Chapter 1

Introduction

Open distributed computing applications are becoming increasingly commonplace in our society. In most cases, these applications are composed of multiple actors or agents, each with its own aims and objectives. In such complex systems, dependencies between these multiple agents are inevitable, and generally speaking, they cannot all be predicted in advance. Therefore a runtime mechanism is needed to manage them and to resolve any conflicts that might ensue in a context-dependent manner. We believe the de facto mechanism for achieving this is *automated negotiation* and this is the area explored in this thesis.

However, designing effective negotiation mechanisms for open distributed applications is a major research challenge. Specifically, there is a high degree of uncertainty in the variables that impact on negotiations. This is because the actions of the actors (i.e. what they are able to achieve), their preferences (i.e. what outcomes they deem possible and would prefer), their honesty (i.e. to what extent they want to reveal private information truthfully), and their reliability (i.e. how good they are at what they say they can do) are not public knowledge. This uncertainty may, in turn, prevent the agents from reaching good agreements during negotiations (because they are not able to make decisions with full knowledge of the effects of their actions). Given this, the underlying motivation of this thesis is to devise techniques to reduce this uncertainty so that agents can reach better agreements through automated negotiation. In particular, this involves modelling the variables that are prone to uncertainty using decision theoretic techniques (e.g. statistics and/or fuzzy reasoning), determining ways in which the output of such techniques can be used in automated negotiation, and detailing how this output can be refined over multiple encounters between the agents in order to make the search for the best agreement quicker. Against this background, we develop three general classes of techniques that aim to enhance the outcome of such repeated encounters. First, we propose that agents model their opponents' reliability through the notion of trust. To this end, we develop the CREDIT[1] trust model. Using CREDIT, agents are able to adapt

[1] **C**onfidence and **RE**putation **D**efining **I**nteraction-based **T**rust (CREDIT).

their negotiation stance in bargaining encounters according to how trustworthy (reliable and honest) they believe their opponent to be in enacting the contents of a contract. Second, we develop the notion of Trust-Based Mechanism Design (TBMD) that uses game theoretic techniques to select the most reliable agents in the system by incentivizing them to honestly reveal their preferences and their trustworthiness. Third, we develop a novel mechanism for Persuasive Negotiation (PN) for reducing the uncertainty in repeated encounters by allowing agents to constrain the space of outcomes that they need to search in order to find an agreement. Thus, in persuasive negotiation, agents can ask for or give rewards, which constrain future encounters, in an attempt to make an offer in the current negotiation more acceptable.

The rest of this chapter is structured as follows. Section 1.1 maps out the general need for automated negotiation in Multi-Agent Systems (MAS). In section 1.2 we discuss the techniques that are used in negotiation and identify those attributes of negotiation encounters that can be uncertain. In section 1.3, we then discuss the issue of trust as a means to reduce the uncertainty about the honesty or reliability of agents. Then, in section 1.4 we discuss how uncertainties about the action set and preferences of agents can be dealt with in persuasive negotiation. The aims and objectives, as well as the main contributions of the thesis, are outlined in section 1.5 and the structure of the remainder of this thesis is given in section 1.6.

1.1 Motivation for Research

Many computer applications are open distributed systems in which the (very many) constituent components are spread throughout a network, in a decentralised control regime, and are subject to constant change throughout the system's lifetime. Examples include the Grid (Foster and Kesselman, 1998), peer-to-peer computing (Ripeanu et al., 2002), the semantic web (Berners-Lee et al., 2001), web services (Seth, 2003), e-business (Kersten and Lo, 2001), m-commerce (Tveit, 2001; Vulkan, 1999), autonomic computing (Kephart and Chess, 2003), and pervasive computing environments (Satyanarayanan, 2001). Such open distributed systems are typically composed of various stakeholders, each with their own, possibly conflicting, interests. Therefore, there is a need to have autonomous components, that represent these stakeholders, and act and interact in flexible ways in order to achieve their design objectives in uncertain and dynamic environments (Simon, 1996). Given this, agent based computing has been advocated as the natural computation model for such systems (Jennings, 2001).

