

# Uncertain Agent Verification through Probabilistic Model-Checking

Paolo Ballarini, Michael Fisher, and Michael Wooldridge

University of Liverpool, Liverpool L69 7ZF, UK,  
{paolo,michael,mjw}@csc.liv.ac.uk

**Abstract.** In many situations an agent’s behaviour can sensibly be described only in terms of a distribution of probability over a set of possibilities. In such case (agents’) decision-making becomes probabilistic too. In this work we consider a probabilistic variant of a well-known (two-players) Negotiation game and we show, first, how it can be encoded into a Markovian model, and then how a probabilistic model-checker such as PRISM can be used as a tool for its (automated) analysis. This paper is meant to exemplify that verification through model-checking can be fruitfully applied also to *uncertain* multi-agent systems. This, in our view, is the first step towards the characterisation of an automated verification method for probabilistic agents.

## 1 Introduction

Because of their notorious complexity, multi-agent systems’ run-time behaviour is extremely hard to predict and understand and, as a result, agents’ verification is a hard task. In recent years significant research effort has been invested in the development of *formal methods* for the specification and verification of multi-agent systems [14] and model-checking has been proved to be a possibility in that respect. Bordini *et al.*, for example, have shown how LTL model-checking techniques [13] can be applied, by means of the SPIN model-checker [9], to the verification of a specific class of *non-probabilistic* rational agents (i.e. BDI agent-systems expressed as *AgentSpeak* programs) [3]. In [5], Dix *et al.*, argue that in several real-life situations an agent behaviour (i.e. the state an agent is in at a certain time) may be known with a given degree of uncertainty. As a consequence a specific language for *probabilistic* agents (i.e. a language that allows for a quantification of such uncertainty) is needed and that is what the authors develop throughout their work. Given that probability distributions are sensible means to describe the uncertainty between several possibilities (i.e. several potential successor states of a given, current state) then the type of analysis involved must also be a *probabilistic* one. In a probabilistic framework, the analyst is interested in discovering the measure with which certain properties (either negative or positive ones) are likely to be true. If we refer to a simple negotiation game where two players, a seller and a buyer, bargain over a single item, then a player’s decision may well be considered a probabilistic one, as an expression of the player’s uncertainty towards their opponent’s behaviour. In such a situation it

is relevant to assess how uncertainty affects the negotiation outcome (hence players’ payoff), which is important in being able to answer questions such as: “what is the probability that an agreement will be reached at all?”, “how likely is it that the bargained item will be sold to a given value  $x$ ?”, and “what type of strategy gets a player a better *expected payoff*?”.

In this work we show how a probabilistic model-checker, such as PRISM[12], can be used as a tool to find answers to those type of questions, once the considered probabilistic agent system has been mapped onto a Markovian model, specifically a Discrete Time Markov Chain (DTMC). So as LTL model-checking, through SPIN, has been proved to be suitable for the analysis of *non-probabilistic* multi-agent systems, we show here that PCTL model-checking [8], through PRISM, is a proper tool for verifying *probabilistic* agent systems. What is still lacking, at present, is an automatic translation of probabilistic agent programs, such as, for example, the ones proposed in [5]. into the Reactive Module language, the input formalism for PRISM. This will be the focus of our future work. The remainder of the paper is organised as follows: in the next Section a brief introduction to the PCTL model-checking and to the PRISM tool is presented. In Section 2 we first introduce the probabilistic version of the Negotiation game we have considered, then we describe the results we have obtained by analysing specific PCTL formulae through PRISM. The final section summarises our analysis and lists directions for future work.

## 2 Alternating-Offers Negotiation Framework

In this section we present the Negotiation framework we have considered in our work and describe the probabilistic extension we have introduced and studied. The Bargaining process we refer to is the Alternating-Offers one discussed in [11] and [6]. In the basic formulation of such a game, two players bargain over a single item by alternatively throwing an agreement proposal and making a decision over the opponent’s last proposed value. Bargainers’ interest is clearly conflicting with the *seller* aiming to maximise the outcome of negotiation (i.e. the *agreed-value*) and the *buyer* aiming to minimise it. Players’ behaviour is characterised by a strategy which essentially determines two aspects: a player’s next offer value and a player’s *acceptance condition* (which describes if the opponent’s most recent proposal is going to be accepted/rejected). Negotiation analysis aims to study properties of the players’ strategies and, in particular, is interested in addressing the existence of dominant strategies and strategic equilibrium. However this type of analysis is not tractable in incomplete information settings, which is what we are considering in this paper. Nonetheless assessing the effectiveness of strategies for automated negotiation is relevant from an AI perspective. In the following we briefly describe the family of time-dependent strategies for automated negotiation between agents introduced by Faratin *et al.* in [6], which is the one we use in our probabilistic framework. Players’ strategies depend on 2 intervals, respectively  $[min_b, max_b]$ , where  $max_b$  is the buyer’s reservation price and  $min_b$  is the buyer’s lowest acceptable offer (i.e.  $min_b=0$ ), and  $[min_s, max_s]$ ,

