representation so that the input size is a polynomial in the number of agents.
- In general, the more succinct a representation is, the harder it is to compute, hence we look for a balance between succinctness and tractability.

Proposition
Let \((V,W)\) be a induced subgraph game. If all the weights are nonnegative then the game is convex.

Proof
\[
\pi(S) + \pi(T) = \sum_{(i,j) \in S} w_{ij} + \sum_{(i,j) \in T} w_{ij} = \sum_{(i,j) \in S \cup T} w_{ij} + \sum_{(i,j) \notin S \cup T} w_{ij} \leq \sum_{(i,j) \notin S \cup T} w_{ij} = \pi(S \cap T) + \pi(S \setminus T) + \pi(T \setminus S) 
\]

Proposition
Let \((V,W)\) be an induced subgraph game. If all the weights are nonnegative then membership of a payoff vector in the core can be tested in polynomial time.

Others results

- Induced subgraph games may not be monotone (because of negative weights).
- Efficient network-flow based algorithm for checking whether an outcome is in the core (when weights are positive) \([\text{Deng} \& \text{Papadimitriou}, 94]\)
- If weights can be negative, checking whether the core is empty is \(\text{NP}\)-complete.
- Checking whether an outcome is in the core is \(\text{coNP}\)-complete \([\text{Deng} \& \text{Papadimitriou}, 94]\)
- Checking whether an outcome is in the kernel is \(\mathcal{K}_2\)-complete.
- In the bargaining set is \(\mathcal{T}_2\)-complete \([\text{Greco} \ et \ al., 2011]\)

Induced subgraph games

An induced subgraph game is a coalitional game defined by an undirected weighted graph \((G = (V,E))\) where \(V\) is the set of vertices and \(W : V \rightarrow \mathbb{R}\) is the set of edges weights. For \((i,j) \in E\), \(w_{ij}\) is the weight of the edge between \(i\) and \(j\).

- \(N = V\), i.e., each agent is a node in the graph.
- for all \(\emptyset \subset C \subseteq N\), \(\pi(C) = \sum_{(i,j) \in E(C)} w_{ij}\)

It is a succinct representation: using an adjacency matrix, we need to provide \(n^2\) entries. However, it is not complete. Some TU games cannot be represented by an induced subgraph game (e.g., a majority voting game).

Theorem
Let \((V,W)\) be an induced subgraph game. The Shapley value of an agent \(i \in V\) is \(\text{Sh}_i(N,v) = \frac{1}{|N|^2} \sum_{E(C) \subseteq N} w_{ij}\). The Shapley value can be computed in \(O(|N|^3)\) time.

Proof
Let \((V,W)\) be an induced subgraph game. We can view this game as the sum of the following \(|E|\) games (i.e., one game per edge): \(G_{ij} = (V,|w_{ij}|,\{(i,j) \in E\})\). For a game \(G_{ij}\), \(i\) and \(j\) are substitutes, and all other agents \(k \neq i, j\) are dummy agents. Using the symmetry axiom, \(\text{Sh}_i(G_{ij}) = \text{Sh}_j(G_{ij})\). Using the dummy axiom, \(\text{Sh}_k(G_{ij}) = 0\). Hence, \(\text{Sh}_i(G) = \frac{1}{|N|^2} \sum_{E(C) \subseteq N} w_{ij}\).

Since \((V,W)\) is the sum of all two-player games, by the additivity axiom, \(\text{Sh}_i = \sum_{E(C) \subseteq N} \text{Sh}_i(G_{ij}) = \sum_{(i,j) \in E} w_{ij}\)

Network flow games

- Players are edges of a network with a source \(s\) and a sink \(t\).
- Each edge \(e\) has a capacity \(c_e \in \mathbb{R}\), indicating how much it can carry.
- The value of a colition \(C \subseteq N\) is the value of the maximum amount of flow that can be sent from \(s\) to \(t\) using only the edges in \(C\).

