Editor: **Michael Wooldridge,** University of Oxford, mjw@cs.ox.ac.uk

Cooperative Game Theory: Basic Concepts and Computational Challenges

Georgios Chalkiadakis, *Technical University of Crete*
Edith Elkind, *Nanyang Technological University*
Michael Wooldridge, *University of Oxford*

Game theory is concerned with decision-making in strategic settings, where you must factor the preferences and rational choices of other players into your decision to make the best choice for yourself. In many such settings, you're on your own: the choice you must make is yours and yours alone, because cooperation with other players is either impossible to implement or without any possible benefits. However, in some situations it is both possible and fruitful to cooperate with other players. Where players can make binding agreements with each other, and there is some added value available by cooperating with others, then it can make sense for players to form coalitions that will work together to mutual advantage. Formal legal contracts are the most obvious mechanism available in the real world for implementing binding agreements. The field of cooperative game theory studies strategic decision-making in settings where binding agreements are possible and where agents can therefore act collectively. In this article, our aims are twofold: first, to give a brief introduction to the main concepts of cooperative game theory; and second, to describe some of the issues that arise when we want to apply these concepts in AI.

Characteristic-Function Games

The most widely studied model of cooperative games is that of *characteristic-function games*. This surprisingly simple model turns out to be rich enough to capture the properties of many cooperative scenarios. A characteristic-function game is given by a pair (N, v), where $N = \{1, \ldots, n\}$ is the set of players in the game, and $v : 2^N \rightarrow \mathbf{R}$ is a function that gives the real-numbered *value*, $v(C)$, of every set of players $C \subseteq N$. The function v is called the *characteristic function* of the game. The idea is that $v(C)$ is the amount that the coalition C could earn should they choose to cooperate. The model doesn't specify exactly how they earn this value, or indeed what "cooperate" means. These two components are assumed to be the only information that players in the game have. Given this information, there are two fundamental questions that cooperative game theory considers:

- Who will cooperate with whom—that is, which coalitions will form?
- After coalitions have formed and earned the value defined by the characteristic function, how will they divide this value amongst themselves? And in particular, how will they divide the value fairly?

Cooperative game theory suggests that a necessary condition for coalition formation is that the coalition is stable, in the sense that no members of the coalition have any incentive to walk away from it. The best-known solution concept formalizing this idea is the *core*. With respect to the second question, a solution concept known as the *Shapley value* provides a unique way to divide coalitional value among players in such a way as to satisfy various fairness criteria.

If we want to use these models and solution concepts in AI, then two further issues arise:

- How can we compactly represent cooperative games? The issue here is that in practice, we cannot represent a cooperative game by listing every coalition $C \subseteq N$ and its corresponding value $v(C)$, because there are $2^{|N|}$ such coalitions.

1541-1672/12/$31.00 © 2012 IEEE
Published by the IEEE Computer Society

We thus need some representation for the characteristic function v that is of size polynomial in $|N|$.
- How can we efficiently compute solution concepts for cooperative games?

To keep things simple, we will restrict our attention when considering solution concepts to the *grand coalition*: the set of all players, N. We will ask whether the grand coalition is stable and how we can fairly divide its value.

Stable Coalitions

The solution concept known as the core suggests that a necessary condition for the formation of a coalition is that no subset of players within the coalition have any incentive to deviate from it. To understand how the core formalizes this idea, we need a bit of notation. A *payoff vector* is a tuple of real numbers $\mathbf{x} = (x_1, \ldots, x_n)$ that divides the value $v(N)$ among all the players in N; thus x_i is the amount given to player i in this payoff vector. A coalition $C \subseteq N$ objects to the payoff vector $\mathbf{x}$ if they could collectively earn more than $\mathbf{x}$ allocates them—that is, if

$$v(C) > \sum_{i \in C} x_i.$$

If this condition is satisfied, then the payoff vector $\mathbf{x}$ could not be implemented, because C would not accept it; they would do better to work on their own, and could divide the surplus obtained among themselves. Now, the core of a game (N, v) is the set of payoff vectors to which no coalition has any objection in the sense we just described. If the core is empty, then this means that the coalition consisting of all agents cannot form: there is no way of distributing the value $v(N)$ to which there are no objections. Conversely, if the core of the game is nonempty, then there is some way of distributing the value $v(N)$ to the players in N such that no coalition can reasonably object, in the sense that no coalition could do any better. Thus, the question, "Is the coalition consisting of all agents stable?" reduces to the question, "Is the core of the game nonempty?" In computational terms, we generally want to answer one of two questions relating to the core:

- whether the core of a given cooperative game (N, v) is non-empty (whether the grand coalition is stable), and
- whether a given payoff vector $\mathbf{x} = (x_1, \ldots, x_n)$ is in the core of a given game (N, v).

For both questions, a naïve exhaustive algorithmic approach won't be feasible: for example, in the latter problem, we would need to check whether there exists a coalition $C \subseteq N$ such that the value of C is greater than their cumulative allocation in $\mathbf{x}$. There will, of course, be $2^{|N|}$ such coalitions, and so naïvely considering each possible coalition in turn will not be practicable.

Fair Division Schemes

Suppose the grand coalition N forms, and they then obtain the value $v(N)$. The next question to be answered is how the value $v(N)$ should be divided among the players N. The *Shapley value* provides a principled way to do this. It proposes that each player $i \in N$ should be given an amount φ_i that satisfies the following axioms:

- *Efficiency*. The total value $v(N)$ should be distributed.
- *Dummy player*. Players who make no contribution should receive nothing.
- *Symmetry*. Players who make the same contribution should receive the same.
- *Additivity*. The value should be additive over the set of all games.

While the final property is arguably somewhat technical (and the formal definition is beyond the scope of this article), it is generally accepted that the other properties are easy to motivate from the point of view of fairness. Central to these axioms is the notion of a player's contribution. Shapley argued that we can measure a player's contribution by simply looking at the value the player adds to a coalition. Formally, player i's contribution to a coalition C is simply $v(C \cup \{i\}) - v(C)$, or the amount extra that C could obtain if they admitted player i as a member. If this value is 0, then there is no benefit to be obtained. Given this definition, the symmetry axiom, for example, means that two players should receive the same value if they make the same contribution to all coalitions.

These axioms, however, say nothing about how to compute such a value (or even whether there is any value that satisfies them). In a remarkable result, Shapley showed that there is a unique solution to these axioms. The basic idea is this: imagine all the possible orders in which the grand coalition could form, one player at a time. If $N = \{1, 2, 3\}$ then the grand coalition could form in six possible ways: 1-2-3, 1-3-2, 2-1-3, 2-3-1, 3-1-2, or 3-2-1.

Then, Shapley suggested, a player should receive the average contribution that he or she makes, over each of these orderings, to the set of players that precedes him or her in the ordering. If $N = \{1, 2, 3\}$, for example, then φ_1 is $M/6$, where $M = [v(\{1\}) - v(\emptyset)] + [v(\{1\}) - v(\emptyset)] + [v(\{2, 1\}) - v(\{2\})] + [v(\{2, 3, 1\}) - v(\{2, 3\})] + [v(\{3, 1\}) - v(\{3\})] + [v(\{3, 2, 1\}) - v(\{3, 2\})]$.

Now, if we compute the Shapley values φ_i in this way, the resulting payoff vector $(\varphi_1, \ldots, \varphi_n)$ satisfies the four axioms; and what is more, it's the only payoff vector that satisfies them.

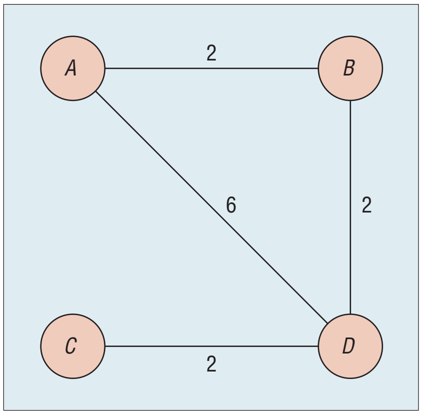

Figure 1. A weighted graph for a four-player game. The lines represent synergies between pairs of players.