where  $min_s$  is the seller's reservation price and  $max_s$  represents the seller's upper bound of a valid offer. A player's initial offer is his most profitable value, which is  $min_b$ , for the buyer and  $max_s$  for the seller. The basic idea of this family of strategies is that a player concedes over time and the pace of concession determines the type of negotiation strategy. Time is thought of as a discrete quantity indicating negotiation periods (i.e. players' turn),  $T$  denotes the time-deadline and a player's offer at time  $t \in [0, T - 1]$ , depicted, respectively, as  $x_b^t$  and  $x_s^t$ , is defined as:

$$x_b^t = min_b + \alpha_b(t)(max_b - min_b) \quad (1)$$

$$x_s^t = min_s + (1 - \alpha_b(t))(max_s - min_s) \quad (2)$$

where  $\alpha_i(t)$ ,  $i \in \{b, s\}$  is the time-dependent function in which the constant  $\beta$  determines the conceding pace of player  $i$ :

$$\alpha_i(t) = \left(\frac{t}{T}\right)^{\left(\frac{1}{\beta}\right)}$$

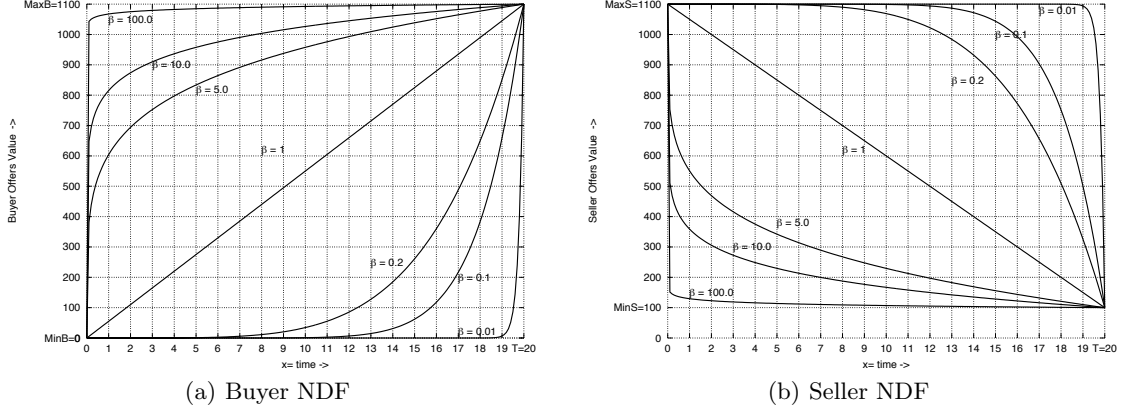
By varying  $\beta \in (-\infty, \infty)$  a whole family of offer functions, also called Negotiation Decision Functions (NDFs), can be obtained (see Figure 1(a) and Figure 1(b)). For  $\beta > 1$ , strategies are referred to as *Conceder* strategies whereas, for  $\beta < 1$ , strategies belong to the so-called *Boulware* class. With  $\beta = 1$ , on the other hand, we have linear strategies for which a player's offer is monotonically incremented (decremented) over time. Finally, the decision making on a received offer is driven by profitability. Hence the buyer will accept the seller's offer at time  $t$  if, and only if, it is more profitable than his next offered value. Formally:  $x_s^t$  is accepted by  $b$  if, and only if,  $x_s^t \leq x_b^{t+1}$  (similarly  $x_b^t$  is accepted by  $s$  if, and only if,  $x_b^t \geq x_s^{t+1}$ ). For the sake of simplicity we have chosen specific settings for the bargainer,s bounds, namely:  $min_b = 0$ ,  $max_b = 1100$  and  $min_s = 100$ ,  $max_s = 1100$ . Such a choice is motivated by the fact that, in our modelling formalism (i.e. the Reactive Module language of the PRISM model-checker) only integer variables are admitted, hence a wide enough interval shall be considered in order to be able to observe the effect of different strategy slopes (having in mind that in the integer domain 1 is the smallest possible slope value). Finally we consider  $T = 20$  as the time deadline for the negotiation process.

## 2.1 Probabilistic decision making

As a result of the decision mechanism introduced in [6], an agreement between the buyer and the seller is reached only if the offer functions (Figure 1(a) and Figure 1(b), respectively) cross each other within the time deadline (i.e.  $T$ ). If this is the case the result of negotiation is positive and a deal is implemented at the crossing point<sup>1</sup>.

<sup>1</sup> In a monetary negotiation the actual agreement would correspond to rounding up/down to the closest unit of exchange (a penny valued price, if we are referring to the UK market) of such crossing point, depending on the accepting agent.

