Chapter 16
Argumentation and Game Theory

Iyad Rahwan and Kate Larson

1 What Game Theory Can Do for Argumentation

In a large class of multi-agent systems, agents are *self-interested* in the sense that each agent is interested only in furthering its individual goals, which may or may not coincide with others' goals. When such agents engage in argument, they would be expected to argue *strategically* in such a way that makes it more likely for their argumentative goals to be achieved. What we mean by arguing strategically is that instead of making arbitrary arguments, an agent would carefully choose its argumentative moves in order to further its own objectives.

The mathematical study of strategic interaction is *Game Theory*, which was pioneered by von Neuman and Morgenstern [13]. A setting of strategic interaction is modelled as a *game*, which consists of a set of players, a set of actions available to them, and a rule that determines the outcome given players' chosen actions. In an argumentation scenario, the set of actions are typically the set of argumentative moves (e.g. asserting a claim or challenging a claim), and the outcome rule is the criterion by which arguments are evaluated (e.g. a judge's attitude or a social norm).

Generally, game theory can be used to achieve two goals:

1. undertake precise analysis of interaction in particular strategic settings, with a view to predicting the outcome;
2. design rules of the game in such a way that self-interested agents behave in some desirable manner (e.g. tell the truth); this is called *mechanism design*;

Both these approaches are quite useful for the study of argumentation in multi-agent systems. On one hand, an agent may use game theory to analyse a given argumentative situation in order to choose the best strategy. On the other hand, we

Iyad Rahwan
British University in Dubai, UAE & University of Edinburgh, UK, e-mail: irahwan@acm.org

Kate Larson
University of Waterloo, Canada e-mail: klarson@cs.uwaterloo.ca

I. Rahwan, G. R. Simari (eds.), *Argumentation in Artificial Intelligence,*
DOI 10.1007/978-0-387-98197-0_16, © Springer Science+Business Media, LLC 2009

may use mechanism design to design the rules (e.g. argumentation protocol) in such a way as to promote good argumentative behaviour. In this chapter, we will discuss some early developments in these directions.

In the next section, we motivate the usefulness of game theory in argumentation using a novel game. After providing a brief background on game theory in Section 3, we introduce our Argumentation Mechanism Design approach in Section 4 and present some preliminary results in Section 5. Finally, we discuss related work in Section 6 and conclude in Section 7

2 The "Argumentative Battle of the Sexes" Game

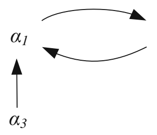
Consider the following situation involving the couple Alice (A) and Brian (B), who want to decide on an activity for the day.[1] Brian thinks they should go to a soccer match (argument α_1) while Alice thinks they should attend the ballet (argument α_2). There is time for only one activity, however (hence α_1 and α_2 defeat one another). Moreover, while Alice prefers the ballet to the soccer, she would still rather go to a soccer match than stay at home. Likewise, Brian prefers the soccer match to the ballet, but also prefers the ballet to staying home. Formally, we can write $u_A(ballet) > u_A(soccer) > u_A(home)$ and $u_B(soccer) > u_B(ballet) > u_B(home)$.

Alice has a strong argument which she may use against going to the soccer, namely by claiming that she is too sick to be outdoors (argument α_3). Brian simply cannot attack this argument (without compromising his marriage at least). Likewise, Brian has an irrefutable argument against the ballet; he could claim that his ex-wife will be there too (argument α_4). Alice cannot stand her! Using Dung's abstract argumentation model [1], which is described in detail in Chapter 2, the argumentative structure of this situation can be modelled as shown in Figure 16.1(a).

Alice can choose to say nothing, utter argument α_2 or α_3 or both. Similarly, Brian can choose to say nothing, utter argument α_1 or α_4 or both. For the sake of the example, we will suppose that Alice and Brian use the grounded semantics as the argumentative foundation of their marriage! The question we are interested in here is: *What will Alice and Bob say?* or at least: *What are they likely to say?*

The strategic encounter, on the other hand, can be modelled as shown in the table in Figure 16.1(b). Each cell corresponds to a strategy profile in which Alice and Brian reveal a particular set of arguments. The numbers in the cells correspond to the utilities they obtain once the grounded extension is calculated on their revealed arguments. For example, if Alice utters $\{\alpha_2\}$ while Brian utters $\{\alpha_1, \alpha_4\}$, we end up with a sub-graph of Figure 16.1(a) in which α_3 is missing. The grounded extension of this argument graph admits arguments $\{\alpha_1, \alpha_4\}$. This corresponds to a situation where Brian wins and the couple head to the soccer. Thus, he gets the highest utility of 2, while Alice gets her second-preferred outcome with utility 1. This representation, shown in Figure 16.1(b), is known as a *normal form game*.

[1] We call this the *argumentative battle of the sexes* game. It is similar, but not identical to the well-known "Battle of the Sexes" game.



α_1: They should go to the soccer.
α_2: They should go to the ballet.
α_3: Alice too sick for the outdoors.
α_4: Brian's ex-wife at the ballet.

(a)

Alice \ Brian	$\{\alpha_1\}$	$\{\alpha_4\}$	$\{\alpha_1, \alpha_4\}$	$\{\}$
$\{\alpha_2\}$	0 0	0 0	1 2	2 1
$\{\alpha_3\}$	0 0	0 0	0 0	0 0
$\{\alpha_2, \alpha_3\}$	2 1	0 0	0 0	2 1
$\{\}$	1 2	0 0	1 2	0 0

Nash Equilibria:
$(\{\alpha_2\}, \{\alpha_1, \alpha_4\})$ $(\{\alpha_2, \alpha_3\}, \{\alpha_1\})$
$(\{\}, \{\alpha_1, \alpha_4\})$ $(\{\alpha_2, \alpha_3\}, \{\})$

(b)

Fig. 16.1 Simple argumentative scenario and its normal form game representation. The outcome is decided using grounded semantics.

The normal-form game can be used to deduce a number of things about this particular scenario. First, it is never in either Alice or Brian's best interest to utter their irrefutable argument (α_3 or α_4) without also stating their preferred activity (α_1 or α_2), since by announcing their preferred activity then they may possibly attend some event (either ballet or soccer) while if one of them only announces their irrefutable argument then both agents are certain to stay at home (the least preferred outcome for both Alice and Brian). That is, $\{\alpha_3\}$ is *weakly dominated* by $\{\alpha_2, \alpha_3\}$ (and $\{\alpha_2\}$) and α_4 is weakly dominated by $\{\alpha_1, \alpha_4\}$ (and $\{\alpha_1\}$).