More specifically, the agent paradigm allows the decomposition of large, complex, and distributed systems into a number of autonomous entities that can interact with each other in order to achieve their individual objectives (Jennings, 2000). To be even more precise, the following definition of an *agent* will be used throughout this work:

Definition 1.1. *An agent is a computer system* situated *in an* environment*,* *and capable of* flexible autonomous action *in this environment in order to meet its* design objectives *(adapted from Wooldridge and Jennings (1995)).*

This definition highlights the fact that an agent must have the following properties:

- Reactivity — the ability to respond to changes to its perceived environment including those changes that result from the actions of other agents.
- Proactiveness — the ability to exploit opportunities to satisfy its goals, rather than constraining itself to predefined rules.
- Social ability — the ability to interact with other agents in its environment to satisfy its goals.

The last of these properties is probably the main defining characteristics of agents that are situated in MAS. In this work, agents within such systems are assured to interact with one another according to some *interaction mechanism* that guides the participants to a particular outcome:

Definition 1.2. *An interaction mechanism is a means by which agents are able to achieve one or more of the following: (i) exchange information, (ii) coordinate their actions and (iii) resolve their conflicts.*

Given this, open distributed systems can be modelled as open multi-agent systems that are composed of autonomous agents that interact with one another using particular interaction mechanisms. Obviously, depending on the nature of the interaction, different types of interaction mechanisms will be used. Broadly speaking, we can characterise the nature of interactions in the following ways:

- *Competitive interactions* — agents interact to satisfy their *own* preferences. These preferences are usually captured through their *utility function* which assigns a score (usually a real value) to particular outcomes in the interaction. In such competitive interactions, agents try to maximise their utility function and are hence termed *selfish* or *self-interested*. Specifically, the agents try to deduce the course of action that maximises their utility given their knowledge of their environment and the possible actions of other agents. This may involve hiding their preferences since doing otherwise might lead to a low utility deal being achieved.[2] Given this, MAS designers have to engineer the system that guides such competitive interactions through *protocols* so that agents do not unduly exploit one another or the overall system in seeking to maximise their individual utility. In so doing, the designer can ensure that the system is fair and incentivises individual stakeholders to participate in it. Generally speaking, these protocols dictate the range of actions that agents can perform (i.e. their action set), the sequence of actions that are permissible (e.g. each agent performing only

[2]Such decision making based on the computation of the utility maximising action relative to other agents' actions is normally termed *strategic* decision making (Rosenschein and Zlotkin, 1994).

one action concurrently with others or a number of actions sequentially with others' actions), and how the agents' actions translate into an outcome (Dash et al., 2003; Rosenschein and Zlotkin, 1994; Sandholm, 1999). Given the system's protocols, the agents' owners need to define the *strategy* of the agents that can achieve their goals (i.e. given the history of actions, what an agent is supposed to do next).

- *Cooperative interactions* — agents interact in order to try and maximise the sum of all their utilities (also termed the *social welfare* (Mas-Colell et al., 1995)) (as opposed to their individual utility in the competitive case). In these interactions, agents totally devote themselves to the group's cause even at the expense of their individual goals (Pynadath and Tambe, 2002) (i.e. even if their individual utility is low in the chosen outcome). In this context, the main problem of the MAS designer is that of devising algorithms (i.e. covering both the protocol used and the strategy of the individual agents) that can find a globally optimum set of actions that still manage to satisfy each agent's constraints (Yokoo and Hirayama, 2000; Becker et al., 2003). The problem of finding the optimum set of actions is usually exacerbated in this case by uncertainties in the knowledge agents have about each other's actions and the number of constraints (or variables) that exist for each agent.

