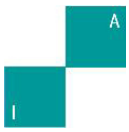
Inteligencia Artificial, 22(63) (2019), 135-149
doi: 10.4114/intartif.vol22iss63pp135-149



INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

The importance of context-dependent learning in negotiation agents

Dan Ezequiel Kröhling[1], Omar Chiotti[1], and Ernesto Martínez[1]

[1]INGAR (CONICET/UTN) - Instituto de Desarrollo y Diseño.
Avellaneda 3657, S3002GJC Santa Fe, República Argentina.

Abstract Automated negotiation between artificial agents is essential to deploy Cognitive Computing and Internet of Things. In this sense, the behavior of those negotiation agents depend significantly on the influence of environmental variables, facts, and events, which made up the context of the negotiation game. This context affects not only a given agent preferences and strategies, but also those of his opponents. In spite of this, the existing literature on automated negotiation is scarce about how to properly account for the effect of the context in learning and evolving strategies. In this paper, a novel context-driven representation of the negotiation game is introduced. Also, a simple negotiation agent that queries available information from context variables, internally models them, and learns how to take advantage of this knowledge by playing against himself using reinforcement learning is proposed. Through a set of episodes of our context-aware agent against other negotiation agents in the existing literature, it is shown that it makes no sense to negotiate without taking relevant context variables into account. Our context-aware negotiation agent has been implemented in the GENIUS tool. Results obtained are significant and quite revealing about the role of self-play in learning to negotiate.

Keywords: Agents, Automated Negotiation, Negotiation Intelligence, Internet of Things, Reinforcement Learning.

1 Introduction

Artificial intelligence has definitely entered the mainstream of business innovation [1, 22]. Huge progresses in the existing technology [21], new theories of intelligence [17, 24, 29], and the increasingly refined comprehension of biological brains of humans and animals [13, 28], have lead to the development of new mathematical models that tackle the problem of creating the so-called intelligent agents in our daily life. Some examples of these are [8, 11, 26].

A topic that has gained attention among AI experts in recent years is the implementation of intelligent negotiation agents. First of all, there is a human reason behind this: people is usually reluctant to get involved in negotiations. As Fatima et al. [10], taken from [3], put it: "When engaged in complex negotiations, people become tired, confused, and emotional, making naive, inconsistent, and rash decisions." This is a human condition: we could see it in our everyday life [12]. A second, yet more technological reason behind the creation of negotiation agents is to accomplish the promise of novel technologies such as Internet of Things [1] and Cognitive Computing [22].

Although great efforts are being made to automatize negotiations between artificial agents, some doubts remain about the design aspects of such artificial entities. In this sense, a number of approaches

ISSN: 1137-3601 (print), 1988-3064 (on-line)

to address this problem have been proposed [23, 7, 16, 31]. We consider the works of Fatima et al. [10] and Baarslag [4] a great compendium of the state of the art in this research area. In [10], differences between single and multiple issues, bilateral and multilateral negotiations are shown, with discrete and continue issues' values. Machine-machine and man-machine negotiations are addressed, together with different negotiation protocols, domains, and the selection of proper negotiation agendas. In [4], offering, accepting, and opponent modelling techniques are presented, alongside a framework to develop negotiation agents.

In spite of the progresses made so far, there is an issue in automated negotiation that, from our point of view, has not been properly accounted for in the design of negotiation agents. This is the importance of the context in negotiations, or the existence of key information that could provide a competitive advantage when used to predict and model the opponent by associative learning, including its strategy and perceptions/assumptions from the context. As an example of this, an agent could be i) informed about issues in the environment beyond the opponent himself, and ii) hypothesize about which information the opponent is actually using to make his predictions and learn. The importance of learning in automated negotiations has been previously recognized [32, 33], yet the context-awareness capability is not widely seen as a key issue [2, 19, 30]. Most of previous works circumscribe the agent learning to ad-hoc decision-making policies that may not capture appropriately the influence of the context on the outcome of a negotiation episode.

In order to clarify our point of view, let us discuss briefly some related work. In [2, 30], the context is represented through a fixed model, but any new variable that could change the course of the negotiation is discarded. Another example is given in [19]. Although a novel approach to model the utility functions of the agents is proposed, these functions are still prefixed and they do not take into account changes in relevant contextual variables. In [14], the existence of a number of opponents as outside options is considered, but context remains the same for different negotiations, when the emergence of new opponents will clearly change that context. In [9], agents make offers and take decisions (accept/reject) based in a simple negotiation domain, summarized in reservation value and discount factor, but agents do not consider the dynamics of the context (or domain) to compute this variables. In [20], context obtains a clearer attention from the authors. Still, learning capabilities are used to model and select a forecast method to some contextual variables, but there is no mention to the strategy or policy used by an agent is going to act or concede during a particular negotiation given the context he is in. Moreover, we consider that the use of fixed learning parameters for each context selected after a previous analysis of those same contexts makes this approach hardly extendable. Finally, in the GENIUS negotiation tool [15] (General Environment for Negotiation with Intelligent multi-purpose Usage Simulation), actually one of the most used negotiation simulators and the one we also choose to run our own computational experiments, the negotiation deadline is even of public (common) knowledge, when that is certainly a decision that agents should be able to make on their own, based on their strategies and the information available to them.

Based on the considerations above, the main hypothesis in this work is that negotiation agents that learn to use in their benefit relevant contextual variables to select and evolve their strategies will reap more benefits than those agents that concentrate only on learning their opponents strategies independently of the context. Accordingly, we aim to create a negotiation setting that includes both environmental variables and context-aware agents. We design a novel context-driven negotiation environment and insert therein a learning agent that takes the context into account. This agent will use the available information alongside with reinforcement learning [25, 27] and Self-Play to generate specific knowledge about the context and select the proper actions as negotiations proceed. We will then exploit this knowledge to interact with other negotiation agents defined in the existing literature, agents that do not take into account contextual variables and yet have won the ANAC (Automated Negotiating Agents Competition) in the last years. We use the GENIUS tool [15] to run the simulated negotiation episodes and obtain significant results.

This paper is organized as follows. Firstly, a conceptual representation of the negotiation environment for context-aware negotiation agents is discussed. Next, we present "Strawberry", our own context-aware negotiation agent. We define its internal design and its Self-Play learning strategy alongside the "Oracle", a conceptual entity that is going to answer the information queries made to gather contextual relevant information by our agent. Later on, negotiation experiments are designed and run to generate results that can test our main hypothesis.

2 Negotiation environment

In this section, the main structure and components that constitute the negotiation environment are presented. As in most related works, a group of agents that agree to negotiate over certain issues are considered. In this work, the focus is on bilateral negotiations between two agents negotiating over one single issue with discrete values using a discrete time line. The alternating offers protocol which is, according to [10], the most influential protocol of all, is used throughout.