Given a game, we are interested in finding the Nash equilibria of the game. A Nash equilibrium is a strategy profile (a listing of one strategy for each agent) where no agent wants to change its strategy, assuming that the other agents do not change. The Nash equilibria are the *stable* outcomes of the game. Consider the strategy profile in which Alice says that she is sick and suggests the ballet (i.e. she utters $\{\alpha_2, \alpha_3\}$) and Brian simply suggests the soccer match (i.e. he utters $\{\alpha_1\}$). This outcome is a Nash equilibrium. On one hand, given that Alice states $\{\alpha_2, \alpha_3\}$, Brian has no incentive to deviate to any other strategy. If he mentions his ex-wife's attendance at the ballet (uttering $\{\alpha_4\}$ or $\{\alpha_1, \alpha_4\}$), he shoots himself in the foot and ends up spending the day at home. And if he stays quiet (uttering $\{\}$), he cannot influence the outcome anyway. On the other hand, assuming that Brian announces $\{\alpha_1\}$ then Alice is best-off stating $\{\alpha_2, \alpha_3\}$ since by doing so, she gets the outcome that she prefers (the ballet). In fact, we list four Nash equilibria in Figure 16.1(b).[2]

The analysis that we just concluded does not allow us to identify a *single* outcome for the example. However, it does identify some interesting strategic phenomena. In particular, it shows that it is never in Alice and Brian's interest for them both to use their irrefutable arguments. For example, if Brian is confident that Alice will state

[2] A reader familiar with game theory will note that we only list the pure strategy Nash equilibria. In addition to these four equilibria, there are three mixed equilibria in which players randomize their strategies.

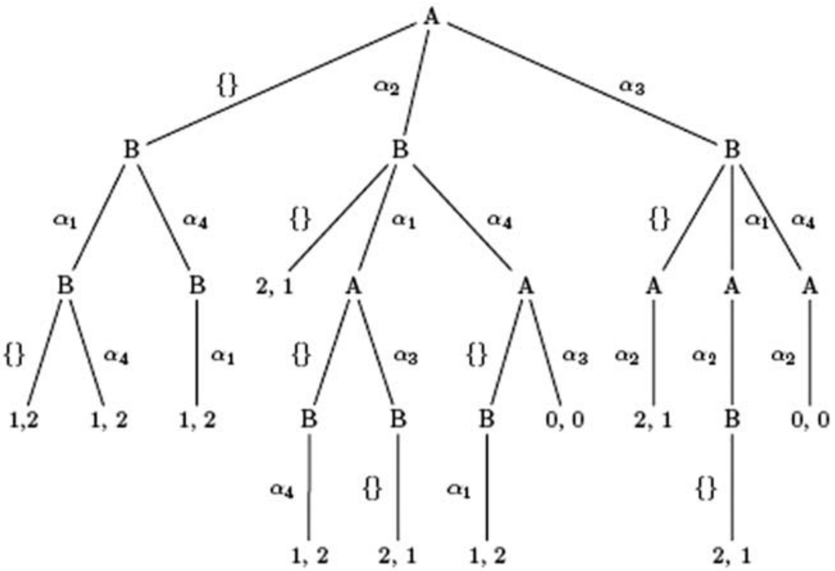


Fig. 16.2 A (pruned) game tree for the argumentative battle of the sexes game.

that she is too sick (α_3) then Brian should not bring up his argument against the ballet.

While the above analysis did not allow us to identify a single outcome of the scenario, at least we were able to rule out so many unstable outcomes. Indeed, in some situations, there is a single Nash equilibrium, which makes predicting the outcome easier.

So far, we used a normal-form representation to model the argument game. While this representation is useful for many purposes due to its simplicity, it fails to capture the *dynamic* aspect of argumentation: the fact that argumentative moves are normally made interactively over multiple time steps. The appropriate tool to model such dynamics are extensive-form games, which we discuss next.

An extensive-form game with perfect information explicitly captures the fact that agents may take turns when choosing actions (for example, declaring arguments). A game tree is used to represent the game. Each node in the tree is associated with an agent whose turn it is to take an action. A path in the tree represents the sequence of actions taken, and leafs nodes are the final outcomes, given the actions on the path to the leaf node. In these games we assume that the actions that each agent takes are fully observable by all the other agents.

We can model the interaction between Alice and Brian using an extensive-form game, if we assume that Alice and Brian take turns uttering arguments. We will assume that i) an agent can only make one argument at a time, ii) agents can not repeat arguments, and iii) if at some step an agent decides not to make an argument, then they are not allowed to make any more arguments in the future. We will also assume, for the sake of the example, that Alice gets to make the first argument. Figure 16.2 shows the game tree for the argumentative battle of the sexes game. Most of the paths which result in an outcome where both agents will get zero have been pruned from the tree. These paths are not played in equilibrium, and their removal allows us to focus on the relevant parts of the tree.

Since Alice gets to make the first move, she has to decide whether to offer no arguments at all $\{\}$, suggest that they go to the ballet (α_2), or present her counter-argument to the soccer match before the soccer match is even brought up (α_3). Based on the argument uttered by Alice, Brian then gets to make a decision. If Alice had made no argument, then as long as Brian announces α_1 (and possibly also α_4) then they will go to the soccer match. Brian will receive utility of two and Alice will receive utility of one (the subtree on the left). If Alice suggested going to the ballet (α_2) then Brian is best off immediately raising his counter-argument to the ballet (α_4). This is because Alice's best counter-argument is then to say nothing, which allows Brian to then present soccer (α_1) as an alternative. This results in Brian getting his favorite outcome (soccer) since Alice's only other option would be to raise argument α_3, resulting in them both staying at home (the least preferred outcome). If instead, Brian had not made an argument or had uttered α_1, then Alice would have been able to to raise her counter-argument, resulting in them both going to the ballet. Finally, if Alice first announces her counter-argument to soccer (α_3) then Brian will end up announcing, at most, argument α_1 since raising the counter-argument to the ballet (α_4) will result in them both staying at home. This means that the outcome will be that both Alice and Brian will go to the ballet. Alice uses this reasoning in order to determine what her initial action should be. She will realize that if she makes no argument, or initially suggests the ballet (α_2) then Brian will be able to take actions so that they end up going to the soccer match. However, if Alice starts with her counter-argument to the soccer match (α_3) then she can force Brian into a situation where he is best off not making his counter-argument to the ballet, and so they will both end up going to the ballet, Alice's preferred outcome. Therefore, in equilibrium, Alice will state her objection to soccer (α_3) first, which will force Brian to either make no argument or make (already defeated) argument α_1, which then allows Alice to counter with the ballet proposal (α_2). This equilibrium is called a *subgame perfect equilibrium* and is a refinement of the Nash equilibria.

We note that by going first, Alice had an advantage over Brian since by carefully choosing her first argument she could force the outcome that she wanted. If Brian had gone first, then he would have been best off first announcing α_4, his counter-argument to the ballet. This would have allowed him to get the outcome that he preferred, that is, the soccer match. Thus in the extensive-form game analysis of argumentation, the *order* in which agents make arguments is critical in the analysis and in the outcome.