#### IV



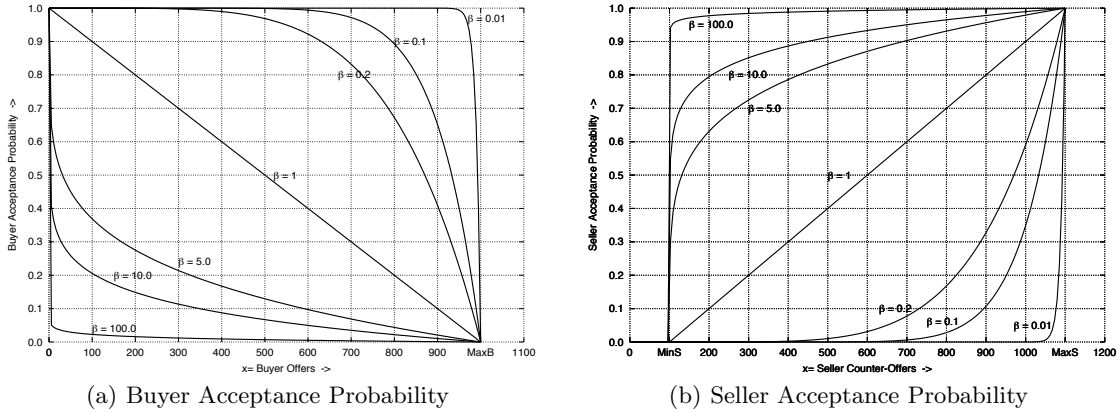
**Fig. 1.** Buyer-Seller Negotiation Decision Functions

In our framework we introduce uncertainty by making a player's decision a probabilistic function of the offered value. By doing so, we enlarge the semantics of the negotiation model so as to cope with the inherent uncertainty that may characterise a player's behaviour when it comes to deciding over a deal proposal. Such uncertainty may reflect several factors such as: lack of information about the opponent's preferences/strategy, a change of a player's attitude as a result of environmental changes and/or time passing. In order to reduce the complexity of our model, uncertainty, rather than explicitly being a function of time, is quantified with respect to the offered value only (however, since we are adopting time-dependent offer functions the acceptance probability is also, indirectly, dependent on time). Taking inspiration from the NDFs (1) and (2), we introduce the acceptance probability functions,  $S\_AP()$  for the seller and  $B\_AP()$  for the buyer, in the following way:

$$S\_AP(x_b^t) = \begin{cases} 0 & \text{if } (x_b^t \leq \min_s) \\ \left( \frac{x_b^t - \min_s}{\max_s - \min_s} \right)^{\frac{1}{\beta_s}} & \text{if } (x_b^t > \min_s) \wedge (x_b^t < x_s^{t+1}) \\ 1 & \text{if } (x_b^t \geq x_s^{t+1}) \end{cases} \quad (3)$$

$$B\_AP(x_s^t) = \begin{cases} 0 & \text{if } (x_s^t \geq \max_b) \\ 1 - \left( \frac{x_s^t}{\max_b} \right)^{\frac{1}{\beta_b}} & \text{if } (x_s^t < \max_b) \wedge (x_s^t > x_b^{t+1}) \\ 1 & \text{if } (x_s^t \leq x_b^{t+1}) \end{cases} \quad (4)$$

Each definition depends on the parameter  $\beta_i$ ,  $i \in \{b, s\}$ , whose value determines the type of function. A player's attitude (in decision making) is called *conservative* if the likelihood of accepting low profitable offers is very small. Thus a conservative attitude, for the buyer, corresponds to  $\beta_b \gg 1$ , whereas, for the seller, is given by  $\beta_s \ll 1$ . The parametric definition for the acceptance proba-



**Fig. 2.** Buyer-Seller Acceptance Probability Functions

bility functions allows us to assess the effect of different player's attitudes on the negotiation outcome (as we will see we verify several configurations of our model corresponding to different combinations,  $(\beta_s, \beta_b)$ , of the acceptance probability functions). Finally we point out that, by replacing the decision-making model of Faratin *et al.* with the probabilistic mechanism determined by (3) and (4), we still maintain the semantics of the original; in fact through (3) and (4) we have that the acceptance of an offer whose utility is higher than the corresponding counter-offer's one is certain.

### 3 The PRISM model of negotiation

In this section, we describe the Markovian model of the negotiation framework (with probabilistic behaviour) introduced in the previous section. We developed a Discrete Time Markov Chain (DTMC) which represents the behaviour of two bargainers alternatively throwing offers according to the NDF families (1) and (2) and adopting the probabilistic decision mechanism described by the family of functions (3) and (4). We have used the PRISM model-checker to implement and verify the DTMC model of negotiation. Before describing some details of the DTMC model we provide brief background to the PRISM tool and to the Probabilistic Computational Tree Logic (PCTL), the temporal logic used for the verification of DTMC models (for a more detailed treatment of the subject the interested reader is referred to the vast literature, examples of which are [1, 8, 10, 12]).