A colloquial definition of the negotiation context can be found in [20]. According to Rodriguez-Fernandez et al., the negotiation context refers to characteristics or circumstances under which the negotiation process occurs. We extend this definition formally, though. The negotiation environment, from the perspective of a single agent, is divided up in two abstract spaces: the agent's private information and the context (see Fig. 1). The agent's private information is composed by all his internal or private variables, those that other agents can not see but could attempt to model observing the actions the agent performs. The context, on the other hand, is composed by all the external or public variables, those that every agent would consider if relevant. In addition to this abstract spaces, the negotiation environment is populated by agents which resort to different models and strategies.

So, for a given agent A, his private information is defined as follows:

$$PI^A = \{X_1^A,\ X_2^A,\ ...,\ X_i^A,\ ...,\ X_l^A\} \tag{1}$$

where X_i^A in equation 1 is agent A's ith private variable. Except A, no other agent could directly see or query this variables (although anyone could try to estimate them by modelling A's behaviour).

Private variables could refer to the needs or urgency of a factory to buy supplies due to low inventory and high demand, the necessity or rush of a family to buy a bigger car or house, or the urgency of a brand to gain a new market through franchises in order to avoid bankruptcy.

Next, we define the negotiation context that agent A considers relevant as:

$$C^A = \{Y_1,\ Y_2,\ ...,\ Y_k,\ ...\} \tag{2}$$

where Y_k in equation 2 is the kth contextual variable that agent A decides to query. It is noteworthy that the values of these variables are public, so any agent could use them to make predictions upon them: every agent could "see" and query their values, if she has the means to do so and considers them relevant. In this way, C^A is a subset of the whole context C biasing agent A behavior.

These contextual variables may not only refer to changes in global currencies as the dollar or variations in the weather forecast when selling renewable solar energy, but also to particular events as the fall of a government leader, the sinking of a freighter that carry supplies to our factory, or even the risk of losing the possibility to negotiate with another parties due to resentment.

Finally, the group of negotiation agents that agent A considers is defined as:

$$NA^A = \{A,\ B,\ ...,\ J,\ ...,\ M\} \tag{3}$$

where J in equation 3 is the jth negotiation agent. This group set is made up of all the agents A knows or considers relevant to negotiate with. Now again, NA^A is a subset of the whole group of negotiation agents NA.

Summing up, the negotiation environment for a given agent A will be defined by:

$$E^A = PI^A \cup C^A \cup NA^A \tag{4}$$

From the point of view of an external, omniscient observer, the negotiation environment will be constituted by all the agents, contextual and private variables in existence. Formally:

$$E = PI \cup C \cup NA \tag{5}$$

where $PI = \{PI^A,\ PI^B,\ ...,\ PI^J,\ ...,\ PI^M\}$, that is, the conjunction of all agents' private information.

In Fig. 1 is presented a pictorial representation of the proposed negotiation environment.

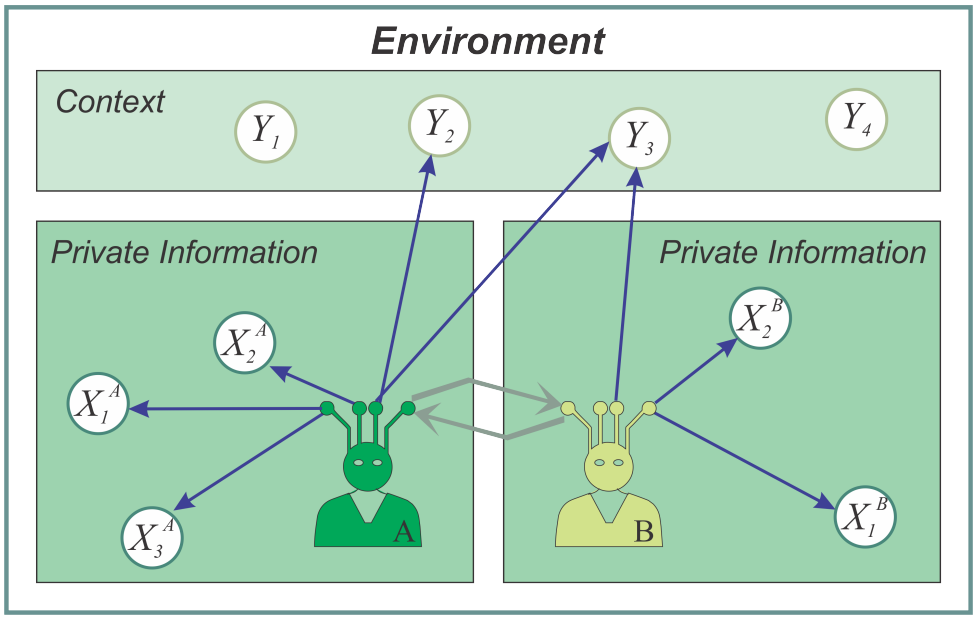


Figure 1: The proposed negotiation environment.

3 Strawberry: a context-aware negotiation agent

In this section, the proposed design for our context-aware agent Strawberry is presented. A component-based architecture proposed by Baarslag in [4], which receives the name of BOA (after Bidding strategy, Opponent model, and Acceptance strategy), is appropriated for implementing Strawberry. However, in order to test our main hypothesis, we incorporate to this architecture the possibility of querying the environment.

All these components and the resulting design are rather simple but will serve our purpose. As can be imagined, the design complexity could be further improved, but this environmental abstraction will suffice to prove our hypothesis about the role of contextual variables in the negotiation game. We profoundly believe that by keeping things simple (as long as it is possible) helps to maintain a clearer view, as well as highlights the essential aspects of our work.

On top of this architecture, two widely known techniques will be used, namely Self-Play [26] and Reinforcement Learning (or RL) [25, 27]. Then, Strawberry will learn to negotiate by simulating negotiation episodes in different contexts against another instance of himself while using the well-known Q-Learning algorithm.

In Fig. 2[1], we present a graphical representation of our agent Strawberry and the different components that will be explained in subsections below.

3.1 Environmental model

We will begin with the description of our environmental model. As we have shown in Section 2, we believe the environment can be modeled by a group of variables and negotiation agents. All we need is to provide our agent Strawberry (A) the means (abstract sensors) to query context-relevant information as deemed necessary, and combine it with his assumptions to model the negotiation environment including other agents.

To this end, we introduce the Oracle, a conceptual entity that could provide our agent real-time data values from the contextual and private variables, and summarize them in two state variables, *necessity* (ν^A) and *risk* (ρ^A). This two variables, as can be seen in Fig. 2, summarize the state of the environment where our agent is inserted.

[1] Adapted from [6].