In this thesis we focus on interaction mechanisms that deal with competitive interactions since this represents the most general class of interactions (i.e. a competitive interaction can be reduced to a cooperative one by changing the nature of the utility function of each agent). In particular, as stated earlier, agents, while having selfish interests, may need to collaborate to achieve their goals. In such contexts, agents usually aim to find an *agreement* that determines a course of action that maximises their individual utilities. To this end, a number of techniques have been devised, forming the general class of *negotiation* mechanisms, more commonly known as *automated negotiation* mechanisms in the MAS literature.

1.2 Automated Negotiation Mechanisms

Negotiation has been defined in many different ways (see (Walton and Krabbe, 1995; Fisher and Ury, 1983; Rosenschein and Zlotkin, 1994; Jennings et al., 2000)). However, fundamentally, its main goal is to achieve an *agreement* over some *issue(s)* of contention. In this thesis we adopt the following definition:

Definition 1.3. *Negotiation is an interaction mechanism that aims to resolve a* conflict of interest *between two or more parties through the use of a defined* protocol *and the strategies of the agents (adapted from (Jennings et al., 2001)).*

The protocol usually determines the sequence of steps agents need to follow during negotiation, while the agents' strategies are part of their reasoning mechanism (which also involves information gathering and analysis, and offer generation components). As can be deduced from the above definition, the *aim* of negotiation is to find an agreement that satisfies the agents' preferences or constraints, but such encounters do not always end up in an agreement (and agents may gain zero or negative utility from this). Non-agreement can happen as a result of a lack of time, an unavailability of viable options for the participants (that could result from a lack of knowledge about the participants' preferences), or an incompatibility between the strategies used by the agents (Fisher and Ury, 1983; Raiffa, 1982). However, if an agreement is feasible and the agents are actually able to achieve it, all parties are normally *committed* to enacting the contents of the agreement (Jennings, 1993). In this work, we define a commitment as follows:

Definition 1.4. *A commitment is a pledge by an agent to ensure that the contents of the commitments are achieved through some actions (adapted from (Jennings, 1993)).*

The properties of the agreement reached (i.e. the type of actions agents commit themselves to) are dictated by the negotiation mechanism used (i.e. the protocol and strategies of agents). For example, if the mechanism allows agents to exhaustively explore the space of all possible agreements, the agreement chosen should be one that maximises all negotiating agents' utilities. In contrast, if the negotiation mechanism only allows an agent to accept or reject only one offer (e.g. in take it or leave it negotiation), the agreement may not be the most efficient one that could be obtained. Moreover, the type of mechanism chosen by the system designer may, in turn, depend on a number of factors, among which we note the following:

- The context of application — while some applications give an upper hand to the system designer to formulate a protocol that meets certain criteria (wanted by the designer), other applications may give more control to the individual agents' owners. For example, in selling licenses for bandwidth to telecommunication companies, a government agency (the system designer) may decide on a particular protocol that the companies need to comply with in placing their offers and, in so doing, elicits their true preferences and maximises the agency's profit (Krishna, 2002). On the other hand, traders in a stock market have to decide on their own (negotiation) strategies in order to get the best profit in the system given the rules that are in place.

- The *uncertainty* prevailing in the application — in most applications negotiations have to take place in an environment where there is a degree of uncertainty. In this context, uncertainty about a particular property or attribute means that there is a lack of information about that property or attribute and there is no statistical model for this. For example, agents may be uncertain about their exact preferences or about the actions they can perform in the environment. Agents may also be uncertain about their opponents' reliability (i.e. how good they are at doing what they say they can do) and their honesty (i.e. whether they tell the truth about the information have). In such cases, the protocol and the agents' strategies used for the negotiation will have to take these into account if the agents are to come to acceptable outcomes. Such uncertainties can be reduced in a number of ways including, but not limited to:

– Developing decision making models that allow agents to model those attributes or properties liable to uncertainty. In such contexts, we expect agents to use decision theoretic techniques such as statistics (Savage, 1954) or fuzzy reasoning (Zadeh, 1965; Mamdani, 1977) that permit such a modelling.