A number of researchers have proposed using extensive-form games of perfect information to model argumentation. For example, Procaccia and Rosenschein [10] proposed a *game-based argumentation framework* where they extend Dung's abstract argumentation framework by mapping argumentation frameworks into extensive-form games of perfect information. A similar approach has recently been proposed by Riveret et al [12], giving an extensive-form game characterisation of Prakken's dialectical framework [8]. In both cases, the authors show how standard backward induction techniques can be used to eliminate dominated strategies and characterise Nash equilibrium strategies.

3 Technical Background

Before we present a precise formal mapping of abstract argumentation into game theory, in this section, we give a brief background on key game-theoretic concepts. Readers who lack background in game theory may consult a more comprehensive introduction to the field, such as [5].

3.1 Game Theory

The field of game theory studies strategic interactions of self-interested agents. We assume that there is a set of self-interested agents, denoted by I. We let $\theta_i \in \Theta_i$ denote the *type* of agent i which is drawn from some set of possible types Θ_i. The type represents the private information and preferences of the agent. An agent's preferences are over *outcomes* $o \in \mathcal{O}$, where $\mathcal{O}$ is the set of all possible outcomes. We assume that an agent's preferences can be expressed by a utility function $u_i(o, \theta_i)$ which depends on both the outcome, o, and the agent's type, θ_i. Agent i prefers outcome o_1 to o_2 when $u_i(o_1, \theta_i) > u_i(o_2, \theta_i)$.

When agents interact, we say that they are playing *strategies*. A strategy for agent i, $s_i(\theta_i)$, is a plan that describes what actions the agent will take for every decision that the agent might be called upon to make, for each possible piece of information that the agent may have at each time it is called to act. That is, a strategy can be thought as a complete contingency plan for an agent. We let Σ_i denote the set of all possible strategies for agent i, and thus $s_i(\theta_i) \in \Sigma_i$. When it is clear from the context, we will drop the θ_i in order to simplify the notation. We let *strategy profile* $s = (s_1(\theta_1), \ldots, s_I(\theta_I))$ denote the outcome that results when each agent i is playing strategy $s_i(\theta_i)$. As a notational convenience we define

$$s_{-i}(\theta_{-i}) = (s_1(\theta_i), \ldots, s_{i-1}(\theta_{i-1}), s_{i+1}(\theta_{i+1}), \ldots, s_I(\theta_I))$$

and thus $s = (s_i, s_{-i})$. We then interpret $u_i((s_i, s_{-i}), \theta_i)$ to be the utility of agent i with type θ_i when all agents play strategies specified by strategy profile $(s_i(\theta_i), s_{-i}(\theta_{-i}))$.

Similarly, we also define:

$$\theta_{-i} = (\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_I)$$

Since the agents are all self-interested, they will try to choose strategies which maximize their own utility. Since the strategies of other agents also play a role in determining the outcome, the agents must take this into account. The *solution concepts* in game theory determine the outcomes that will arise if all agents are rational and strategic. The most well known solution concept is the *Nash equilibrium*. A Nash equilibrium is a strategy profile in which each agent is following a strategy which maximizes its own utility, given its type and the strategies of the other agents.

Definition 16.1 (Nash Equilibrium). A strategy profile $s^* = (s_1^*, \ldots, s_I^*)$ is a *Nash equilibrium* if no agent has incentive to change its strategy, given that no other agent changes. Formally,

$$\forall i, \forall s_i', u_i(s_i^*, s_{-i}^*, \theta_i) \geq u_i(s_i', s_{-i}^*, \theta_i).$$

Although the Nash equilibrium is a fundamental concept in game theory, it does have several weaknesses. First, there may be multiple Nash equilibria and so agents may be uncertain as to which equilibrium they should play. Second, the Nash equilibrium implicitly assumes that agents have perfect information about all other agents, including the other agents' preferences.

A stronger solution concept in game theory is the *dominant-strategy equilibrium*. A strategy s_i is said to be *dominant* if by playing it, the utility of agent i is maximized no matter what strategies the other agents play.

Definition 16.2 (Dominant Strategy). A strategy s_i^* is *dominant* if

$$\forall s_{-i}, \ \forall s_i', \ u_i(s_i^*, s_{-i}, \theta_i) \geq u_i(s_i', s_{-i}, \theta_i).$$

Sometimes, we will refer to a strategy satisfying the above definition as *weakly dominant*. If the inequality is strict (i.e. $>$ instead of $\geq$), we say that the strategy is *strictly dominant*.

A dominant-strategy equilibrium is a strategy profile where each agent is playing a dominant strategy. This is a very robust solution concept since it makes no assumptions about what information the agents have available to them, nor does it assume that all agents know that all other agents are being rational (i.e. trying to maximize their own utility). However, there are many strategic settings where no agent has a dominant strategy.

A third solution concept is the *Bayes-Nash equilibrium*. We include it for the sake of completeness. In the Bayes-Nash equilibrium the assumption made for the Nash equilibrium, that all agents know the preferences of others, is relaxed. Instead, we assume that there is some common prior $F((\Theta_1, \ldots, \Theta_I))$, such that the agents' types are distributed according to F. Then, in equilibrium, each agent chooses the strategy that maximizes it's expected utility given the strategies other agents are playing and the prior F.

Definition 16.3 (Bayes-Nash Equilibrium). A strategy profile $s^* = (s_i^*, s_{-i}^*)$ is a Bayes-Nash equilibrium if

$$E_{\theta_{-i}}[u_i((s_i^*(\theta_i), s_{-i}^*(\cdot)), \theta_i)] \geq E_{\theta_{-i}}[u_i((s_i'(\theta_i), s_{-i}^*(\cdot)), \theta_i)] \; \forall \theta_i, \forall s_i'.$$

3.2 Mechanism Design

The problem that mechanism design studies is how to ensure that a desirable system-wide outcome or decision is made when there is a group of self-interested agents who have preferences over the outcomes. In particular, we often want the outcome to depend on the preferences of the agents. This is captured by a *social choice function*.

Definition 16.4 (Social Choice Function). A *social choice function* is a rule $f: \Theta_1 \times \ldots \times \Theta_I \rightarrow \mathcal{O}$, that selects some outcome $f(\theta) \in \mathcal{O}$, given agent types $\theta = (\theta_1, \ldots, \theta_I)$.

The challenge, however, is that the types of the agents (the $\theta_i's$) are private and known only to the agents themselves. Thus, in order to select an outcome with the social choice function, one has to rely on the agents to reveal their types. However, for a given social choice function, an agent may find that it is better off if it does not reveal its type truthfully, since by lying it may be able to cause the social choice function to choose an outcome that it prefers. Instead of trusting the agents to be truthful, we use a *mechanism* to try to reach the correct outcome.