Necessity and risk, altogether, stand for the positive and negative effects of closing a particular deal over our agent, given the state in which the environment is. While ν explains the necessity our agent has, according to his private variables, to close the deal, ρ accounts for the contextual variables, representing the risk our agent exposes himself if he does close that deal.

Next, the definition of those variables is given:

$$\nu^A = max\{X_1^A,\ X_2^A,\ ...,\ X_l^A\} \quad ; \quad 0 \leq \nu^A \leq 1 \tag{6}$$

$$\rho^A = max\{Y_1,\ Y_2,\ ...,\ Y_k\} \quad ; \quad 0 \leq \rho^A \leq 1 \tag{7}$$

These variables, ν^A and ρ^A, represent the state for associative learning used by Strawberry.

There are two important aspects to mention here. Firstly, although more variables could have been defined to represent the state of the environment, the use of agent necessity and perceived risk of its decision is a simple representation of each perspective that help us prove our main hypothesis, namely the importance of contextual information in the negotiation game. Secondly, as can be seen, a conservative approach is used in the computation of those state variables: necessity and risk are defined by the greatest necessity and the greatest risk the agent is exposed to. Again, such an approach is only a first approximation to prove our hypothesis about the importance of context-aware negotiation.

Necessity and risk, in this work, have two main attributes. In the first place, these are private variables: opponents can not see if an agent is in great necessity or at a low risk during a negotiation (although they can estimate it given the agent's behaviour, as previously said). In the second place, these variables are fixed during a negotiation episode. This aspect may seem a limitation, but it is not. In automated negotiations, offers are rapidly interchanged between agents, and the whole negotiation episode does not last more than a couple of seconds. In this way, the assumption of a fixed environment during a particular negotiation is not an issue to worry about. Despite this, if the negotiation episode takes place during an extended period of time, it would be desirable that a negotiation agent could perceive relevant changes in the environment to adjust his behavior in the remaining part of the episode.

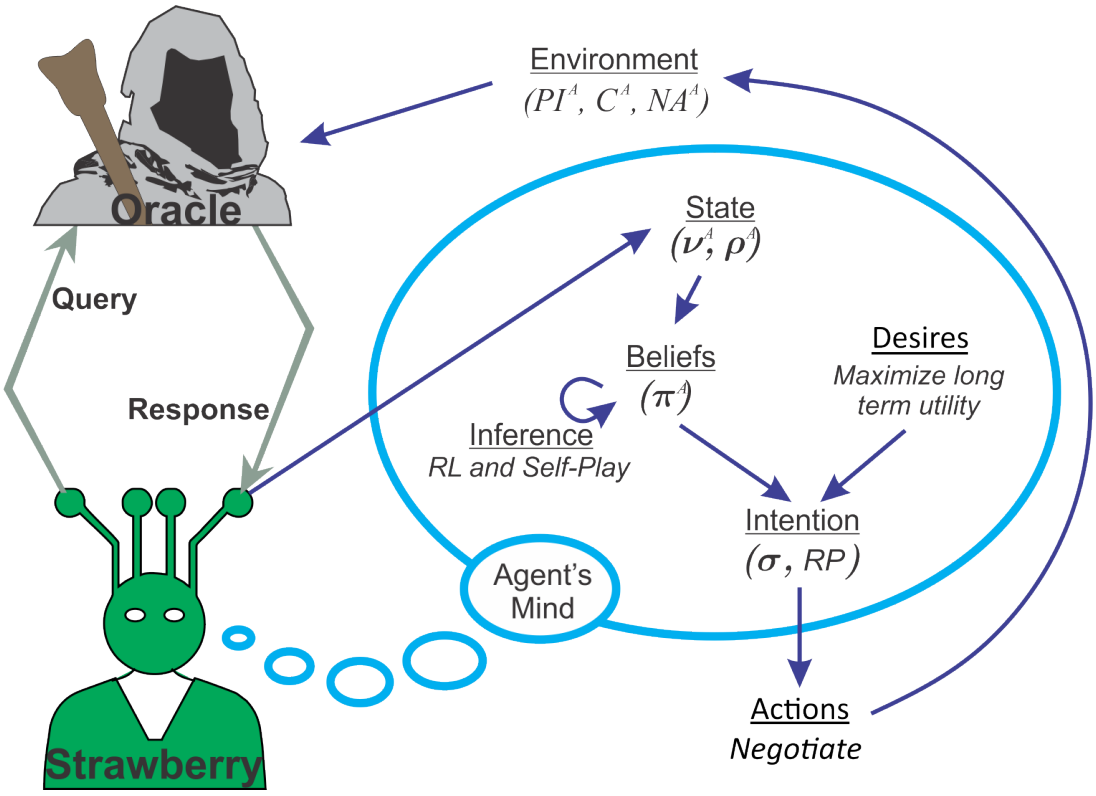


Figure 2: The internal design of Strawberry, our context-aware negotiation agent.

3.2 Bidding strategy

Taken from Fatima [10] and references therein, a heuristic concession strategy for Strawberry is defined as follows:

$$O_t = IP + (RP - IP) * \sigma(\beta, t, n) \tag{8}$$

where O_t is the offer Strawberry will make at time t, IP is the initial price, which is assumed to be the best deal the agent considers it can get from the negotiation, and RP is the reserve price, which is the worst deal the agent can achieve at the end of a negotiation episode. The concession strategy σ is based on:

$$\sigma = \left(\frac{t}{n}\right)^{1/\beta} \tag{9}$$

where t is the time elapsed from the beginning of the negotiation, n is the deadline, and β defines the concession rate.

3.3 Acceptance strategy

The acceptance strategy to be used is AC, taken from Baarslag [4], where its effectiveness was demonstrated. It could be summarized as follows: our agent will accept an offer from his opponent B if and only if this offer supposes a higher utility for our agent than the utility he would obtain from his own next offer. In other words, Strawberry will accept the offer if:

$$u(O_t^B) \geq u(O_{t+1}^A) \tag{10}$$

where $u(O_t^{agent})$ is the long-term utility Strawberry will obtain from the offer O made by the *agent* at time t.

3.4 Self-Play and Reinforcement Learning

Strawberry will use his self-play capacity to acquire some information of the public context by modelling its influence in negotiation game. To this end, he will play with another instance of himself, adapting his policy π^A to select the hyperparameters that define how to properly behave in a negotiation episode.

This does not invalidate the importance of learning from other opponents: the importance of modelling the opponent was clearly stated in previous works [5, 31]. Still, the self-play learning phase could help our agent gain an initial understanding of the context he is in and how does this context affects him.