– Adapting the protocol to permit agents to reduce the number of variables over which the uncertainty applies. This may involve using a protocol that forces the agents to reveal all the information available to each of them (Krishna, 2002) or constraining the number of actions that they may perform (Hovi, 1998; Mas-Colell et al., 1995),.

Given this, a number of automated negotiation mechanisms have been devised to cater for different contexts and uncertainties. We can broadly classify these into following categories (see figure 1.1):

- Bargaining — this typically involves the exchange of offers between the interacting agents until an agreement is reached (this is often termed 'negotiation' in some cases (Jennings, 2001; Faratin et al., 1998)). In this context, each offer implies a conditional commitment on the part of the sending agent that it will enact the contents of the offer if and only if the recipient sends an 'agree' message. The contents of the offer or the negotiation object can vary from the very simple (e.g. based on price or quality only) to the extremely complex (e.g. involving trade-offs between price and quality) (Klein et al., 2003; Faratin et al., 2002). The negotiation object may also be dynamically changed by adding other issues during the negotiation process or by constraints imposed during other (concurrent or previous) negotiation encounters (games).

  Bargaining is appealing in situations where it is not possible to have a central authority that can generate an outcome that maximises the utility of all interacting agents. Also, bargaining protocols do not usually assume known preferences, reliability levels, action sets, or degree of honesty of the agents. Typically they only impose the sequence of exchange of offers (e.g. alternating offers or 'take it or leave it') or the participation rules that determine when agents are allowed to leave the negotiation or send offers for example. In such cases, these uncertainties are left mostly to the agent designers to model and use in their bargaining strategy (Faratin et al., 1998; Jennings et al., 2001) (i.e. in this case a strategy is a mapping from the history of offers to the next offer to be generated). To this end, the agents' owners may use some form of heuristic that provide general rules on how to add issues to the negotiation object, the type of offer to be