#### 3.1 The PCTL logic and PRISM

Markov processes [7] are a subclass of stochastic processes suitable for modelling systems such that the probability of possible future evolutions depends uniquely

on the current state rather than on its *past history*. A system's timing is also taken care of with Markov chain models leading to either DTMCs, for which time is considered as discrete quantity, or Continuous time Markov chains (CTMC), where time is continuous. Both DTMC and CTMC models can be encoded in PRISM by means of a variant of the *Reactive Modules* formalism of Alur and Henzinger [1].

For issuing queries, PRISM uses either the *Probabilistic Computational Tree Logic* (PCTL) [8], if the underlying model is a DTMC, or the Continuous Stochastic Logic (CSL) [2] for referring to CTMC models. These languages are variations of the temporal logics used for more conventional LTS model checking (i.e. the CTL of Clarke *et al.* [4]). Formally a labelled DTMC is defined as follows:

**Definition 1.** *Given a set of atomic propositions  $AP$ , a labelled DTMC  $\mathcal{M}$  is a tuple  $(S, \mathbf{P}, L)$  where  $S$  is a finite set of states,  $\mathbf{P}: S \times S \rightarrow [0, 1]$  is the transition probability matrix such that  $\forall s \in S, \sum_{s' \in S} \mathbf{P}(s, s') = 1$  and  $L: S \rightarrow 2^{AP}$  is a labelling function.*

A *path* in a given DTMC  $\mathcal{M} = (S, \mathbf{P}, L)$  and its *probability measure* are formally characterised in the following definitions.

**Definition 2.** *A path  $\sigma$  from state  $s_0$  is an infinite sequence  $\sigma = s_0 \rightarrow s_1 \rightarrow \dots \rightarrow s_n \rightarrow \dots$  such that  $\forall i \in \mathbb{N}, \mathbf{P}(s_i, s_{i+1}) > 0$ . Given  $\sigma$ ,  $\sigma[k]$  denotes the  $k$ -th element of  $\sigma$ .*

The probability measure of a set of infinite paths with common finite prefix  $\sigma \uparrow n = s_0 \rightarrow \dots \rightarrow s_n$  is defined as the product of the probability of the transitions in the prefix  $\sigma \uparrow n$ .

**Definition 3.** *Let  $\sigma \uparrow n = s_0 \rightarrow \dots \rightarrow s_n$  be a finite path of  $\mathcal{M}$ . The probability measure of the set of (infinite) paths prefixed by  $\sigma \uparrow n$  is*

$$Prob(\sigma \uparrow n) = \prod_{i=0}^{n-1} \mathbf{P}(s_i, s_{i+1})$$

if  $n > 0$ , whereas  $Prob(\sigma \uparrow n) = 1$  if  $n = 0$ .

The PCTL language used to refer to DTMC models is formally introduced in the next definition.

**Definition 4 (PCTL syntax).** *For a set of atomic propositions  $AP$ , the syntax of PCTL state-formulae ( $\phi$ ) and path-formulae ( $\varphi$ ) is inductively defined as follows:*

$$\begin{aligned} \phi &:= a \mid tt \mid \neg\phi \mid \phi \wedge \phi \mid \mathcal{P}_{\triangleleft p}(\varphi) \\ \varphi &:= \phi \ U^{\leq t} \ \phi \end{aligned}$$

where  $a \in AP$ ,  $p \in [0, 1]$ ,  $t \in \mathbb{N}^* \cup \{\infty\}$  and  $\triangleleft \in \{\geq, >, \leq, <\}$ ,

The PCTL semantics is as the CTL one except for probabilistic path-formulae. The formula  $\mathcal{P}_{\leq p}(\varphi)$  is satisfied in a state  $s$  if, and only if, the probability measure of paths starting at  $s$  and satisfying  $\varphi$ , denoted  $Prob(s, \varphi)$ , fulfils the bound  $\leq p$ . Formally:

$$s \models \mathcal{P}_{\leq p}(\phi' U^{\leq t} \phi'') \quad \text{iff} \quad Prob(s, (\phi' U^{\leq t} \phi'')) \leq p$$

where the semantics of  $(\phi' U^{\leq t} \phi'')$  with respect to a path  $\sigma$  is defined as:

$$\sigma \models \phi' U^{\leq t} \phi'' \quad \text{iff} \quad \exists i \leq t : \sigma[i] \models \phi'' \wedge \forall j < i, \sigma[j] \models \phi'$$

Essentially, PCTL extends CTL's expressiveness in two ways: by allowing a continuous *path-quantification* (i.e. CTL *existential* and *universal* path quantifiers are replaced by a single *continuous* quantifier, namely  $\mathcal{P}_{\leq p}$ )<sup>2</sup> and by introducing a *discrete* time-bounding for Until-formulae. For a complete treatment of PCTL we refer the reader to [8].

### 3.2 Modelling Probabilistic Negotiation with PRISM

In this section we describe how we have built the DTMC model of negotiation through PRISM. For the sake of space we do not include any sample from the actual PRISM source file. However we discuss the most relevant characteristics of the resulting models, some of which are a consequence of the expressiveness constraints of PRISM input language. The model consists of three modules: a *Timer* (to keep track of turns); a *Buyer*; and a *Seller*. The modules' parallel composition is synchronised over two events: time-elapsing (driven by the timer) and offer proposals (alternatively driven by the buyer and seller).