Strawberry's final desire is not only to maximize his next possible reward r but also to maximize his long term utility, as stated in Fig. 2. This learning strategy is implemented by the Q-Learning algorithm, which consists of a function that iterates over the expected cumulative rewards for future time steps in a negotiation episode (how many will depend on the tuning of the algorithm hyper-parameters) given the perceived state s_t^A of the environment providing that the agent takes a certain action a from the set of possible actions. The action to take is determined by a policy π^A derived from the Q-values, which are the way this algorithm represents the immediate and long-term utility for every state-action pair. At the end of each negotiation episode, Strawberry observes the immediate reward r and the next state s_{t+1}^A that the environment returns, and obtains the Q-value from the best action the agent can take in that situation (indicated by $max_a\ Q(s_{t+1}^A, a)$ in equation 11). Then, the Q-Learning algorithm actualizes the Q-value according to:

$$Q(s_t^A, a_t) = Q(s_t^A, a_t) + \alpha\ \left[r_{t+1} + \gamma\ max_a\ Q(s_{t+1}^A, a) - Q(s_t^A, a_t)\right] \tag{11}$$

We choose each state for Strawberry (A) to be defined by the tuple:

$$s^A = (\nu^A,\ \rho^A) \tag{12}$$

where ν^A and ρ^A are the necessity and risk associated with the perceived state of its environment, as mentioned earlier.

The possible actions for Strawberry are defined by:

$$a = (RP,\ \beta) \tag{13}$$

where RP and β are the variables our agent will choose to vary his negotiation strategy.

A group of hyper-parameters need to be set in this algorithm to work. These are ϵ, α and γ. ϵ defines the exploratory nature of our agent, that is, the probability our agent takes an exploratory move (normally, a random move) rather than exploiting its knowledge (following the current estimation of an optimal policy). α is the learning rate parameter, which gives more weight to recent rewards than to past rewards. γ is the discount rate, with which the agent will try to maximize the sum of the discounted rewards it is going to receive in the future. We could say that by means of these three parameters we have past, present and future into account through the chosen values α, ϵ, and γ, respectively.

3.5 A formal definition of Strawberry's behavior

Here, the different internal conditions Strawberry can be in are discussed. It becomes necessary to explain that these conditions are different to the perceived environmental states that have been defined earlier, since its internal conditions in Fig. 3 correspond to states of a state machine diagram in a given negotiation episode.

In Fig. 3, the two basic activities for the Strawberry agent are shown: it may either learning through Self-Play or actually negotiating against other opponents. Of course, the fact that our agent does not learn while negotiating with other opponents is imposed to avoid exploration when there exist enough knowledge to exploit from. This favor evaluating how good the policy learned through Self-Play actually is.

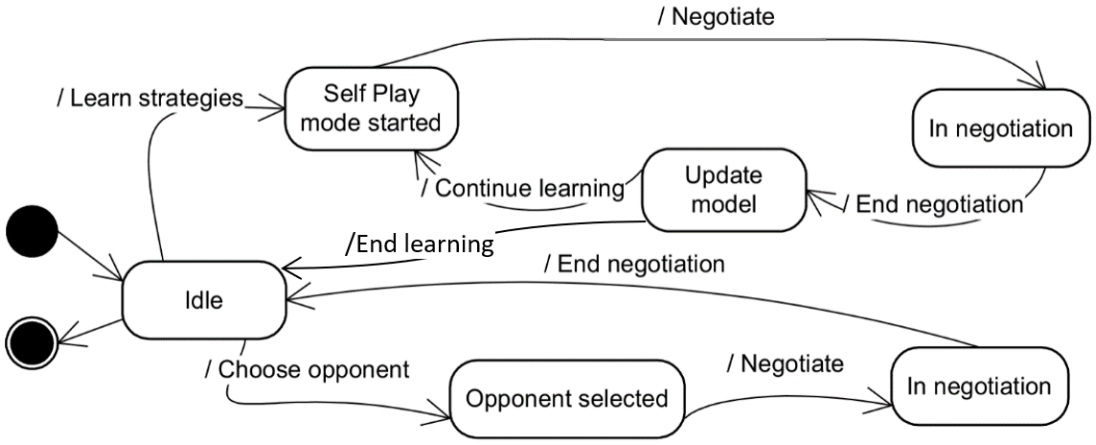


Figure 3: Strawberry's general state machine diagram.

In Fig. 4, an entire negotiation episode is shown. When the negotiation episode starts, Strawberry requests the Oracle some information from the context. In the next step, he selects a negotiation strategy based on the policy he has already learned or is actually learning. To continue with, the negotiation itself is represented: first, the agent makes an offer; then, he waits and evaluates the counteroffer; then he chooses whether to accept or to reject (cases in which the negotiation episode ends), or to make a counteroffer. If the opponent makes the first offer, then the agent must evaluate it and take the proper decision. If the opponent accepts or rejects an offer of the Strawberry agent, then the negotiation episode ends.

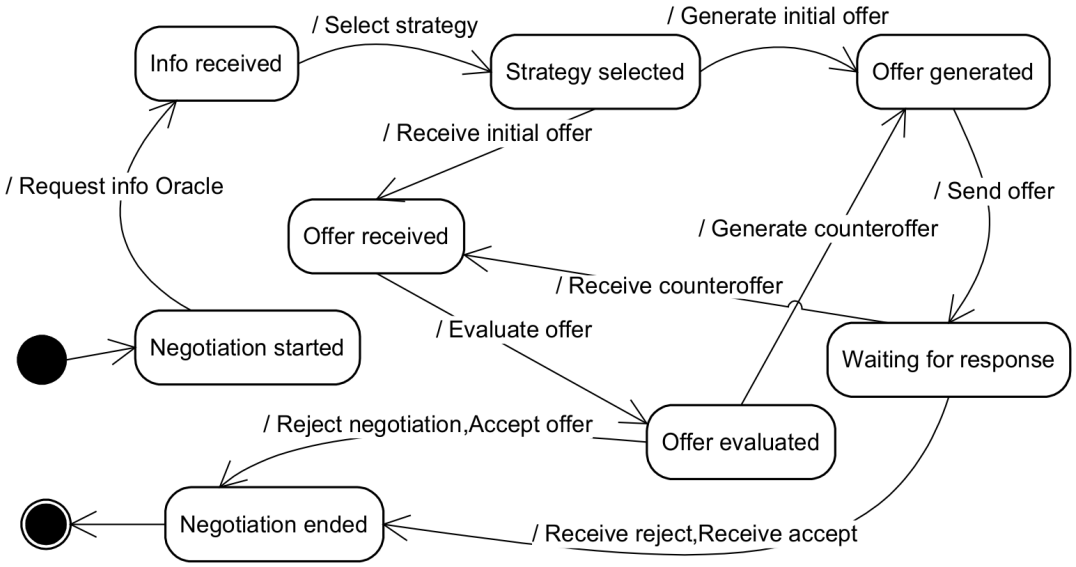


Figure 4: The negotiation process of Strawberry in a state machine diagram.