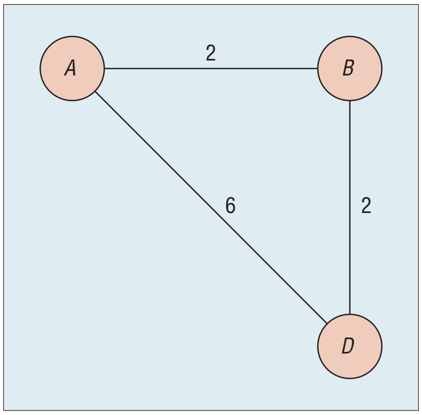

Figure 2. A three-player coalition. Player *C* is not involved.

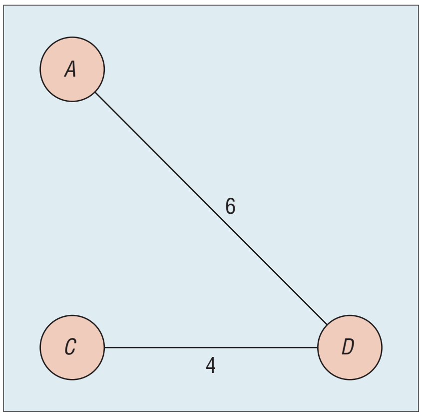

Figure 3. A three-player coalition. Player *B* is not involved.

As this example illustrates, there's some fearsome combinatorial complexity inherent in the Shapley value: if there are n players in the game, we must average over $n!$ orderings of the players. In this example, we have to average over $3! = 6$ orderings; and even in this trivial case the arithmetic starts to become tedious. So, computing the Shapley value by directly averaging a player's contribution over all orderings of the players is not going to be practicable in general.

Compact Representations

Because a naïve representation of cooperative games isn't feasible, much effort has been devoted to developing and investigating compact representations for cooperative games—representations of games that require space at most polynomial in the number of players n. However, a general rule of thumb is that the more compact a representation is, the higher the complexity of the associated computational problems. We therefore seek representations that strike a practical balance among compactness, representational power, and computational tractability. In this section, we examine a few representations that have been proposed in the literature.

Weighted Graph

The first representation was proposed in 1994.[1] The idea is to represent the characteristic function v as a weighted graph, in which vertices correspond to players. For example, suppose we have four players, $N = \{A, B, C, D\}$. Figure 1 represents a characteristic function for such a game.

The idea is that the edges represent synergies between players. To compute the value of a coalition, we add together the weights of all the edges in the subgraph corresponding to that coalition.

For example, suppose we want to compute the value $v(S)$ of the coalition $S = \{A, B, D\}$. First, we take the subgraph induced by this set of vertices: we eliminate from the original graph all vertices not in S, and all the edges that these eliminated vertices are connected to. Figure 2 shows the result. To get the value of coalition $S = \{A, B, D\}$, we add together the weights on all the edges that remain: $v(\{A, B, D\}) = 2 + 6 + 2 = 10$.

Figure 3 shows the subgraph for a different coalition $S = \{A, C, D\}$. The value of this new coalition is $v(\{A, C, D\}) = 6 + 4 = 10$.

This representation is compact: for a game with n players, we only need to record at most n^2 edges and their weights—for example, by using an adjacency matrix. However, the representation is not complete, in the sense that there are cooperative games that the induced subgraph scheme can't represent.

What about computing the solution concepts? It turns out that, for this representation, the problem of checking whether a particular payoff vector is in the core of the game is computationally hard—that is, co-NP–complete. However, computing the Shapley value for this representation is easy. We can divide a graph containing m edges down into m component games, one for each edge. In the original example, there are four edges, and so there are four edge games. We can compute the Shapley value of a player in each edge game, and simply add these together to get the Shapley value of the player in the overall game. The justification for doing so directly follows from one of Shapley's axioms: additivity.