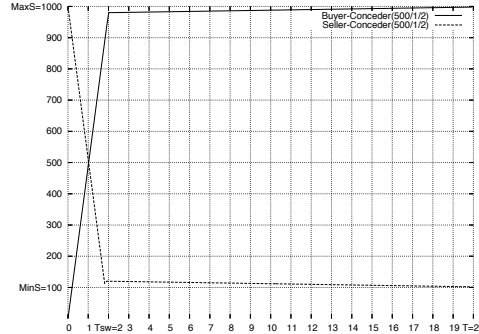
**Piecewise approximation of the NDFs:** PRISM input language allows for representing (finitely many) integer values only (i.e. only finite-states system can be modelled). As a result the continuous NDFs (1) and (2) are, in our PRISM models, approximated by (sampling from) piecewise linear functions consisting of two pieces<sup>3</sup> (see, for example, Figure 3). The offer function has three parameters: the slope of the first piece, the slope of the second piece and the boundary (switch time) between the pieces (hence a setting for a NDF approximation is given by a triple  $(n_1, n_2, n_3)$  where the first 2 elements represent the slopes of the first and second piece while the third element is the switch-time; for example, the curves in Figure 3 refer to a setting  $(500, 1, 2)$ ). The desired setting (boulware/conceder strategy) is chosen through model configuration so that different *tactic profiles* are verified.

<sup>2</sup> PCTL is a superset of CTL's as:  $E(\phi U \psi) \equiv \mathcal{P}_{>0}(\phi U \psi)$  and  $A(\phi U \psi) \equiv \mathcal{P}_{\geq 1}(\phi U \psi)$

<sup>3</sup> A two piece line is a good approximation for *extreme* bargaining tactics (i.e.  $\psi \sim 0$  or  $\psi \gg 1$ ), which is the type of non-linear tactics we address in this work. Less extreme strategies may be better approximated by multi-piece lines, which would require minimal modifications to our model in order to be coped with.

probability for each possible negotiation outcome are derived).

**Model’s configuration:** the model we developed is designed to be highly configurable through a number of constants. Before running a verification experiment a configuration is chosen by setting up: the Buyer and Seller NDFs (each of which requires three parameters, first-piece-slope, second-piece-slope, switch-time), the buyer and seller reservation-price and initial-offer, the buyer and seller acceptance attitude (i.e. the parameter  $\beta$  for both acceptance probability functions (3) and (4)) and the time-deadline (which we set to 20 for all experiments). In order to be able to assess the effect of substantially different NDFs (i.e. to compare how NDF’s slopes affect the result of negotiation) we had to allow for a wide enough *acceptance interval* (i.e. the interval  $[min_s, max_b]$ ). As a result we have chosen a default settings of  $[1, 1000]$  for such an interval. This allows us to sensibly verify the effect of slopes values up to 2 orders of magnitude different (i.e. 1-10-100). This is the reason why the graphs reporting the results of our analysis (see next section) refer to such a setting (the x-axis range is referred to  $[0, 1000]$ ).



**Fig. 3.** NDFs’ piecewise linear approximation

## 4 Analysis

In this section we describe the result of the verification performed through the PRISM model-checker on the DTMC model of Negotiation.

We observe that, the probabilistic decision making mechanism encoded within the bargaining DTMC model results in a probability distribution over the set of possible outcomes of the negotiation (i.e. the interval  $[min_s, max_s] \subset \mathbb{N}$ ), hence over a strategy profile’s payoff.



We have used the verification facilities of PRISM to evaluate the distribution of probability of two distinct aspects of the model behaviour: the value at which an agreement is reached, and the delay for reaching an agreement. These measurements can be automatically derived by running a number of PCTL formulae verifications (the so-called PRISM ‘experiment’ facility, by means of which a parametric PCTL formula is iteratively verified for each specified value of its parameters). The corresponding PCTL-like PRISM temporal formula for verifying the probability of reaching an agreement at value  $x$  is as follows:

$$P=?(tt U(\textit{agreement}) \wedge (\textit{PURCHASE}=x)), \quad (5)$$

whereas the temporal formula for determining the probability of an agreement to be reached at time  $t$  is:

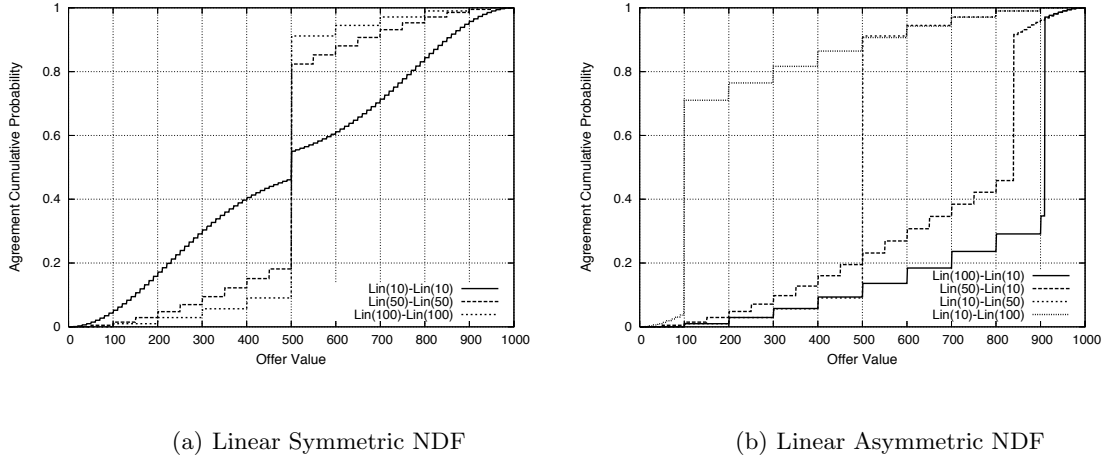
$$P=?(tt U(\textit{agreement}) \wedge (\textit{TIME}=t)). \quad (6)$$

The reader should not be misled by formulae (5) and (6), in which the PCTL syntax described by Definition 4 is slightly abused by means of the ‘=?’ notation. This is the way PRISM allows for specifying an experiment with respect to one or more parameters. For example, the result of running a PRISM experiment on (5), is that the probability of reaching (at some point in the future) a state in which an agreement is implemented at  $x$ , is computed for every value of the  $x$ -range which has given as input of the experiment (hence by choosing  $[min_s, max_b]$  as input range for an experiment over (5), we end up deriving the distribution of probability over the set of possible agreement values).

In the following we report about the results of the model analysis obtained by verification of (5) and (6) through PRISM experiments. These are grouped in three different categories, each one of which copes with a different aspect of the model analysis.

**Agreement distribution as a function of players NDFs:** here we discuss experiments which aim to assess how players’ NDFs affect the probabilistic outcome of negotiation. For this reason, in these experiments we have compared model’s configurations by varying combinations of NDFs while sticking with a specific (fixed) combination of Acceptance Probability Function (specifically the one corresponding to  $\beta_b = 0.2$  and  $\beta_s = 5$ ). The results we present are grouped according to different combination of NDFs. Each combination is denoted by a pair  $F_b(x_b/y_b), F_s(x_s/y_s)$  where  $F_a$  ( $a \in \{b, s\}$ ) denotes the type of function for player  $a$  (i.e. either *Lin*, *Boul*, or *Conc*), whereas  $x_a$  and  $y_a$  denote, respectively the first and second piece’s slopes<sup>4</sup>. Hence, for example, *Lin*(10)-*Lin*(100) denotes a profile for which both players are using a linear offer function, the buyer with slope 10 and the seller with slope 100, whereas the profile *Boul*(1/100)-*Conc*(100/1) corresponds to the buyer using a Boulware offer function with first slope 1 and second slope 100 and the seller using a conceder tactic with first and second slope respectively 100 and 1. Figure 4(a) and 4(b) compares the cumulative probability distribution (over the default interval,  $[1, 1000]$ , of possible

<sup>4</sup> as linear functions consist of a single piece, then for  $Lin_a$  only  $x_a$  is used.

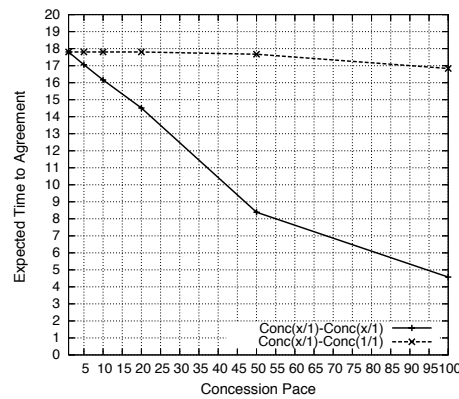


**Fig. 4.**  $Lin(x)$ - $Lin(y)$  NDF profiles

agreements), for several configurations of, respectively, symmetrical and asymmetrical, linear NDFs. Some general indications can be drawn from these pictures. For example if players use symmetrical NDFs then a higher concession pace tends to favour the seller. This is also confirmed by the expectation values which for the symmetrical case show a (slow) increasing trend  $Exp(Lin(10)) \sim 498$ ,  $Exp(Lin(50)) \sim 499$  and  $Exp(Lin(100)) \sim 500$  confirming the seller's advantage with fast concession behaviour. In case of asymmetrical NDFs (Figure 4(b)), instead, it is generally more convenient for a player to minimise his concession pace, as this is going to get him a more profitable expected outcome. For example if we compare the curves for profiles  $Lin(50)$ - $Lin(10)$ , and  $Lin(100)$ - $Lin(10)$  the probability tend to cumulate closer to the supremum of the interval (which is good for the seller). Again this is confirmed by looking at the expectation values, which show the same tendency, with  $Exp(Lin(50)Lin(10)) \sim 692$ , whereas  $Exp(Lin(100)Lin(10)) \sim 804$ .