4 GENIUS

As has been said before, the GENIUS simulation software will be used to test our main hypothesis regarding the key role of contextual variables during a negotiation process. GENIUS is a specialized non-commercial environment for simulating negotiations, where a given agent design can be implemented and then faced against a set of previously available agents. Single negotiations or tournaments could be run, based on deadlines that are common, public knowledge. The GENIUS present the results in a table and a chart, where the Pareto frontier, the Nash equilibrium, and other social welfare metrics could be seen.

The GENIUS simulator is a practical tool to try new agents in the field of automated negotiations. Nevertheless, there is no context in this negotiations and, from our point of view, this makes the negotiation environment too unrealistic. As we would like to prove that agents should take the context into account so as to make more rational decisions, we implemented the negotiation environment represented in Section 2 and included it in GENIUS. We will use these new concepts in the next sections to define and develop the computational experiments.

5 Experiments

In this section, the experimental setting is described, including how negotiation simulations were run, and the results obtained after negotiation episodes.

Firstly, some changes and additions to GENIUS are discussed. New features were developed in Java, using the IntelliJ IDEA environment, a package that gave us the possibility to adapt GENIUS to our needs. Finally, we developed our agent, Strawberry, with the capability to query contextual variables to the Oracle mentioned in section 3.

5.1 Experimental setting

The first step to address our hypothesis was to create Strawberry's private and contextual variables. Without any loss of generality, only two variables in each space are considered (X_1, X_2, Y_1, and Y_2), each one taking random integer values between 0 and 3. These variables will be queried through the Oracle, normalized to 1 when required by Strawberry, and internally modelled using ν^A and ρ^A.

The second step was to select a domain in GENIUS that could gave us simple but revealing results, bearing in mind that the focus here are single-issue negotiations. The domain selected was the "pie domain", a problem usually addressed in game theory [18] and also available in GENIUS. In this domain, two agents negotiate over a pie that is divided into a number of pieces (we choose this number to be a thousand). The aim for each agent is to get as many pieces as it can, taking into account that, if the deadline is reached without a deal, every one gets zero pieces. The utility is given by how many pieces an agent gets by the end of the negotiation episode divided by 1000.

The third step was to define the concrete aspects of the Q-Learning algorithm. We designed a reward function upon which the environment would give Strawberry a reward r at the end of each negotiation episode, depending on the outcome of the negotiation (either an agreement is reached or not) and the environment state s^A at which the negotiation episode takes place. This reward function is defined as follows:

- If the negotiation ended successfully (an agreement is reached):

$$r = u(O_{t=end}) \tag{14}$$

  that is, the utility that the agent obtains from the last offer reported.

- If the negotiation ended unsuccessfully:

$$r = \begin{cases} -1 & \text{if } X_1 = 3 \\ -1/3 & \text{if } X_2 = 3 \\ -2/3 & \text{if } Y_1 = 3 \ \wedge \ Y_2 = 0 \\ 1/3 & \text{if } X_1 \leq 1 \\ 2/3 & \text{if } X_1 \leq 1 \ \wedge \ X_2 \leq 1 \end{cases} \tag{15}$$

The rationale behind this function is that the agent would be not only concerned by the result of the negotiation, but also by the perceived state of his private information, the context, and how these affect him.

A number of hyper-parameters had to be set in order to make Strawberry capable of learning from reinforcements. As a common rule of thumb, α and ϵ are usually set to 0.1 [25], and γ to 0.9, values that contribute to a fast learning by means of a reasonable exploration-exploitation trade-off. These are typical default values used in the reinforcement learning literature.

Given that our agent learns using tabular Q-Learning, states need to be defined in discrete values. For this purpose, we divided the possible values for ν^A and ρ^A, into three intervals $[0; 0,33]$, $(0,33; 0,66]$, and $(0,66; 1]$, as both vary between 0 and 1. In this sense, for our agent, a necessity of 0.1 and a necessity of 0.29 would be taken in the same group, semantically a "low necessity" internal state. Likewise, a perceived risk of 0.8 and other of 1 would be taken in the same group, semantically a "high risk" internal state.

We also define the actions allowed to Strawberry. We set three different values for β: 0.5, 1.0, and 2.0, and five possible values for RP: 0.0, 0.2, 0.4, 0.6, and 0.8, which provide Strawberry with 3 * 5 = 15 different concession strategies. Summing up, we will have a maximum of 3 * 3 * 3 * 5 = 135 Q-values to learn.

5.2 Experimental design

The experiments were conducted in the pie domain, where a deadline of 180 rounds was used for simulating the negotiation episodes. The experiments were divide up in three phases: the learning phase, the negotiation phase, and the Self-Play improvement phase.

In the learning phase, the Strawberry agent is set to negotiate against himself using the tournament setting that GENIUS provides. Self-Play simulations were run for 10, 100, 500, 1000, 2000, 5000, and 10000 negotiation episodes in order to see how much learning may affect the subsequent negotiation phase.

In the negotiation phase, tournaments are played against some of the existent negotiation agents in GENIUS. We have chosen some simple ones and others really difficult to beat, winners of previous ANAC competitions. The chosen opponent types are:

- Random Party (RP)
- BoulwareNegotiationParty (B)
- ConcederNegotiationParty (C)
- CUHKAgent2015 (CUHK)
- AgentFSEGA (FSEGA)
- Agent_K (K)
- IAMcrazyHaggler (Haggler)
- AgentSmith (Smith)
- Gahboninho (G)
- BRAMAgent (BRAMA)

A short description of each opponent will be given. Random Party is a naive agent that makes always a random offer, with the only restriction that he always concedes more than in his previous turn. The BoulwareNegotiationParty and ConcederNegotiationParty use time-dependent negotiation strategies. The Boulware concedes less at the beginning and more at the end of the negotiation if no agreement has been reached yet. The Conceder works the other way around: concedes more at the beginning and less at the end, if no agreement was reached yet. The CUHKAgent2015 estimates the concession degree of the opponent in order to make concessions. AgentFSEGA predicts the opponent's negotiation strategy and then concedes accordingly. Agent_K uses the mean and variance of the utility of all received offers, and then tries to determine the best offer he might get. He then accepts or rejects based on the probability of receiving a better offer. IAMcrazyHaggler makes random proposals among those having a high utility to him. AgentSmith constructs an opponent model and then concedes slowly towards the opponent's offers. Gahboninho uses a meta-learning strategy, trying at first to identify if the opponent is modelling him, and then exploiting this behaviour. Finally, the BRAMAgent uses opponent modeling to propose offers likely to be accepted by the opponent.