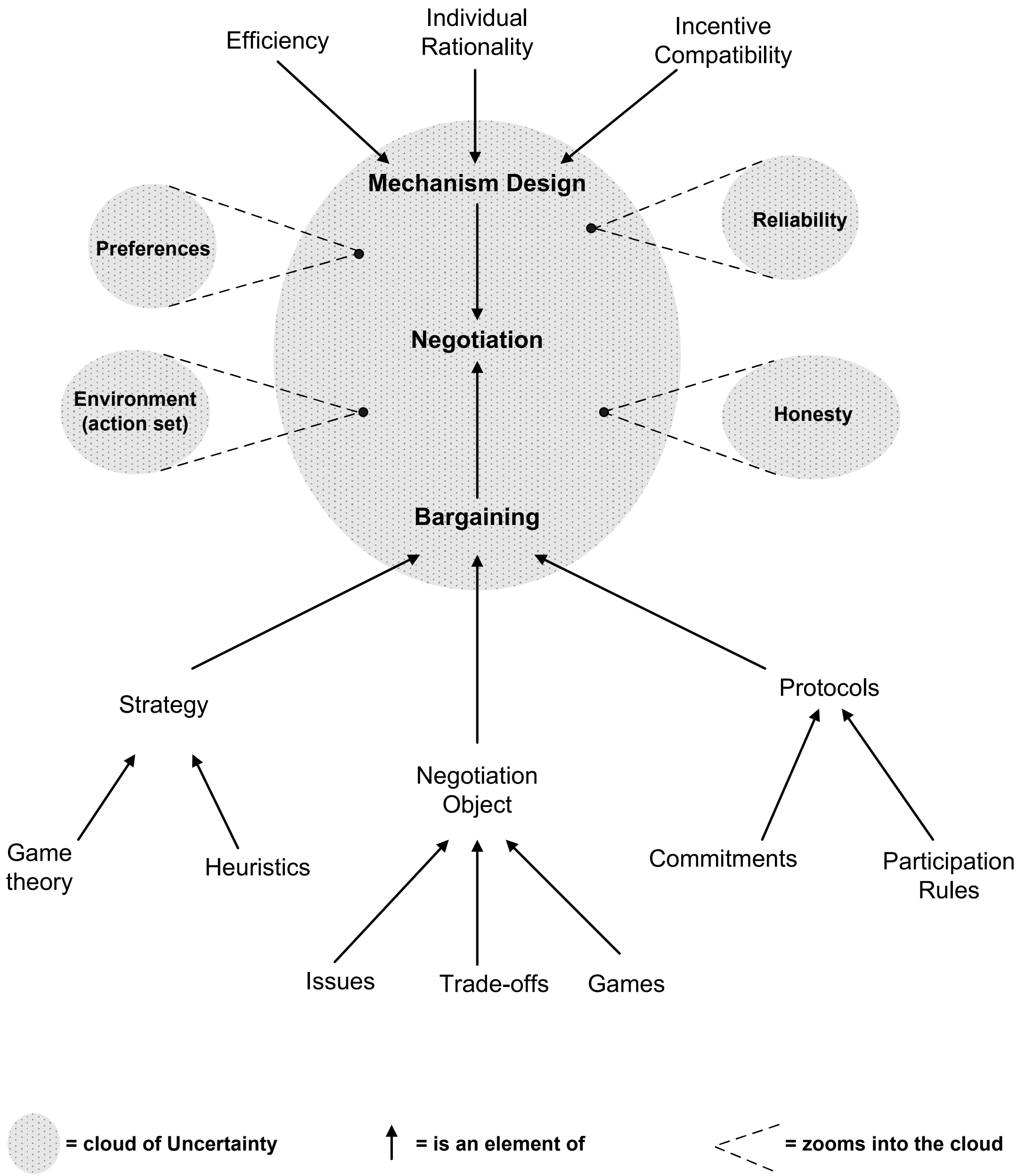

Figure 1.1: Approaches to negotiation in multi-agent systems and the cloud of uncertainty covering various aspects of the interaction.

sent, or the trade-offs that can be made between different issues. The way these different functions are performed define the agent's negotiation *stance* (i.e. how it shapes the negotiation encounter to its advantage). Heuristics generally try to reach good outcomes (i.e. those that give a high positive utility to the participating agents) rather than optimal ones (Jennings et al., 2001). In contrast, optimal outcomes that maximise the sum of the utility of participating agents are usually sought by game-theoretic techniques (Nash, 1953; Muthoo, 1999). In this context, optimal outcomes are those that maximise the sum of the utility of participating agents. To achieve this outcome, the agents' preferences and all their possible actions are usually assumed to be known. However, as can be seen, such approaches often make overly strong assumptions about the availability of information about the agents' private preferences and action set.

- Mechanism Design (MD) — this involves the development of a protocol specifying an exact sequence (and number) of actions (imposed by the system designer) to ensure that agents act in such a way that the resulting behaviour satisfies certain properties sought for by the system designer (Dash et al., 2003). To this end, the system designer assumes that the agents present in the system interact in a *game-theoretic* way (meaning that each agent models the effect of its actions on other agents' actions). The mechanism thus devised is to ensure that, at *equilibrium*, the intended properties are satisfied. The equilibrium here determines the state reached when all agents choose their utility maximising course of action and the main properties sought for by such mechanisms include: (i) pareto efficiency (i.e. maximising the sum of the utility of all agents in such a way that no other allocation exists where an agent gains more utility and no other agent is worse off); (ii) incentive compatibility (i.e. enforcing truthful revelation about the agents' preferences or other attributes); and (iii) individual rationality (i.e. agents are better off participating in the mechanism than opting out). To achieve such properties, game-theoretic mechanisms generally assume a completely known action set and that each agent knows its preferences perfectly (but not those of its opponent). To achieve such outcomes, the system designer provides incentives to agents to behave in a certain way through the specification of a payment scheme (i.e. how payments are made to agents which sell goods) and an allocation scheme (i.e. how goods are allocated to agents which pay for them) that takes into account the utility-maximising nature of agents. Usually, the protocols used in mechanism design imply a centralised authority that regiments the interactions (i.e. decides the agreements for the agents after knowing their preferences).