A mechanism $\mathcal{M} = (\Sigma, g(\cdot))$ defines the set of allowable strategies that agents can chose, with $\Sigma = \Sigma_1 \times \cdots \times \Sigma_I$ where Σ_i is the strategy set for agent i, and an outcome function $g(s)$ which specifies an outcome o for each possible strategy profile $s = (s_1, \ldots, s_I) \in \Sigma$. This defines a game in which agent i is free to select any strategy in Σ_i, and, in particular, will try to select a strategy which will lead to an outcome that maximizes its own utility. We say that a mechanism *implements* social choice function f if the outcome induced by the mechanism is the same outcome that the social choice function would have returned if the true types of the agents were known.

Definition 16.5 (Implementation). A mechanism $\mathcal{M} = (\Sigma, g(\cdot))$ *implements* social choice function f if there exists an equilibrium s^* such that

$$\forall \theta \in \Theta, \; g(s^*(\theta)) = f(\theta).$$

While the definition of a mechanism puts no restrictions on the strategy spaces of the agents, an important class of mechanisms are the *direct-revelation mechanisms* (or simply *direct mechanisms*).

Definition 16.6 (Direct-Revelation Mechanism). A *direct-revelation mechanism* is a mechanism in which $\Sigma_i = \Theta_i$ for all i, and $g(\theta) = f(\theta)$ for all $\theta \in \Theta$.

In words, a direct mechanism is one where the strategies of the agents are to announce a type, θ_i' to the mechanism. While it is not necessary that $\theta_i' = \theta_i$, the important *Revelation Principle* (see below for more details) states that if a social choice function, $f(\cdot)$, can be implemented, then it can be implemented by a direct mechanism where every agent reveals its true type [5]. In such a situation, we say that the social choice function is *incentive compatible*.

Definition 16.7 (Incentive Compatible). The social choice function $f(\cdot)$ is *incentive compatible* (or *truthfully implementable*) if the direct mechanism $\mathcal{M} = (\Theta, g(\cdot))$ has an equilibrium s^* such that $s_i^*(\theta_i) = \theta_i$.

If the equilibrium concept is the dominant-strategy equilibrium, then the social choice function is *strategy-proof*. In this chapter we will on occasion call a mechanism incentive-compatible or strategy-proof. This means that the social choice function that the mechanism implements is incentive-compatible or strategy-proof.

3.3 The Revelation Principle

Determining whether a particular social choice function can be implemented, and in particular, finding a mechanism which implements a social choice function appears to be a daunting task. In the definition of a mechanism, the strategy spaces of the agents are unrestricted, leading to an infinitely large space of possible mechanisms. However, the *Revelation Principle* states that we can limit our search to a special class of mechanisms [5, Ch 14].

Theorem 16.1 (Revelation Principle). *If there exists some mechanism that implements social choice function f in dominant strategies, then there exists a direct mechanism that implements f in dominant strategies and is truthful.*

The intuitive idea behind the Revelation Principle is fairly straightforward. Suppose that you have a, possibly very complex, mechanism, $\mathcal{M}$, which implements some social choice function, f. That is, given agent types $\theta = (\theta_1, \ldots, \theta_I)$ there exists an equilibrium $s^*(\theta)$ such that $g(s^*(\theta)) = f(\theta)$. Then, the Revelation Principle states that it is possible to create a new mechanism, $\mathcal{M}'$, which, when given θ, will then execute $s^*(\theta)$ on behalf of the agents and then select outcome $g(s^*(\theta))$. Thus, each agent is best off revealing θ_i, resulting in $\mathcal{M}'$ being a truthful, direct mechanism for implementing social choice function f.

The Revelation Principle is a powerful tool when it comes to studying implementation. Instead of searching through the entire space of mechanisms to check whether one implements a particular social choice function, the Revelation Principle states that we can restrict our search to the class of truthful, direct mechanisms. If we can not find a mechanism in this space which implements the social choice function of interest, then there does not exist any mechanism which will do so.

It should be noted that while the Revelation Principle is a powerful analysis tool, it does not imply we should only design direct mechanisms. Some reasons why one rarely sees direct mechanisms in the “real world” include (among others);

- they can place a high computational burden on the mechanism since it is required to execute agents' strategies,
- agents' strategies may be computationally difficult to determine, and
- agents may not be willing to reveal their true types because of privacy concerns.

4 Argumentation Mechanism Design

Mechanism design (MD) is a sub-field of game theory concerned with the following question: *what game rules guarantee a desirable social outcome when each self-interested agent selects the best strategy for itself?* In other words, while game theory is concerned with a given strategic situation modelled as a game, mechanism design is concerned with designing the game itself. As such, one might actually call it *reverse game theory*.

In this section we define the mechanism design problem for abstract argumentation. We dub this new approach 'Argumentation Mechanism Design' (ArgMD).

Let $AF = \langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework with a set of arguments $\mathcal{A}$ and a binary defeat relation $\mathcal{R}$. We define a mechanism with respect to AF and semantics $\mathcal{S}$, and we assume that there is a set of I self-interested agents. We define an agent's type to be its set of arguments.

Definition 16.8 (Agent Type). Given an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$, the *type* of agent i, $\mathcal{A}_i \subseteq \mathcal{A}$, is the set of arguments that the agent is capable of putting forward.

There are two things to note about this definition. Firstly, an agent's type can be seen as a reflection of its expertise or domain knowledge. For example, medical experts may only be able to comment on certain aspects of forensics in a legal case, while a defendant's family and friends may be able to comment on his/her character. Also, such expertise may overlap, so agent types are not necessarily disjoint. For example, two medical doctors might have some identical argument, and so on.

The second thing to note about the definition is that agent types do not include the defeat relation. In other words, we implicitly assume that the notion of defeat is common to all agents. That is, given two arguments, no agent would dispute whether one attacks another. This is a reasonable assumption in systems where agents use the same logic to express arguments or at least multiple logics for which the notion of defeat is accepted by everyone (e.g. conflict between a proposition and its negation). Disagreement over the defeat relation itself requires a form of hierarchical (meta) argumentation [7], which is a powerful concept, but is beyond the scope of the present chapter.