**Agreement delay as a function of players NDFs:** here we discuss the effect that players' NDFs have on the delay for reaching an agreement. Again we consider several combinations of NDFs, but this time we deal with the verification of 6. Figure 5 depicts the expected time-to-agreement as a function of the players' concession pace. It indicates that a faster concession results in a quicker agreement. It should be noted that the discrepancy between the symmetrical case (i.e.

$Conc(x/1) - Conc(x/1)$  and the asymmetrical one (i.e.  $Conc(x/1) - Conc(1/1)$ ) is a due to the early crossing of NDF curves. In fact, when both player speed up concession at the same pace (symmetrical), NDFs curves intersect earlier (and for  $x > 25$  within the time-deadline  $T = 20$ ) than they do when only one player (asymmetrical) is increasing the pace of concession.



**Fig. 5.** Expected Time-to-Agreement as a function of concession pace

**Agreement distribution as a function of players uncertainty:** here we discuss the effect of players' uncertainty (i.e. Acceptance Probability functions) on the outcome of negotiation. We performed a number of experiments for the verification of (5), but this time we compared combinations of values for the parameters  $(\beta_b, \beta_s)$  of (3) and (4), while imposing a constant configuration for the NDFs. Figure 6(a) reports about the expected value of an agreement as a function of the  $\beta$  parameters of (3) and (4). We recall that, in that respect, a *conservative* attitude corresponds to  $\beta_b > 1$ , for the buyer, and to  $\beta_s < 1$  for the seller. The curves in Figure 6(a) allows for comparing the effect of increasing the *conservativeness* of a player while the other one's is maintained constant (in this specific case, linear). As expected, we can conclude that, conservativeness (in probabilistic decision making) is desirable for a player, as it improves the expected value of a deal (decreasing it when the buyer becomes more conservative and increasing when the seller becomes more conservative).

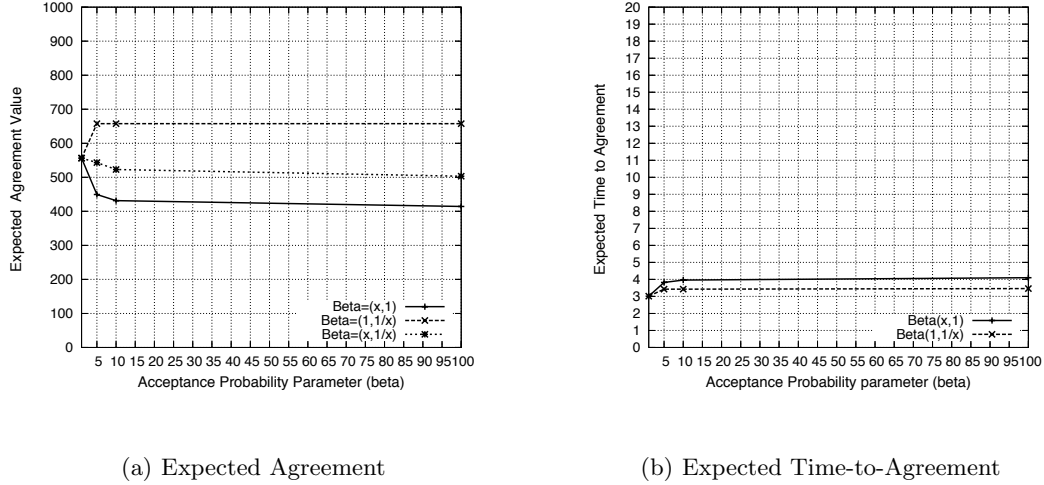


Fig. 6. Effect of uncertainty on Agreement value and Agreement delay

**Agreement delay as a function of players uncertainty:** similarly, here we discuss the effect that players' probabilistic decision making on the delay required for reaching an agreement. Figure 6(b) reports about the expected time-to-agreement as a function of the  $\beta$  parameter of (3) and (4). Again the indication of these results is confirms the intuitiveness, which is: a more conservative attitude results in a (slightly) longer expected delay. It should be pointed out that the small variability of curves in Figure 6(a) and Figure 6(b) is a direct consequence of the small variability of the acceptance probability functions (3) and (4) corresponding to extreme values of  $\beta$  (i.e.  $\beta_b \gg 1$  and  $\beta_s \ll 1$ ). Finally from Figure 6(b) one may conclude that the seller's conservativeness affects the delay for reaching an agreement slightly less than the buyer's does. This is, in fact is not true, and the difference between these two results is due to the fact that for our experiments we must impose an order for the bargaining (i.e. what player is going to start the negotiation). All the pictures reported in this paper refer to experiments in which the buyer is the starting player.