In general, both bargaining and mechanism design are subject to some uncertainty regarding similar or different attributes. For example, mechanism design reduces the uncertainty about the agents' preferences by enforcing a protocol which elicits these preferences. In contrast, bargaining seeks to elicit these preferences through an iterative exchange of offers which is not guaranteed to find an agreement that satisfies the agents' preferences. Therefore, as shown in figure 1.1, there exists a number of attributes that are subject to uncertainty and we view these as a cloud that envelops the negotiation process. Here we will concentrate on the attributes that most obviously affect negotiations such as:[3]

- Honesty — in competitive interactions agents may lie about their preferences or reliability in order to maximise their utility and this may, in turn, lead to inefficiency in the system. In such cases, the system designer needs to provide the right incentives to elicit truthful revelation of such information. This is usually achieved through engineering the protocol using some form of game theoretic analysis (i.e. mechanism design). In cases where this is not possible, agents may analyse the honesty of their opponents over multiple encounters and avoid those that are most dishonest in the long run.

- Reliability — in cases where a negotiation opponent's reliability of performing a particular task is not perfect, an agent might want to add some more stringent conditions to the agreement reached between them (e.g. specify a quality standard to be met or a compensation to be paid if expectations not met). This aims to make sure that the enactment of the agreement by the opponent is in line with what the agent expects. In such cases, in order to be able to analyse the reliability of an opponent, the agent may need to model this attribute statistically over multiple encounters and elicit a decision from that model at negotiation time. In this context, the reliability and honesty of agents is captured through the concept of *trust* (see chapter 3 for more details).

  Definition 1.5. *Trust is a belief an agent has that the other party will* do what it says it will *(being honest and reliable) or* reciprocate *(being reciprocative for the common good of both), given an* opportunity to defect *to get higher payoffs (adapted from (Dasgupta, 1998)).*

  Thus, through a trust model, it is possible to capture the probability of losing utility in an interaction with a particular agent by virtue of its trustworthiness (i.e. its reliability and honesty). Hence, through the use of a trust model, the risk[4] that agents incur in interactions can be significantly reduced.

- Preferences — when each agent in a negotiation encounter knows its opponents preferences, the outcome is usually easy to predict according to

[3]Other attributes, such as the communication mechanism used or the computational capability of the agents, are also subject to uncertainty, but in this thesis we will assume these are already factored into the decision making models of the agents.

[4]We conceive of an environment as being prone to uncertainty, when every possible event in the environment has an equal chance of happening. Risk, instead, arises when there is a probability that an event causing some utility loss will happen (Zeckhauser and Viscusi, 1990). These probabilities can be hard to estimate especially in the types of open distributed systems in which we are interested.

game theory (Mas-Colell et al., 1995). In mechanism design, the protocol is usually devised in such a way that these preferences are elicited. However, when preferences are not known and agents are in a bargaining encounter, they have to use efficient techniques to search the space of offers that meets their opponent's preferences. To assist in this process, the agents could also exchange more information (on top of an offer) which gives partial information about their preferences (i.e. without completely revealing them).

- Environment (action set) — when agents do not know each other's possible actions, it is hard to act strategically (as per game theory) to find an agreement (which dictates a set of actions to the participants) that maximises the utility of participating agents. Moreover, if the space of all possible actions is very large, negotiating agents may find it computationally hard to find a solution in a negotiation encounter. In such cases, the system designer might need to formulate a protocol that reduces the space of actions that agents need to search to find an agreement.