Solution concepts studied by Granot and Granot 92. Computing nucleolus is easy when weights are one, hard otherwise [Deng 09]

Variant: can the network carry a flow of at least \(k\)? [Bachrach et al. 09]

Example
Let \((V,W)\) be an induced subgraph game. Let \(\pi(S) + \pi(T)\) be a polynomial in the number of agents.

\[
\pi(S) + \pi(T) = \sum_{(i,j) \in S} w_{ij} + \sum_{(i,j) \in T} w_{ij} + \sum_{(i,j) \in S \setminus T} w_{ij} + \sum_{(i,j) \notin S \cup T} w_{ij} \leq \sum_{(i,j) \notin S \cup T} w_{ij} = \pi(S \cap T) + \pi(S \setminus T) + \pi(T \setminus S) 
\]

\begin{itemize}
  \item Can we efficiently compute a solution?
  \item Compact representation
\end{iteminstance}
A representation for superadditive games

Instead of storing a value for each coalition, we can store the positive synergies between agents.
Let \((N,v)\) be a superadditive game and \((N,S)\) its synergy representation. Then for a coalition \(C \subseteq N\),
\[
v(C) = \left( \max_{\{\ell \in \mathcal{E} | \ell \subseteq C\}} \sum_{\ell \subseteq C} v(\ell) \right),
\]
where \(\mathcal{E}\) is the set of all partition of \(\ell\).

Example: \(N = \{1,2,3\}\), \(v(\{\}) = 1\), \(v(\{1\}) = 3\), \(v(\{1,3\}) = 2\), \(v(\{2,3\}) = 2\), \(v(\{1,2,3\}) = 4\).
We can represent this game by \(v(\{\}) = 1\), \(v(\{1\}) = 3\).
This representation may still require a space exponential in \(v\).

The idea:
Let \(v\) be a superadditive game and \(N\) the number of agents, but for many games, the space required is much less.

A logical approach: Marginal contribution nets (MC-nets)

The idea:
- represent each player by a boolean variable,
- treat the characteristic vector of a coalition as a truth assignment.
- the truth assignment can be used to check whether a formula is satisfied and to compute the value of a coalition.

Let \(N\) be a collection of atomic variables.

Definition (Rule)
A rule has a syntactic form \(\phi(w)\) where \(\phi\) is called the pattern and \(w\) is a boolean formula containing variables in \(N\) and \(\phi\) is called the weight, and \(w\) is a real number.

Examples:
\(a \land b; 5\): each coalition containing both agents \(a\) and \(b\) increase its value by 5 units.
\(b; 2\): each coalition containing \(b\) increase its value by 2.

Theorem (Expressivity)
- MC-nets can represent any game when negative literals are allowed in the patterns, or when the weights can be negative.
- When the patterns are limited to conjunctive formula over positive literals and weights are nonnegative, MC-nets can represent all and only convex games.

Proposition
MC-nets generalize the induced subgraph game representation (strict generalization) and the multi-issue representation.
**Theorem**

Given a TU game represented by an MC-net limited to conjunctive patterns, the Shapley value can be computed in time linear in the size of the input.

**Proof sketch:** we can treat each rule as a game, compute the Shapley value for the rule, and use ADD to sum all the values for the overall game. For a rule, we cannot distinguish the contribution of each agent, by SYM, they must have the same value. It is a bit more complicated when negation occurs (see Ieong and Shoham, 2005).

**Proposition**

Determining whether the core is empty or checking whether an imputation lies in the core are coNP-hard.

**Proof sketch:** due to the fact that MC-nets generalize over weighted graph games.

---

**Summary**

- We saw some representations that are more succinct or that may help to compute faster some solution concepts.
- representation for specific games (not complete): induced subgraph game, superadditive representation
- general representations, that may require less space in some cases (multi-issue, MC-nets)
- Computing some solution concepts become easy in some case (Shapley value with WGG, Multi-issue, MC nets; empty core for superadditive representation).

---

**Coming next**

- NTU games