Simulations were run in two different settings: Strawberry against all other agents together, and Strawberry against each one of them, separately. Each tournament consisted of 100 negotiations were Strawberry used the knowledge previously gained through Self-Play, but did not learn whilst negotiating with the other opponents.

Finally, the Strawberry agent is put to negotiate against himself, but this time without doing any learning, in order to see if he achieves better agreements in Self-Play compared to the different learning episodes he has previously done.

5.3 Results and analysis

After the learning phase, the negotiation phase has given us some interesting results that are depicted in Fig. 5. At first glimpse, the cumulative utility of Strawberry against the average opponent starts below 300 and reaches 400 as he learns to account for the environment. This behavior was rather expected since the reinforcement learning method resorts to the association of the goodness of an action to the value of contextual variables. On the other hand, it is also possible to see that taking the environment into account could change the perspective of negotiation itself: Strawberry had received, on average, better outcomes than his negotiation counterparts when considering the context of the negotiation. This reasoning tempts us to say that the environment should be taken into account so as to gain a competitive advantage.

Another important observation to make from Fig. 5: if some agent considers environmental variables, there will be an increase in the rest of the agents cumulative utilities, not only in his own. This astonishing result gives rise to two different theories. The first one is that it could be possible that if one of the agents takes some variables into account that the other does not, better agreements are reached, with a tendency to improve social welfare. The second theory, the one we think could explain better this phenomena, is that, as Strawberry learns, he makes more rational decisions and does not take so many actions at random. In this context, the other agents could build a better model out of him and predict better what his moves are going to be. In other words, they model the part of the environment they do not see through the model they made of Strawberry's negotiation behavior. We think these are two key aspects we have discovered through our research.

In Fig. 6, we can see the utilities obtained by Strawberry against each particular agent, and the utilities obtained by his opponents, which do not take environmental variables into account. Again, as expected, the utilities obtained by Strawberry are always better when the rewards of the environment

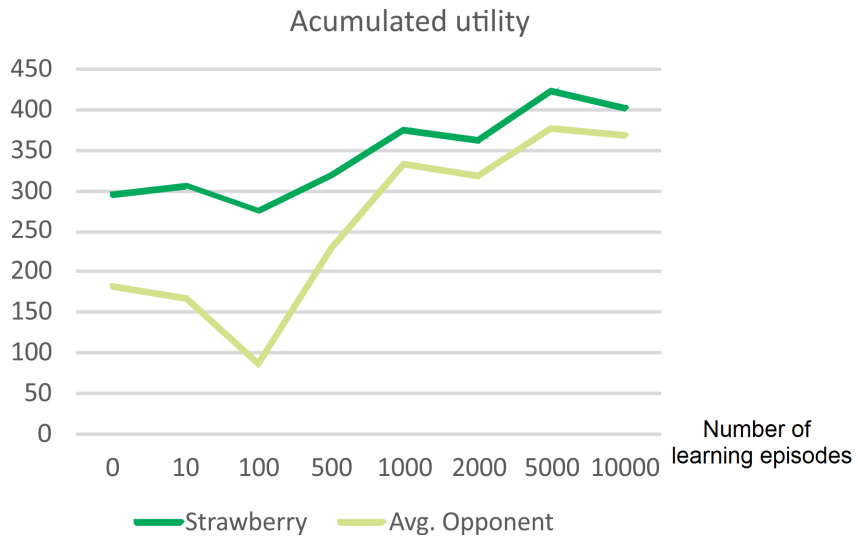


Figure 5: Strawberry's accumulated utility over 100 tournament scenario against the average opponent, that is to say, the average utility the different opponents mentioned in Section 5.2 have obtained. The horizontal axis shows how many learning negotiations with Self Play has Strawberry made. The vertical axis shows the accumulated utility throughout the 100 tournament negotiation session.

are considered than when they are not (the rewards that GENIUS gives). Another thing we could see is that as agents get more complex (e.g., with opponent models and flexible or though strategies, etc.), they make it more difficult for Strawberry to get a good deal. Particularly, the CUHK agent seems to behave really tough, not making too much room for Strawberry to have a competitive edge, but still getting a great deal of accumulated utility. It is worth asking how much an agent like this would gain if he were taking the context into account, as Strawberry does.

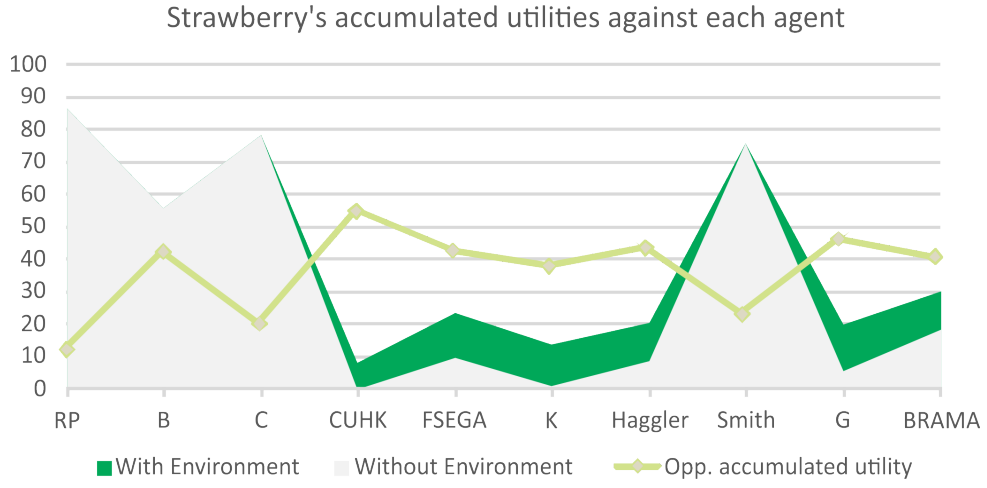


Figure 6: Accumulated utility obtained by Strawberry after 10000 learning negotiation sessions in Self Play against each particular agent, when it is considered the reward from the environment and when it is not, and the utilities obtained by his opponents.