Against this background, in this thesis we aim to develop models that can reduce the impact of the above uncertainties on the effectiveness of bargaining and mechanism design techniques. In general, this can be achieved either by engineering new protocols or enriching the strategy of an agent in order to make the system, as a whole, more robust to uncertainty.

In more detail, in bargaining models in multi-agent systems, the uncertainty about preferences and the environment are increasingly being researched using a new class of techniques, here termed *argumentation-based negotiation* techniques, of which persuasive negotiation is a special category (see chapter 2). These models attempt, in various ways, an exploration of agents' preferences and actions. Currently, such models limit themselves to very abstract implementations (i.e. make no connection to a real application). Moreover, no existing agent-based bargaining model deals with the uncertainties underlying the reliability of agents or the honesty of agents. Similarly, in mechanism design where action sets are assumed to be known and honesty is elicited, some attention has been given to the uncertainty with respect to preferences of agents (Mas-Colell et al., 1995). However, there is a dearth of mechanisms that deal with uncertainty about the reliability of agents.

Given these lacunae, we aim to develop a new persuasive negotiation mechanism that aims to achieve better outcomes in less time than current bargaining techniques. To this end, we will clearly specify both the protocol and the strategies of the participating agents in such a way that the uncertainty about the agents' action sets and preferences is reduced. We also aim to develop modelling techniques, based on the concept of trust, that can be used by agents to reduce the uncertainty they have about their counterparts' reliability and honesty both in bargaining and mechanism design. In so doing, we will develop mechanisms that can generate better outcomes than current models when faced with uncertainty. Finally, we aim to show the applicability of our models by providing an example application where our persuasive negotiation mechanism and trust model can be used.

In the following sections we outline the landscape within which we develop our models. We will therefore describe issues that need to be dealt with in the area of trust and argumentation-based negotiation respectively.

1.3 Trust in Multi-Agent Systems

Broadly speaking, there are two main approaches to trust in multi-agent systems which we will focus on in this thesis. Firstly, to allow agents to trust each other, there is a need to endow them with the ability to reason about the reciprocative nature, reliability or honesty of their counterparts. This ability is captured through trust models. Such models aim to enable agents to calculate the amount of trust they can place in their interaction partners. A high degree of trust in an agent would mean it is likely to be chosen as an interaction partner and (possibly) a reciprocative strategy used towards it over multiple interactions in order to elicit the best pay-off in the long run (Axelrod, 1984). Conversely, a low degree of trust in an agent would result in it not being selected (if other, more trusted, interaction partners are available) or a non-reciprocative strategy adopted against it over *multiple interactions* (if there is no better alternative). In this way, trust models aim to guide an agent's decision making in deciding on how, when, and who to interact with. However, in order to achieve this, trust models initially require agents to gather some knowledge about their counterparts' characteristics. This can be achieved in a number of different ways including: (i) through inferences drawn from the outcomes of multiple direct interactions with these partners forming the agent's *confidence* in them or (ii) through indirect information provided by others forming the *reputation* of these partners. The combination of an agent's confidence and reputation measures (through some decision mechanism) can then be used to derive a general notion of trust that the agent has in its counterparts.

Secondly, while trust models pertain to the reasoning and information gathering ability of agents, the other main approach to trust concerns the design of protocols of interactions (i.e. through mechanism design techniques). As stated in section 1.2, one of the main aims of MD is to devise systems that are incentive compatible. This is normally achieved by providing the right incentives in the form of payments that are made from the mechanism to the agents involved in it. Thus, agents are compelled to be honest by the system.

From these two perspectives, it can be seen that trust pervades multi-agent interactions at all levels (i.e. at the protocol level and at the agent's reasoning level). With respect to designing agents and open multi-agent systems we therefore conceptualise trust in the following ways:

- **individual-level trust**, whereby an agent has some beliefs about the honesty, reliability, or reciprocative nature of its interaction partners.