It remains to compute the Shapley value of a player in an edge game. But it is easy to see that the value a player gets from an edge is 0 if the player isn't connected to the edge, and half the weight of the edge if it is connected. This latter fact follows from Shapley's efficiency and symmetry axioms. Thus, the Shapley value of a player in the induced subgraph representation is half the sum of the weights on the edges to which that player is connected, and is hence computable in polynomial time. For the original example graph we gave previously therefore, the Shapley value φ_A of player A is $(2 + 6)/2 = 4$, while the Shapley value φ_D of player D is $(6 + 4 + 2)/2 = 6$, and so on. This result is particularly interesting because it directly appeals to Shapley's axioms in order to compute the Shapley value.

Rule-Based Representations

A similar trick can be used for other additive representations. One such representation is called *marginal-contribution nets* (MC-nets).[2] In the MC-nets representation, we describe the characteristic function via a set of rules, of the form *pattern* ⟼ *value*.

Here, *pattern* is a Boolean condition over players, and *value* is a real number. To compute the value of a coalition S using this representation, we sum up the right-side values over all the rules whose left side is satisfied by S. As an example, consider the following rules:

$A \wedge B \longmapsto 2$

$B \wedge D \longmapsto 7$

$A \wedge B \wedge C \longmapsto 6$

To compute the value $v(S)$ of the coalition $S = \{A, B, D\}$, two rules apply: the first (because the coalition S contains all the agents listed on the left side of the rule) and the second (for the same reason). The third rule doesn't apply because the coalition S doesn't contain the agent C. Thus $v(\{A, B, D\}) = 2 + 7 = 9$. In the same way, the value of the coalition $\{A, B, C\}$ is $2 + 6 = 8$, while the value of the grand coalition $N = \{A, B, C, D\}$ is $2 + 7 + 6 = 15$. (All three rules apply to the grand coalition.)

If we assume that the left sides of rules are in the simple form of the example (conjunctions of players), then we can easily compute the Shapley value using a similar trick as for the weighted-graph representation. We treat each rule as a component game, and the Shapley value of a player in the overall game is the sum of the Shapley values for that player in the component games. To compute the Shapley value in a component game, we again appeal to Shapley's efficiency and symmetry axioms: the players listed on the left side of a rule share the value on the right side equally among themselves. Thus, for example, the Shapley value of player A in the example game is $(2/2) + (6/3) = 1 + 2 = 3$, while the Shapley value of player D is $7/2 = 3.5$. However, core-related problems are hard for MC-nets, as in the weighted-graph representation, because MC-nets are a generalization of the weighted-graph representation. (Every edge in a weighted graph translates to a rule with just two agents on the left side.) Thus, the hardness results for the weighted-graph representation carry over immediately to MC-nets.

Patterns on the left side of rules could in principle be any Boolean condition over the set of players—we aren't restricted to conjunctions of positive literals as in the example rules. If we allow for richer representations, then this representation is complete, in the sense that every cooperative game (N, v) can be represented as a set of rules. This is apparent from the fact that we can have one rule for every possible coalition $C \subseteq N$: the left side of the rule is constructed so that it only matches the coalition C, while the right side is simply the value $v(C)$. However, if we use these richer conditions, it becomes computationally hard to compute the Shapley value. It then becomes an interesting research question to consider how rich we can allow patterns on the left side of rules to be without losing the attractive computational properties of simple conjunctive conditions.