Given the agents' types (argument sets) a social choice function f maps a type profile into a subset of arguments;

$$f : 2^{\mathcal{A}} \times \ldots \times 2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$$

While our definition of an argumentation mechanism will allow for generic social choice functions which map type profiles into subsets of arguments, we will be particularly interested in *argument acceptability* social choice functions. We denote by $Acc(\langle \mathcal{A}, \mathcal{R} \rangle, \mathcal{S}) \subseteq \mathcal{A}$ the set of acceptable arguments according to semantics $\mathcal{S}$.[3]

Definition 16.9 (Argument Acceptability Social Choice Functions). Given an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ with semantics $\mathcal{S}$, and given an agent type profile $(\mathcal{A}_1, \ldots, \mathcal{A}_I)$, the *argument acceptability social choice function* f is defined as the set of acceptable arguments given the semantics $\mathcal{S}$. That is,

$$f(\mathcal{A}_1, \ldots, \mathcal{A}_I) = Acc(\langle \mathcal{A}_1 \cup \ldots \cup \mathcal{A}_I, \mathcal{R} \rangle, \mathcal{S}).$$

As is standard in the mechanism design literature, we assume that agents have preferences over the outcomes $o \in 2^{\mathcal{A}}$, and we represent these preferences using utility functions where $u_i(o, \mathcal{A}_i)$ denotes agent i's utility for outcome o when its type is argument set $\mathcal{A}_i$.

Agents may not have incentive to reveal their true type because they may be able to influence the final argument status assignment by lying, and thus obtain higher utility. There are two ways that an agent can lie in our model. On one hand, an agent might create new arguments that it does not have in its argument set. In the rest of the chapter we will assume that there is an *external verifier* that is capable of checking whether it is possible for a particular agent to actually make a particular argument. Informally, this means that presented arguments, while still possibly defeasible, must at least be based on some sort of demonstrable 'plausible evidence.' If an agent is caught making up arguments then it will be removed from the mechanism. For example, in a court of law, any act of perjury by a witness is punished, at the very least, by completely discrediting all evidence produced by the witness. Moreover, in a court of law, arguments presented without any plausible evidence are normally discarded (e.g. "*I did not kill him, since I was abducted by aliens at the time of the crime!*"). For all intents and purposes this assumption (also made by Glazer and Rubinstein [2]) removes the incentive for an agent to make up facts.

A more insidious form of manipulation occurs when an agent decides to *hide* some of its arguments. By refusing to reveal certain arguments, an agent might be able to break defeat chains in the argument framework, thus changing the final set of acceptable arguments. For example, a witness may hide evidence that implicates the defendant if the evidence also undermines the witness's own character. This type of lie is almost impossible to detect in practice, and it is this form of strategic behaviour that we will be the most interested in.

As mentioned earlier, a strategy of an agent specifies a complete plan that describes what action the agent takes for every decision that a player might be called upon to take, for every piece of information that the player might have at each time that it is called upon to act. In our model, the actions available to an agent involve announcing sets of arguments. Thus a strategy $s_i \in \Sigma_i$ for agent i would specify for

[3] Here, we assume that $\mathcal{S}$ specifies both the classical semantics used (e.g. grounded, preferred, stable) as well as the acceptance attitude used (e.g. sceptical or credulous).

each possible subset of arguments that could define its type, what set of arguments to reveal. For example, a strategy might specify that an agent should reveal only half of its arguments without waiting to see what other agents are going to do, while another strategy might specify that an agent should wait and see what arguments are revealed by others, before deciding how to respond. In particular, beyond specifying that agents are not allowed to make up arguments, we place no restrictions on the allowable strategy spaces, when we initially define an argumentation mechanism. Later, when we talk about *direct* argumentation mechanisms we will further restrict the strategy space.

We are now ready to define our argumentation mechanism. We first define a generic mechanism, and then specify a direct argumentation mechanism, which due to the Revelation Principle, is the type of mechanism we will study in the rest of the chapter.

Definition 16.10 (Argumentation Mechanism). Given an argumentation framework $AF = \langle \mathcal{A}, \mathcal{R} \rangle$ and semantics $\mathcal{S}$, an *argumentation mechanism* is defined as

$$\mathcal{M}_{AF}^{\mathcal{S}} = (\Sigma_1, \ldots, \Sigma_I, g(\cdot))$$

where Σ_i is an argumentation strategy space of agent i and $g : \Sigma_1 \times \ldots \times \Sigma_I \to 2^{\mathcal{A}}$.

Note that in the above definition, the notion of dialogue strategy is broadly construed and would depend on the protocol used. In a *direct* mechanism, however, the strategy spaces of the agents are restricted so that they can only reveal a subset of arguments.

Definition 16.11 (Direct Argumentation Mechanism). Given an argumentation framework $AF = \langle \mathcal{A}, \mathcal{R} \rangle$ and semantics $\mathcal{S}$, a *direct argumentation mechanism* is defined as

$$\mathcal{M}_{AF}^{\mathcal{S}} = (\Sigma_1, \ldots, \Sigma_I, g(\cdot))$$

where $\Sigma_i = 2^{\mathcal{A}}$ and $g : \Sigma_1 \times \ldots \Sigma_I \to 2^{\mathcal{A}}$.

In Table 16.1, we summarise the mapping of multi-agent abstract argumentation as an instance of a mechanism design problem.

MD Concept	**ArgMD Instantiation**
Agent type $\theta_i \in \Theta_i$	Agent's arguments $\theta_i = \mathcal{A}_i \subseteq \mathcal{A}$
Outcome $o \in \mathcal{O}$	Accepted arguments $Acc(.) \subseteq \mathcal{A}$
Utility $u_i(o, \theta_i)$	Preferences over $2^{\mathcal{A}}$ (what arguments end up being accepted)
Social choice function $f : \Theta_1 \times \ldots \times \Theta_I \to \mathcal{O}$	$f(\mathcal{A}_1, \ldots, \mathcal{A}_I) = Acc(\langle A_1 \cup \ldots \cup \mathcal{A}_I, \mathcal{R} \rangle, \mathcal{S})$. by some argument acceptability criterion
Mechanism $\mathcal{M} = (\Sigma, g(\cdot))$ where $\Sigma = \Sigma_1 \times \cdots \times \Sigma_I$ and $g : \Sigma \to \mathcal{O}$	Σ_i is an argumentation strategy, $g : \Sigma \to 2^{\mathcal{A}}$
Direct mechanism: $\Sigma_i = \Theta_i$	$\Sigma_i = 2^{\mathcal{A}}$ (every agent reveals a set of arguments)
Truth revelation	Revealing $\mathcal{A}_i$

Table 16.1 Abstract argumentation as a mechanism

5 Case Study: Implementing the Grounded Semantics

In this section, we demonstrate the power of our ArgMD approach by showing how it can be used to systematically analyse the strategic incentives imposed by a well-established argument evaluation criterion. In particular, we specify a direct-revelation argumentation mechanism, in which agents' strategies are to reveal sets of arguments, and where the mechanism calculates the outcome using sceptical (grounded) semantics.[4] That is, we look at the grounded semantics as if it was designed as a mechanism and analyse it from that perspective. We show that, in general, this mechanism gives rise to strategic manipulation. We prove, however, that under various conditions, this mechanism turns out to be strategy-proof.