In the Self-Play improvement phase, we have reached other conclusions. As can be seen in Fig. 7, Strawberry achieves better and better agreements as he learns considering the context of the negotiation, thus increasingly maximizing the social welfare over negotiations. Also, in the graph on the right side,

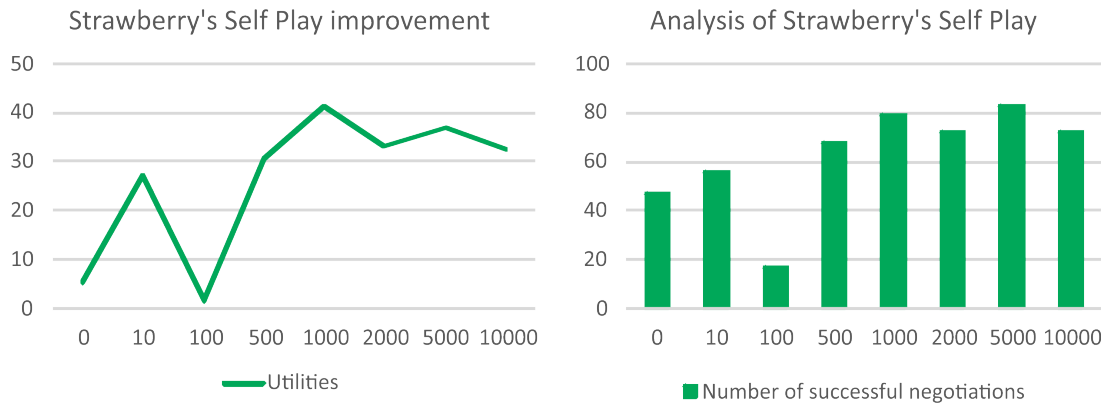


Figure 7: The graphic on the right shows Strawberry's Self Play improvement in accumulated utility as he learns. In the left graphic, we see how many successful negotiations reaches Strawberry's when making 100 negotiation sessions against himself.

we see how Strawberry gets more successful negotiations as he learns from evaluative feedback from negotiation episodes. Fig. 6 and 7 vividly highlight the importance of contextual-learning in negotiations, as can be expected, following the previous results shown in Fig. 5.

A final comment can be made about reaching an equilibrium. We have shown that Strawberry gets better through Self-Play and learning. However, he does not tend to reach the Nash equilibrium that GENIUS proposes, as can be seen in Fig. 8, when playing against himself. In fact, there are no great changes in the mean distance to the Nash point. This has, from our point of view, a simple explanation: GENIUS does not consider the contextual variables to calculate the Nash equilibrium. As the learning advances, mainly after the 500 tournaments learning, there is a difference of approximately 0.3 units, which highlights that the Nash equilibrium is not actually where the GENIUS locates it. In conclusion, it could be stated that is not possible to find the real Nash equilibrium of a negotiation game unless the relevant contextual variables that affect all agents' utilities are taken into account.

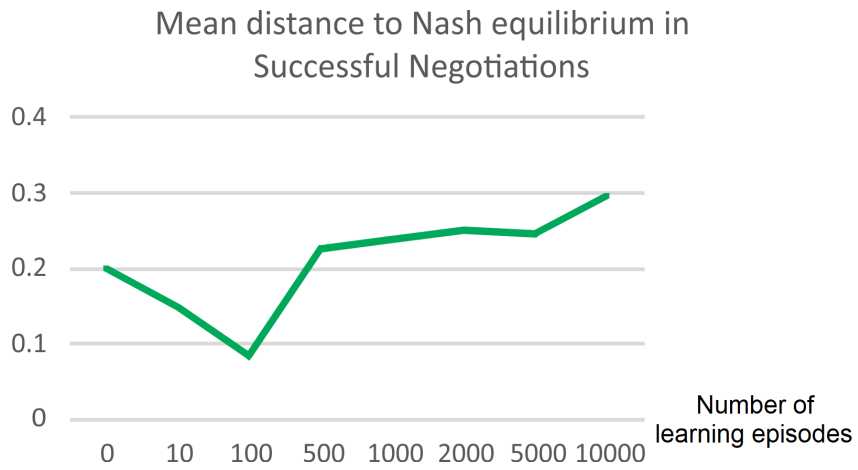


Figure 8: This graphics presents in the vertical axis the mean distance to the Nash equilibrium proposed by GENIUS in 100 Self-Play episodes once learning has ended. The horizontal axis shows the number of learning episodes previously made by Strawberry.

6 Concluding remarks

The importance of taking the context into account when two agents are negotiating over a certain issue has been addressed. A novel way of modeling the negotiation environment based on characterizing both the agent's private variables, which consists of the agent's strategies and preferences, and the context, where, besides other negotiation agents, a group of external variables that influence the utilities and values of concerned agents, are used to learn negotiation strategies.

The proposed Strawberry agent resorts to contextual variables to gain a competitive advantage. We have presented the way our agent account for contextual variables, based on which his bidding and acceptance strategies are built up. Then, we have explained how Strawberry would obtain some negotiation knowledge using the Q-Learning algorithm and Self-Play.

Computational experiments were designed to assess the validity of our central hypothesis: the advantage of including contextual variables in a negotiation agent. Results obtained confirm our earlier thoughts whereas other results are rather unexpected. Strawberry is quite competitive in a heterogeneous environment composed of a number of agents, even though he makes no explicit model of his opponents. We have proven that the utilities agents perceive are different whether we take or not the variables of the context and the agent's private variables into account. These results sustain our main hypothesis, but more learning experiments are needed. We have seen, along with Strawberry's improvement, his opponent improvements as he learns. We theorize that the models other agents make out of Strawberry help them discover implicitly the variables Strawberry takes into account, although they do not know of their existence. It can also be stated, from the results shown in Fig. 6, that the Strawberry agent reap higher utilities when he takes these variables into account. The importance of Self-Play for cheap Learning is highlighted through the results obtained. Hence, social welfare can be increased as agents learns collectively through inexpensive simulation-based on Self-Play learning.

As a final word, it can be said that our hypothesis seems correct from the point of view of the Nash equilibrium. If we take the contextual variables into account, it makes no sense to find an equilibrium between the strategies of the negotiation agents considered in isolation. If contextual variables are not perceived by any of the agents, then they would attempt to reach an equilibrium that is nonexistent. Our agent Strawberry, simple as it is, when playing against himself shows us that the equilibrium is somewhere else, not just "in the middle". In other words, should we assume that, when two people claim for a piece of pie, the whole is to be always divided exactly in two? We thing we should not, and the results support our earlier thoughts that this division does not only depend on the agents and the pie itself, but also on external variables agents should not leave aside.

7 Future work

There are a number of research avenues that can be taken to further this research. For example, we suspect these results could be extended to multiple-parties negotiations, different protocols, multiple issues, various domains, and so on. We have run our simulations in GENIUS, but other tools could also be used to obtain more comprehensive results and to finally test our central hypothesis. Strawberry could be also improved. It was found that it is not so good against top agents like CUHK or Ghaboninho. To this aim, the environmental model it builds can be more complex and new bidding and acceptance strategies could be added. Self-Play could be just a component of the learning process; learning to negotiate must be extended to many other agents in order to get better outcomes, modeling not just the external variables, but also the opponent model of the environment and the other opponents using Theory of Mind and evolving strategies using multi-agent Q-learning. More experiments should be made so as to ratify these new approaches. Different external variables can be provided by the Oracle: more complex variables which may give rise to unexpected behavior that will make it more difficult not only for Strawberry to model, but also to the rest of the interacting agents that may be disconcerted by Strawberry's actions. A new intrinsically-motivated reward function should be designed, that could give Strawberry a competitive edge to identify sooner what is going on in the environment by pinpointing key variables. Deadlines could vary between negotiation episodes. We see this as a great restriction imposed by GENIUS: that the deadline is indeed public knowledge. More sophisticated opponents could be addressed, and querying strategies to the Oracle could be also part of the learning process.