Weighted-Voting Games

Our final representation is particularly interesting because it plays a significant role in our everyday lives. A *weighted-voting game* is a type of simple cooperative game: a game where every coalition either gets the value 0 (they are "losing") or 1 ("winning"). In a weighted-voting game, each player $i \in N$ is associated with a *weight*, w_i, and the overall game has a *quota*, given by a real number q. A coalition $C \subseteq N$ is then said to be winning if the sum of their weights meets or exceeds the quota and losing otherwise:

$$v(C) = \begin{cases} 1 & \text{if } \sum_{i \in C} w_i \geq q \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, weighted-voting games have a compact representation: we just need to represent the weights and overall quota. Computing the Shapley value for weighted-voting games turns out to be NP-hard, but checking whether an outcome is in the core for weighted-voting games is computationally easy.[3]

Weighted-voting games are particularly important because they are widely used in real-world voting scenarios. For example, consider a political voting system, such as the House of Commons in the UK or the Senate in the US. In such settings, the players are the political parties, the weight of a voter corresponds to how many votes that party has (how many seats they hold), and the quota is the number of votes required for a vote to succeed. In such settings, the Shapley value has an interesting interpretation: it measures how much power parties have—their ability to influence the overall decision.

Cooperative game theory is concerned with strategic decision-making in settings where binding agreements are possible. In such situations, coalitions can form to exploit the benefits of cooperation. In this brief article, we hope to have given a flavor of the kinds of models used to study cooperative games, the kinds of solution concepts proposed by cooperative game theory, and some of the issues that arise when we use cooperative game theory in AI and computer science. For further reading, Martin Osborne and Ariel Rubinstein provide a good treatment of cooperative game theory, setting it clearly in the wider context and concerns of game theory.[4] And we have written a graduate-level introduction to cooperative game theory, focusing particularly on the concerns of AI and computer science.[5] ■

References

1. X. Deng and C. Papadimitriou, "On the Complexity of Cooperative Solution Concepts," *Mathematics of Operations Research*, vol. 19, no. 2, 1994, pp. 257–266.
2. S. Ieong and Y. Shoham, "Marginal Contribution Nets: A Compact Representation Scheme for Coalitional Games," *Proc. 6th ACM Conf. Electronic Commerce* (ACM-EC 05), ACM, 2005, pp. 193–202.
3. E. Elkind et al., "On the Computational Complexity of Weighted Voting Games," *Annals Mathematics and Artificial Intelligence*, vol. 56, no. 2, 2009, pp. 109–131.
4. M. Osborne and A. Rubinstein, *A Course in Game Theory*, MIT Press, 1994.
5. G. Chalkiadakis, E. Elkind, and M. Wooldridge, *Computational Aspects of Cooperative Game Theory*, Morgan & Claypool, 2011.

Georgios Chalkiadakis is an assistant professor in the Department of Electronic and Computer Engineering at the Technical University of Crete. Contact him at gehalk@ece.tuc.gr.

Edith Elkind is an assistant professor in the School of Physical and Mathematical Sciences at Nanyang Technological University. Contact her at eelkind@ntu.edu.sg.

Michael Wooldridge is a professor of computer science at the University of Oxford. Contact him at mjw@cs.ox.ac.uk.

cn *Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.*

ADVERTISER INFORMATION • MAY/JUNE 2012

Advertising Personnel

Marian Anderson: Sr. Advertising Coordinator
Email: manderson@computer.org
Phone: +1 714 816 2139 | Fax: +1 714 821 4010

Sandy Brown: Sr. Business Development Mgr.
Email: sbrown@computer.org
Phone: +1 714 816 2144 | Fax: +1 714 821 4010

Advertising Sales Representatives (display)

Central, Northwest, Far East:
Eric Kincaid
Email: e.kincaid@computer.org
Phone: +1 214 673 3742
Fax: +1 888 886 8599

Northeast, Midwest, Europe, Middle East:
Ann & David Schissler
Email: a.schissler@computer.org, d.schissler@computer.org
Phone: +1 508 394 4026
Fax: +1 508 394 1707

Southwest, California:
Mike Hughes
Email: mikehughes@computer.org
Phone: +1 805 529 6790

Southeast:
Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 585 7070
Fax: +1 973 585 7071

Advertising Sales Representatives (Classified Line)

Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 585 7070
Fax: +1 973 585 7071

Advertising Sales Representatives (Jobs Board)

Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 585 7070
Fax: +1 973 585 7071