In a direct argumentation mechanism, each agent i's available actions are $\Sigma_i = 2^{\mathcal{A}}$. We will refer to a specific action (i.e. set of declared arguments) as $\mathcal{A}_i^{\circ} \in \Sigma_i$.

We now present a direct mechanism for argumentation based on a sceptical argument evaluation criteria. The mechanism calculates the grounded extension given the union of all arguments revealed by agents.

Definition 16.12 (Grounded Direct Argumentation Mechanism). A *grounded direct argumentation mechanism* for argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ is $\mathcal{M}_{AF}^{grnd} = (\Sigma_1, \ldots, \Sigma_I, g(.))$ where:

- $\Sigma_i \in 2^{\mathcal{A}}$ is the set of strategies available to each agent;
- $g : \Sigma_1 \times \cdots \times \Sigma_I \rightarrow 2^{\mathcal{A}}$ is an outcome rule defined as: $g(\mathcal{A}_1^{\circ}, \ldots, \mathcal{A}_I^{\circ}) = Acc(\langle \mathcal{A}_1^{\circ} \cup \cdots \cup \mathcal{A}_I^{\circ}, \mathcal{R} \rangle, \mathcal{S}^{grnd})$ where $\mathcal{S}^{grnd}$ denotes sceptical grounded acceptability semantics.

To simplify our analysis, we will assume below that agents can only lie by hiding arguments, and not by making up arguments. Formally, this means that $\forall i$, $\Sigma_i \in 2^{\mathcal{A}_i}$.

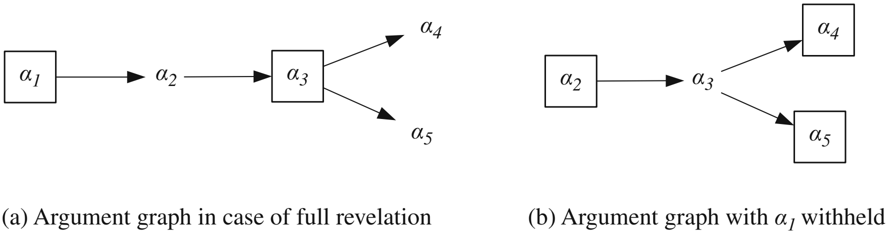
For the sake of illustration, we will consider a particular family of preferences that agents may have. According to these preferences, every agent attempts to maximise the number of arguments in $\mathcal{A}_i$ that end up being accepted. We call this preference criteria the *individual acceptability maximising preference*.

Definition 16.13 (Acceptability maximising preferences). An agent i has *individual acceptability maximising preferences* if and only if $\forall o_1, o_2 \in \mathcal{O}$ such that $|o_1 \cap \mathcal{A}_i| \geq |o_2 \cap \mathcal{A}_i|$, we have $u_i(o_1, \mathcal{A}_i) \geq u_i(o_2, \mathcal{A}_i)$.

Let us now consider aspects of incentives using mechanism $\mathcal{M}_{AF}^{grnd}$ through an example.

Example 16.1. Consider grounded direct argumentation mechanism with three agents x, y and z with types $\mathcal{A}_x = \{\alpha_1, \alpha_4, \alpha_5\}$, $\mathcal{A}_y = \{\alpha_2\}$ and $\mathcal{A}_z = \{\alpha_3\}$ respectively. And suppose that the defeat relation is defined as follows: $\mathcal{R} = \{(\alpha_1, \alpha_2), (\alpha_2, \alpha_3), (\alpha_3, \alpha_4), (\alpha_3, \alpha_5)\}$. If each agent reveals its true type (i.e. $\mathcal{A}_x^{\circ} = \mathcal{A}_x$; $\mathcal{A}_y^{\circ} = \mathcal{A}_y$; and

[4] In the remainder of the chapter, we will use the term *sceptical* to refer to *sceptical grounded*, since the chapter focuses on the grounded semantics.



(a) Argument graph in case of full revelation

(b) Argument graph with α_1 withheld

Fig. 16.3 Hiding an argument is beneficial (case of acceptability maximisers)

$\mathcal{A}_z^\circ = \mathcal{A}_z$), then we get the argument graph depicted in Figure 16.3(a). The mechanism outcome rule produces the outcome $o = \{\alpha_1, \alpha_3\}$. If agents have individual acceptability maximising preferences, with utilities equal to the number of arguments accepted, then: $u_x(o, \{\alpha_1, \alpha_4, \alpha_5\}) = 1$; $u_y(o, \{\alpha_3\}) = 1$; and $u_z(o, \{\alpha_2\}) = 0$.

It turns out that the mechanism is susceptible to strategic manipulation, even if we suppose that agents do not lie by making up arguments (i.e., they may only withhold some arguments). In this case, for both agents y and z, revealing their true types weakly dominates revealing nothing at all. However, it turns out that agent x is better off revealing $\{\alpha_4, \alpha_5\}$. By withholding α_1, the resulting argument network becomes as depicted in Figure 16.3(b), for which the output rule produces the outcome $o' = \{\alpha_2, \alpha_4, \alpha_5\}$. This outcome yields utility 2 to agent x, which is better than the truth-revealing strategy.

Remark 16.1. Given an arbitrary argumentation framework AF and agents with acceptability maximising preferences, mechanism $\mathcal{M}_{AF}^{grnd}$ is *not* strategy-proof.

The following theorem provides a full characterisation of strategy-proof mechanisms for sceptical argumentation frameworks for agents with acceptability maximising preferences.

Theorem 16.2. *Let AF be an arbitrary argumentation framework, and let $\mathcal{E}_{\mathcal{GR}}(AF)$ denote its grounded extension. Mechanism $\mathcal{M}_{AF}^{grnd}$ is strategy-proof for agents with acceptability maximising preferences if and only if AF satisfies the following condition: $\forall i \in I, \forall S \subseteq \mathcal{A}_i$ and $\forall \mathcal{A}_{-i}$, we have $|\mathcal{A}_i \cap \mathcal{E}_{\mathcal{GR}}(\langle \mathcal{A}_i \cup \mathcal{A}_{-i}, \mathcal{R}\rangle)| \geq |\mathcal{A}_i \cap \mathcal{E}_{\mathcal{GR}}(\langle (\mathcal{A}_i \backslash S) \cup \mathcal{A}_{-i}, \mathcal{R}\rangle)|$.*

Although the above theorem gives us a full characterisation, it is difficult to apply in practice. In particular, the theorem does not give us an indication of how agents (or the mechanism designer) can identify whether the mechanism is strategy-proof for a class of argumentation frameworks by appealing to their graph-theoretic properties. Below, we provide an intuitive, graph-theoretic condition that is sufficient to ensure that $\mathcal{M}_{AF}^{grnd}$ is strategy-proof when agents have focal arguments.