References

[1] Ajay Agrawal, Joshua Gans, and Avi Goldfarb. *Prediction machines: The simple economics of artificial intelligence*. Harvard Business Review Press, Boston, Massachusetts, 2018.

[2] Bedour Alrayes, Özgür Kafalı, and Kostas Stathis. Concurrent bilateral negotiation for open e-markets: The conan strategy. *Knowledge and Information Systems*, 2017.

[3] Dan Ariely. *Predictably irrational: The hidden forces that shape our decisions*. Harper Perennial, New York, revised and expanded ed. edition, 2010.

[4] Tim Baarslag. *Exploring the strategy space of negotiating agents: A framework for bidding, learning and accepting in automated negotiation*. Springer theses. Springer, Switzerland, 2016.

[5] Tim Baarslag, Mark J.C. Hendrikx, Koen V. Hindriks, and Catholijn M. Jonker. *Learning about the opponent in automated bilateral negotiation: a comprehensive survey of opponent modeling techniques*, volume 30. Springer US, 2016.

[6] Chris L. Baker, Julian Jara-Ettinger, Rebecca Saxe, and Joshua B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1(4):1–10, 2017.

[7] Natalia Criado Pacheco, Carlos Carrascosa, Nardine Osman, and Vicente Julián Inglada. *Multi-Agent Systems and Agreement Technologies*, volume 10207. Springer International Publishing, Cham, 2017.

[8] Daisuke Wakabayashi. Waymo's autonomous cars cut out human drivers in road tests, 2017.

[9] Eden S. Erez, Inon Zuckerman, and Dror Hermel. Automatic negotiation: Playing the domain instead of the opponent. *Journal of Experimental and Theoretical Artificial Intelligence*, 29(3):597–616, 2017.

[10] S. Fatima, S. Kraus, and M. Wooldridge. *Principles of Automated Negotiation*. Cambridge University Press, 2014.

[11] David Ferrucci, Anthony Levas, Sugato Bagchi, David Gondek, and Erik T. Mueller. Watson: Beyond jeopardy! *Artificial Intelligence*, 199-200:93–105, 2013.

[12] Roger Fisher and William Ury. *Sí, ¡de acuerdo! Como negociar sin ceder*. Libros universitarios y profesionales. Serie Norma de desarrollo gerencial. Norma, [Colombia], 1985.

[13] Paul W. Glimcher, editor. *Neuroeconomics: Decision making and the brain*. Academic Press, London and San Diego, CA, 1st ed. edition, 2009.

[14] Cuihong Li, Joseph Giampapa, and Katia Sycara. Bilateral negotiation decisions with uncertain dynamic outside options. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 36(1):31–44, 2006.

[15] Raz Lin, Sarit Kraus, Tim Baarslag, Dmytro Tykhonov, Koen Hindriks, and Catholijn M. Jonker. Genius: An integrated environment for supporting the design of generic automated negotiators. *Computational Intelligence*, 30(1):48–70, 2014.

[16] Ariel Monteserin and Analía Amandi. Argumentation–based negotiation planning for autonomous agents. *Decision Support Systems*, 51(3):532–548, 2011.

[17] Philip Ball. How life (and death) spring from disorder. *Quanta Magazine*, 2017.

[18] Ariel D. Procaccia. Cake cutting: Not just child ' s play. *Communications of the ACM*, 56(7):78–87, 2013.

[19] Fenghui Ren and Minjie Zhang. A single issue negotiation model for agents bargaining in dynamic electronic markets. *Decision Support Systems*, 60(1):55–67, 2014.

[20] J. Rodriguez-Fernandez, T. Pinto, F. Silva, I. Praça, Z. Vale, and J. M. Corchado. Context aware q-learning-based model for decision support in the negotiation of energy contracts. *International Journal of Electrical Power and Energy Systems*, 104(October 2017):489–501, 2019.

[21] Stuart Russell. Artificial intelligence: The future is superintelligent. *Nature*, 548(7669):520–521, 2017.

[22] Arvind Sathi. *Cognitive (internet of) things: Collaboration to optimize action.* Nature America Inc, New York NY, 1st edition, 2016.

[23] Seventh International Conference on Learning Representations. *Emergent Communication through Negotiation*, 2018.

[24] Shane Legg. *Machine Super Intelligence: PhD Thesis.* Shane Legg, 2008.

[25] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction.* Adaptive computation and machine learning. MIT Press, Cambridge Mass., 2nd edition, 2018.

[26] Gerald Tesauro. Temporal difference learning and td-gammon. *Communications of the ACM*, 38(3):58–68, 1995.

[27] Christopher John Cornish Hellaby Watkins. *Learning from Delayed Rewards.* PhD thesis, King's College, Cambridge, UK, 1989.

[28] Hermanes Albertus de Weerd and Rineke Verbrugge. *If you know what I mean: Agent-based models for understanding the function of higher-order theory of mind.* University of Groningen and Rijksuniversiteit Groningen, Groningen, 2015.

[29] A. D. Wissner-Gross and C. E. Freer. Causal entropic forces. *Physical Review Letters*, 110(16):1–5, 2013.

[30] G. Yang, Y. Chen, and J. P. Huang. The highly intelligent virtual agents for modeling financial markets. *Physica A: Statistical Mechanics and its Applications*, 443:98–108, 2016.

[31] Farhad Zafari and Faria Nassiri-Mofakham. Popponent: Highly accurate, individually and socially efficient opponent preference model in bilateral multi issue negotiations. *IJCAI International Joint Conference on Artificial Intelligence*, 2016(April):5100–5104, 2017.

[32] Daniel Dajun Zeng and Katia Sycara. Benefits of learning in negotiation. In *Proceedings of the fourteenth National Conference on Artificial Intelligence and ninth Innovative Applications of Artificial Intelligence Conference*, volume July 27-31 of *AAAI '97/IAAI '97*, Menlo Park and London, 1997. The AAAI Press, Menlo Park, California.

[33] Yi Zou, Wenjie Zhan, and Yuan Shao. Evolution with reinforcement learning in negotiation. *PLoS ONE*, 9(7), 2014.