## 5 Conclusions

In this paper we have shown a different approach to the analysis of multi-agent system which can be used as an alternative to (or in conjunction with) analytical

methods and/or simulation. We have illustrated how the analysis of specific negotiation (mixed) strategies can be performed by means of probabilistic model checking when players' decision is made through a probabilistic device. This is achieved by developing an *ad hoc* probabilistic model which is then verified against probabilistic properties with the PRISM model checker. This analysis has helped in comparing the effect that several strategic variables has on two relevant features of the negotiation process: the (expected) value at which an agreement is reached and the (expected) time at which a deal is implemented. The results of our verification provided us with some insights about the behaviour of a system with uncertain players. Specifically we have shown that, a slower concession pace is generally preferable for both players (unless in the case of symmetrical strategies for which the seller is slightly advantaged by faster concession) as well as a conservative attitude in (probabilistic) decision making. With respect to the delay for reaching an agreement our analysis proves that faster concession paces (probabilistically) speed up the negotiation whereas a conservative decision making (slightly) slows it down.

The model of Negotiation we have developed here is strongly related to the one introduced by Li *et al.* in [11]. In [11], the authors study an alternating offers "uncertain" dynamic framework, where several agents may compete with each other (by setting aggressive reservation prices) for selling a single item/service to a (single) buyer. The uncertainty in that framework refers to two aspects: the availability of competing sellers (i.e. the time of arrival of a seller) and the reservation price of a newly arrived seller. The authors then introduce specific heuristics which allows the buyer to dynamically adjust his strategy as a consequence of changes in the environment (arrival of new sellers) and use simulation to obtain estimations of the framework behaviour. We would like to point out the main difference between our approach to model uncertainty and the one in [11]. In [11], what is uncertain is the environment behaviour (competitors may enter the market with a given probability after each time unit, but players' behaviour is not inherently probabilistic), whereas in our framework the uncertainty is represented into the players behaviour (by means of the probabilistic decision making mechanism). In a sense our model is more general as it abstracts away from the specific cause of uncertainty (a change in the environment such the arrival of a new player, for example), and encodes uncertainty directly within players' strategies. Furthermore our approach differs from the one studied in [11] with respect to the verification technique: we use model checking, through which an exhaustive/exact verification of the model is achieved, as opposed to simulation, which is based on estimations derived from non-exhaustive verification of the system model.

Following [11], future developments of this work include the extension of our approach to studying the effect of multiple, uncertain, competing players on the outcome of negotiation (such a model is currently under development). Finally we would like to stress that, to the best of our knowledge, this work provides a novice contribution which shows how a well established and effective automated verification technique as model-checking for the analysis of game-like scenarios.

**Acknowledgement:** This work was supported by an EC Marie Curie Fellowship under contract number HPMF-CT-2001-00065.

## References

1. R. Alur and T. Henzinger. Reactive modules. *Formal Methods in System Design*, 15(1):7–48, 1999.
2. C. Baier, B. Haverkort, H. Hermann, and J.P. Katoen. Model-checking algorithms for continuous-time Markov chains. *IEEE Trans. on Software Eng.*, Vol. 29(6):pp. 524–541, June 2003.
3. R. H. Bordini, M. Fisher, C. Pardavila, and M. Wooldridge. Model checking agentspeak. In *Proceedings of the Second International Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, Melbourne, Australia, 2003.
4. E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, 2000.
5. Jürgen Dix, Mirco Nanni, and V. S. Subrahmanian. Probabilistic agent programs. *ACM Trans. Comput. Logic*, 1(2):208–246, 2000.
6. P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Int. Journal of Robotics and Autonomous Systems*, 24(3-4):159–182, 1998.
7. W. Feller. *An introduction to probability theory and its applications*. John Wiley and Sons, 1968.
8. H. A. Hansson and B. Jonsson. A framework for reasoning about time and reliability. In *Proc. 10th IEEE Real-Time Systems Symposium*, pages 102–111, Santa Monica, Ca., 1989. IEEE Computer Society Press.
9. Gerard J. Holzmann. *The Spin Model Checker*. Addison Wesley, 2003.
10. M. Kwiatkowska, G. Norman, and D. Parker. Probabilistic symbolic model checking with prism: A hybrid approach. *International Journal on Software Tools for Technology Transfer (STTT)*, 2004.
11. C. Li, J.Giampapa, and K. Sycara. Bilateral negotiation decisions with uncertain dynamic outside options. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 36, Part C: Special Issue on Game-theoretic Analysis and Stochastic Simulation of Negotiation Agents(No. 1), 2006.
12. D. Parker. Prism web site. [www.cs.bham.ac.uk/~dxp/prism](http://www.cs.bham.ac.uk/~dxp/prism).
13. A. Pnueli. The temporal logic of programs. In *Proceedings of the Eighteenth IEEE Symposium on the Foundations of Computer Science*, pages 46–57, 1977.
14. M. Wooldridge. Agent-based software engineering. *IEE Proceedings on Software Engineering*, 144(1):26–37, 1997.