Let $\alpha, \beta \in \mathcal{A}$. We say that α *indirectly defeats* β, written $\alpha \hookrightarrow \beta$, if and only if there is an odd-length path from α to β in the argument graph.

Theorem 16.3. *Suppose agents have individual acceptability maximising preferences. If each agent's type corresponds to a conflict-free set of arguments which does not include (in)direct defeats (formally $\forall i \nexists \alpha_1, \alpha_2 \in \mathcal{A}_i$ such that $\alpha_1 \hookrightarrow \alpha_2$), then $\mathcal{M}_{AF}^{grnd}$ is strategy-proof.*

Note that in the theorem, $\hookrightarrow$ is over all arguments in $\mathcal{A}$. Intuitively, the condition in the theorem states that *all* arguments of every agent must be conflict-free (i.e. consistent), both explicitly and implicitly. Explicit consistency implies that no argument defeats another. Implicit consistency implies that other agents cannot possibly present a set of arguments that reveal an indirect defeat among one's own arguments. More concretely, in Example 16.1 and Figure 16.3, while agent x's argument set $\mathcal{A}_x = \{\alpha_1, \alpha_4, \alpha_5\}$ is conflict-free, when agents y and z presented their own arguments α_2 and α_3, they revealed an implicit conflict in x's arguments. In other words, they showed that x contradicts himself (i.e. committed a *fallacy* of some kind).

In addition to characterising a sufficient graph-theoretic condition for strategy-proofness, Theorem 16.3 is useful for individual agents. As long as the agent knows that it is *not* possible for a path to be created which causes an (in)direct defeat among its arguments (i.e., a fallacy to be revealed), then the agent is best off revealing all its arguments. The agent only needs to know that no argument imaginable can reveal conflicts among its own arguments.

We now ask whether the *sufficient* condition in Theorem 16.3 is also *necessary* for agents to reveal all their arguments truthfully. Example 16.2 shows that this is not the case. In particular, for certain argumentation frameworks, an agent may have truthtelling as a dominant strategy despite the presence of indirect defeats among its own arguments.

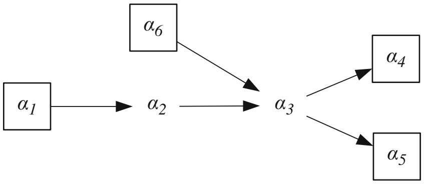


Fig. 16.4 Strategy-proofness despite indirect self-defeat

Example 16.2. Consider the variant of Example 16.1 with the additional argument α_6 and defeat (α_6, α_3). Let the agent types be $\mathcal{A}_x = \{\alpha_1, \alpha_4, \alpha_5, \alpha_6\}$, $\mathcal{A}_y = \{\alpha_2\}$ and $\mathcal{A}_z = \{\alpha_3\}$ respectively. The full argument graph is depicted in Figure 16.4. Under full revelation, the mechanism outcome rule produces the outcome $o = \{\alpha_1, \alpha_4, \alpha_5, \alpha_6\}$.

Note that in Example 16.2, truth revelation is now a dominant strategy for x (since it gets all its arguments accepted) despite the fact that $\alpha_1 \hookrightarrow \alpha_4$ and $\alpha_1 \hookrightarrow \alpha_5$. This hinges on the presence of an argument (namely α_5) that cancels out the negative effect of the (in)direct self-defeat among x's own arguments.

6 Related Work

6.1 Pareto Optimality of Outcomes

A well-known property of the grounded semantics is that it is extremely sceptical, accepting only undefeated arguments and arguments defended by undefeated arguments. An interesting question, then, is whether it is possible to be more inclusive (i.e. being more credulous) in order to produce argumentation outcomes that are more *socially* desirable. For example, consider the simple argument graph in Figure 16.5 and suppose we have two agents with types $\mathcal{A}_1 = \{\alpha_1\}$ and $\mathcal{A}_2 = \{\alpha_2\}$ who both reveal their arguments. The grounded extension (Figure 16.5(a)) is empty here.

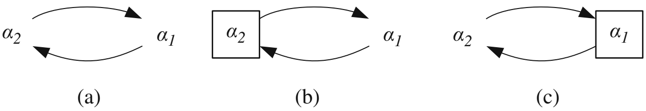


Fig. 16.5 Preferred extensions ‘dominate’ the grounded extension

Suppose the judge chooses one of the preferred extensions instead (Figure 16.5(b) or (c)). Clearly, when compared to outcome (a), each preferred extension makes one agent better-off without making the other worse-off. Formally, we say that outcomes (b) and (c) each Pareto dominates outcome (a). An outcome that is not Pareto dominated is called *Pareto optimal*.

Recently, Rahwan and Larson [11] presented an extensive analysis of Pareto optimality in abstract argumentation, and established correspondence results between different semantics on one hand and the Pareto optimal outcomes on the other.

6.2 Glazer and Rubinstein’s Model

Another game-theoretic analysis of argumentation was presented by Glazer and Rubinstein [2]. The authors explore the mechanism design problem of constructing rules of debate that maximise the probability that a listener reaches the right conclusion given arguments presented by two debaters. They study a very restricted setting, in which the world state is described by a vector $\omega = (w_1, \ldots, w_5)$, where each ‘aspect’ w_i has two possible values: 1 and 2. If $w_i = j$ for $j \in \{1,2\}$, we say that aspect w_i supports outcome O_j. Presenting an argument amounts to revealing the value of some w_i. The setting is modelled as an extensive-form game and analysed. In particular, the authors investigate various combinations of *procedural rules* (stating in which order and what sorts of arguments each debater is allowed to state) and *persuasion rules* (stating how the outcome is chosen by the listener). In terms of procedural rules, the authors explore: (1) *one-speaker debate* in whichone debater chooses two arguments to reveal; (2) *simultaneous debate* in which the two debaters simultaneously reveal one argument each; and (3) *sequential debate* in which one debater reveals one argument followed by one argument by the other. Our mechanism is closer to the simultaneous debate, but is much more general as it enables the simultaneous revelation of an arbitrary number of arguments. Glazer and Rubinstein investigate a variety of persuasion rules. For example, in one-speaker debate, one rule analysed by the authors states that ‘a speaker wins if and only if he presents two arguments from $\{a_1, a_2, a_3\}$ or $\{a_4, a_5\}$.’ In a sequential debate, one persuasion rule states that ‘if debater D_1 argues for aspect a_3, then debater D_2 wins if and only if he counter-argues with aspect a_4.’ These kinds of rules are arbitrary and do not follow an intuitive notion of persuasion (e.g. like scepticism). The sceptical mechanism presented in this chapter provides a more natural criterion for argument evaluation, supplemented by a strong solution concept that ensures all agents have incentive to reveal their arguments, and thus for the listener to reach the correct outcome. Moreover, our framework for argumentation mechanism design is more general in that it can be used to model a variety of more complex argumentation settings.