- **system-level trust**, whereby the actors in the system are forced to be honest by the rules of encounter (i.e. protocols and mechanisms) that regulate the system.

The above approaches can be seen as being complementary to each other since they suit different contexts. Thus, while protocols aim to ensure the honesty of agents at the system level, they are limited in that they require a central authority (to compute outcomes or receive private information) and assume agents are completely reliable. In contrast, where the system cannot be completely centralised and agents cannot be assumed to be completely reliable, trust models at the individual level provide an alternative approach to measuring trust in a distributed fashion and are only limited by the agents' own sensing and reasoning capability (see chapter 3 for more details).

As can be seen from figure 1.2, while the individual level trust models enable an agent *to reason* about its level of trust in its opponents, the system level mechanisms aim to *ensure* that these *opponents' actions* can actually be trusted. In more detail, using their trust models, agents can:

- reason about strategies to be used towards trustworthy and untrustworthy interaction partners (e.g. being reciprocative or selfish towards them) given a calculation of payoffs over future interactions (i.e. using learning and evolutionary models).
- reason about the information gathered through various means (e.g. either directly or through reputation models) about potential interaction partners (i.e. using reputation models).
- reason about the motivations and capabilities of these interaction partners to decide whether to believe in their trustworthiness (i.e. using socio-cognitive models).

In contrast, the mechanisms and protocols described (i.e. enforcing system-level trust) aim to force agents to *act and interact* truthfully by:

- imposing conditions that would cause them to lose utility if they did not abide by them (i.e. using trustworthy interaction mechanisms).
- using their reputation to promote their future interactions with other agents in the community or demote future interactions whenever they do not behave well (i.e. using reputation mechansims).
- imposing specified standards of good conduct that they need to satisfy and maintain in order to be allowed in the system (i.e. using security mechanisms).

In general, these two approaches to trust have, however, rarely been used to deal with the uncertainties that arise in negotiation (except in the process of partner selection, see chapter 3 for more details). For example, in bargaining no trust modelling technique has been devised to allow agents to influence agreements according to the believed reliability or honesty of their counterparts.

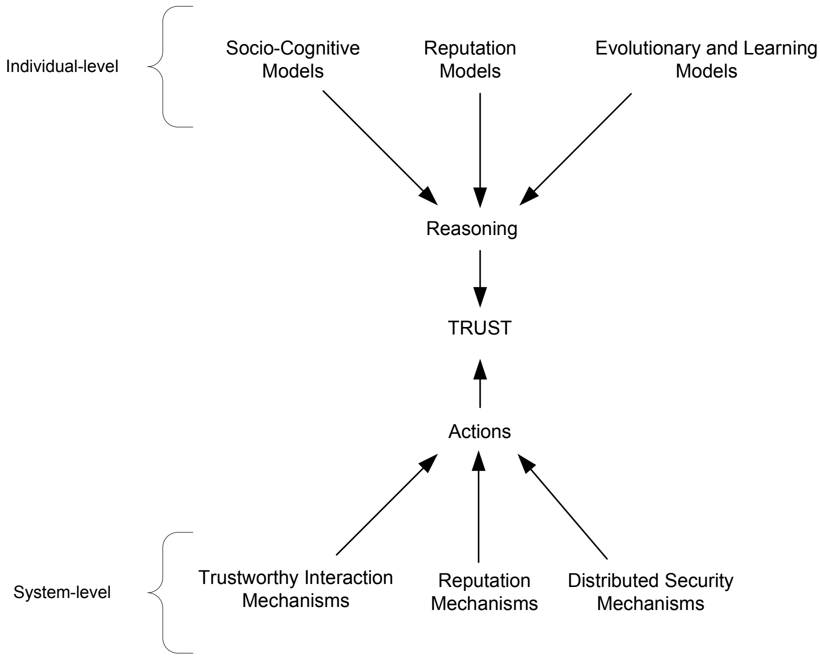

Figure 1.2: A classification of approaches to trust in multi-agent systems.