6.3 Game Semantics

It is worth contrasting our work with work on so-called *game semantics* for logic, which was pioneered by logicians such as Paul Lorenzen [4] and Jaakko Hintikka [3]. Although many specific instantiations of this notion have been presented in the literature, the general idea is as follows. Given some specific logic, the truth value of a formula is determined through a special-purpose, multi-stage dialogue game between two players, the *verifier* and *falsifier*. The formula is considered true precisely when the verifier has a winning strategy, while it will be false whenever the falsifier has the winning strategy. Similar ideas have been used to implement dialectical proof-theories for defeasible reasoning (e.g. by Prakken and Sartor [9]).

In a related development, Matt and Toni recently proposed a game-theoretic approach to characterise argument strength [6]. Thus, the acceptability of each argument is rated between 0 and 1 by using a two-person zero-sum game with imperfect information between a proponent and an opponent.

There is a fundamental difference between the aims of game semantics and our ArgMD approach. In game semantics, the goal is to interpret (i.e., characterise the truth value of) a specific formula by appealing to a notion of a winning strategy. As such, each player is carefully endowed with a specific set of formulae to enable the game to characterise semantics correctly (e.g. the verifier may own all the disjunctions in the formula, while the falsifier is given all the conjunctions).

In contrast, ArgMD is about designing rules for argumentation among self-interested players who may have incentives to manipulate the outcome given a variety of possible individual preferences (specified in arbitrary instantiations of a utility function). Our interest is in conditions that guarantee truth-revelation given different classes of preferences. Game semantics have no similar notion of strategic manipulation by hiding information. Moreover, our framework allows an arbitrary number of players (as opposed to two agents).

6.4 Argumentation and Cooperative Games

Another notable early link between argumentation and *cooperative game theory* has been proposed by Dung in his seminal paper [1]. Let $A = \{a_1, \ldots, a_{|A|}\}$ be a set of agents. A cooperative game is defined by specifying a value to $V(C)$ to each coalition $C \subseteq A$ of agents. An outcome of the game is a vector $\mathbf{u} = (u_{a_1}, \ldots, u_{a_{|A|}}) \in \mathbb{R}^{|A|}$ specifying a vector of utilities, one per agent.

Outcome $\mathbf{u}$ dominates outcome $\mathbf{u}'$ if there is a (nonempty) coalition $K \subseteq A$ in which agents get more utility (as a whole) in $\mathbf{u}$ than in $\mathbf{u}'$. An outcome is said to be *stable* if no outcome dominates it (i.e. if no subset of agents has *incentive* to leave their own coalition and be all individually better off). A *solution* of the cooperative game is a set of outcomes S satisfying the following conditions:

1. No $s \in S$ is dominated by an $s' \in S$.
2. Every $s \notin S$ is dominated by some $s' \in S$.

Dung argued that an n-person game can be seen as an argumentation framework $\langle \mathcal{A}, \mathcal{R} \rangle$ in which the set of arguments $\mathcal{A}$ is the set of all possible outcomes of the cooperative game, and the defeat relation is defined as $\mathcal{R} = \{(\mathbf{u}, \mathbf{u}') \mid \mathbf{u} \text{ dominates } \mathbf{u}'\}$. Dung shows that with this characterisation, the set of solutions to the cooperative game corresponds to the set of stable extensions of an abstract argumentation framework. This enabled Dung to describe the well-known *stable marriage problem* as a problem of finding a stable extension.

Another important notion in cooperative game theory is that of the *core*: the set of (feasible) outcomes which are not dominated by any other outcome. Dung showed that the core corresponds to $\mathcal{F}(\varnothing)$ where $\mathcal{F}$ is the characteristic function of the corresponding argumentation framework.

7 Conclusion

In this chapter, our aim was to demonstrate the importance of game theory as a tool for analysing strategic argumentation. We showed how normal form games and extensive-form games can be used to analyse equilibrium strategies in strategic argumentation. We then introduced Argumentation Mechanism Design (ArgMD) as a new framework for designing and analysing argument evaluation criteria. With ArgMD, designing new argument acceptance criteria becomes akin to designing auction protocols in strategic multi-agent settings. The goal is to design rules that ensure, under precise conditions, that agents have no incentive to manipulate the outcome. We believe this approach will become increasingly important before argumentation can be applied in open agent systems.

References

1. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
2. J. Glazer and A. Rubinstein. Debates and decisions: On a rationale of argumentation rules. *Games and Economic Behavior*, 36:158–173, 2001.
3. J. Hintikka and G. Sandu. Game-theoretical semantics. In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 361–410. Elsevier, Amsterdam, The Netherlands, 1997.
4. P. Lorenzen. Ein dialogisches konstruktivitätskriterium. In *Infinitistic Methods*, pages 193–200. Pergamon Press, Oxford, UK, 1961.
5. A. Mas-Colell, M. D. Whinston, and J. R. Green. *Microeconomic Theory*. Oxford University Press, New York NY, USA, 1995.
6. P.-A. Matt and F. Toni. A game-theoretic measure of argument strength for abstract argumentation. In S. Hölldobler, C. Lutz, and H. Wansing, editors, *Logics in Artificial Intelligence, 11th European Conference, JELIA 2008*, volume 5293 of *Lecture Notes in Computer Science*, pages 285–297. 2008.
7. S. Modgil. Hierarchical argumentation. In *Proceedings of the 10th European Conference on Logics in Artificial Intelligence. Liverpool, UK*, 2006.
8. H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15(6):1009–1040, 2005.
9. H. Prakken and G. Sartor. Argument-based logic programming with defeasible priorities. *Journal of Applied Non-classical Logics*, 7:25–75, 1997.
10. A. D. Procaccia and J. S. Rosenschein. Extensive-form argumentation games. In *Proceedings of the Third European Workshop on Multi-Agent Systems (EUMAS-05), Brussels, Belgium*, pages 312–322, 2005.
11. I. Rahwan and K. Larson. Pareto optimality in abstract argumentation. In D. Fox and C. Gomes, editors, *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI-2008)*, Menlo Park CA, USA, 2008.
12. R. Riveret, H. Prakken, A. Rotolo, and G. Sartor. Heuristics in argumentation: A game-theoretical investigation. In P. Besnard, S. Doutre, and A. Hunter, editors, *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA)*, pages 324–335. IOS Press, Amsterdam, The Netherlands, 2008.
13. J. von Neuman and O. Morgenstern. *The Theory of Games and Economic Behaviour*. Princeton University Press, Princeton NJ, USA, 1944.