A Fine-Tuning Aggregation Convolutional Neural Network Surrogate Model of Strategy Selecting Mechanism for Repeated-Encounter Bilateral Automated Negotiation

Shengbo Chang[a] and Katsuhide Fujita[b]
Department of Electrical Engineering and Computer Science, Faculty of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo, Japan

Keywords: Automated Negotiation, Strategy Selecting Mechanism, Surrogate Model, Fine-Tuning.

Abstract: Negotiation with the same opponent for multiple times for each in a different domain commonly occurs in real life. We consider this automated negotiation problem as repeated-encounter bilateral automated negotiation (RBAN), in which it is essential to learn experiences from the history of coping with the opponent. This study presents a surrogate-model-based strategy selecting mechanism that learns experiences in RBAN by fine-tuning the proposed aggregation convolutional neural network (CNN) surrogate model (ACSM). ACSM is promised to assess strategies more precisely by applying CNN to extract features from a matrix showing the outcomes' utility distribution. It ensures the abundance of extracted features by aggregating multiple CNNs trained with diverse opponents. The fine-tuning approach adapts ACSM to the opponent in RBAN by feeding the present negotiation results to ACSM. We evaluate ACSM and the fine-tuning approach experimentally by selecting a strategy for a time-dependent agent. The experiments of negotiating with four Automated Negotiating Agents Competition (ANAC) champions and six basic agents are performed. ACSM is tested on 600 negotiation scenarios originating from ANAC domains. The fine-tuning approach is tested on 60 RBNA sessions. The experimental results indicate that ACSM outperforms an existing feature-based surrogate model, and the fine-tuning approach is able to adapt ACSM to the opponent in RBAN.

1 INTRODUCTION

Negotiations with the same opponent multiple times in a new domain each time happen in real life. For example, a retailer may need to negotiate with one diverse product supplier about the price, diversities, and amount in each season as the preference of customers change (Chkoniya and Mateus, 2019). Moreover, the identity of the opponent negotiating with is known in the setting of the Automated Negotiation League in the Automated Negotiating Agents Competition (ANAC) 2022 (Aydogan et al., 2022), indicating that candidates could change their strategies by learning from their past experience with the given opponent. In this study, we consider the strategy selection of the repeated-encounter bilateral automated negotiation (RBAN, i.e., a sequence of bilateral automated negotiation with the same opponent multiple times and each time in a different scenario) (Renting et al., 2022).

There is no single strategy that could dominate all possible settings (Ilany and Gal, 2016). Previous studies (Baarslag et al., 2012; Baarslag et al., 2013; Ya'akov Gal and Ilany, 2015) demonstrated that the best negotiation strategy varies with the negotiation scenario, even for the same opponent. Therefore, selecting the optimal strategy for each scenario is essential in RBAN. The similar behavior pattern of the opponent in different scenarios is a distinguishable feature of RBAN that asks negotiators to use their experience and learn from the negotiation history to cope with the opponent. Though several studies focused on strategy selection (Ilany and Gal, 2016; Kawata and Fujita, 2020; Wu et al., 2021; Baarslag et al., 2013; Sengupta et al., 2021; Renting et al., 2020; Güneş et al., 2017; Fujita, 2014; Fujita, 2018), few considered the problem of RBAN to the best of our knowledge.

Surrogate models, generally used in algorithm selection, predict the outputs for unknown algorithm parameter inputs by regressing the known inputs with outputs. A surrogate model for strategy selection in automated negotiation usually uses the negotiation scenario features and a strategy configuration as input,

[a] https://orcid.org/0000-0001-6992-7566
[b] https://orcid.org/0000-0001-7867-4281

Chang, S. and Fujita, K.
A Fine-Tuning Aggregation Convolutional Neural Network Surrogate Model of Strategy Selecting Mechanism for Repeated-Encounter Bilateral Automated Negotiation.
DOI: 10.5220/0011701300003393
In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence (ICAART 2023) - Volume 2*, pages 277-288
ISBN: 978-989-758-623-1; ISSN: 2184-433X

and its output is the predicted evaluation value (Ilany and Gal, 2016; Renting et al., 2020). For the given negotiation scenario, existing surrogate models use expert scenario features to predict the performance of a strategy configuration (Ilany and Gal, 2016; Renting et al., 2020; Renting et al., 2022). Since these features relied on human intuition, prediction accuracy is usually lost to some extent. The 2-dimension outcome space (our agent utility - the opponent utility) of a negotiation scenario could represent the scenario exhaustively. Additionally, convolution neural networks (CNN) can be trained to extract useful features from the 2D outcome space automatically. Therefore, training the CNN to extract scenario features is a feasible way of overcoming the disadvantages of human intuition.

This study extends the existing strategy selecting mechanisms with a CNN-based surrogate model and an online learning method for RBAN. The contributions of this study are as follows:

- An approach of extracting scenario features with CNN on a discrete size-fixed outcome distribution map which indicates the number of outcomes falling within bins of a predefined utility range.
- A surrogate model (ACSM) aggregates multiple CNN-components that implement the feature-extracting approach trained with diverse opponents, ensuring the robustness and feature abundance when against an unknown opponent. A fine-tuning approach adapts the proposed surrogate model to the facing opponent in RBAN efficiently.
- The feature-extracting approach is validated experimentally by comparing ACSM with an expert-feature-based neural network surrogate model (NNSM) in various scenarios of single negotiation. The fine-tuning approach is validated experimentally by comparing ACSM with fine-tuning with one without fine-tuning in RBAN.

The remainder of this paper is structured as follows: Section 2 presents related works; Section 3 presents RBAN; Section 4 introduces ACSM with the proposed feature-extracting method; Section 5 introduces the strategy selecting mechanism of fine-tuning ACSM for RBAN; Sections 5 demonstrates the experimental results; Section 6 summarizes this paper and discusses future possibilities.

2 RELATED WORK

This work focuses on strategy selection for each negotiation scenario in RBAN. The area related mostly is strategy selection in automated negotiation.

(Ilany and Gal, 2016) proposed a Meta-agent that includes a strategy portfolio used in the ANAC. They proposed several expert features to help build surrogate models for evaluating negotiation strategies in the given scenarios. Additionally, they extended them to an online reinforcement learning version, when the learned model is flawed. They trained the surrogate model to predict the average performance when against a set of opponents. Extending their work, (Renting et al., 2020) introduced the sequential model-based optimization mechanism for general algorithm configuration to select strategy parameters for a dynamic agent under a set of opponents and domains. The mechanism searches the configuration space accelerated by an expert-feature-based surrogate model. They also applied AutoFolio to construct a strategy selector by domain and opponent features (Renting et al., 2022). These studies rely on the feature-based surrogate model. However, in this study, we consider a new way of extracting negotiation setting features with CNN.

(Fujita, 2018) proposed an approach to estimate the opponent strategy and preference in multiple times negotiation that could achieve better Pareto efficiency. (Kawata and Fujita, 2020) employed a reinforcement learning method to select the strategy for multiple times negotiation inspired by (Ilany and Gal, 2016). (Taiji and Ikegami, 1999) proposed a strategy for the repeated prisoner's dilemma game that uses a recurrent neural network to predict future interactions with each other. This strategy optimizes the next moves in the game. These works are applicable for the repeated negotiation where the opponent and negotiation domain are fixed. (Güneş et al., 2017) applied boosting on bidding and acceptance strategies. They proposed two versions of boosting learning: learning to select a strategy and learning to combine the output of different strategies. (Sengupta et al., 2021) proposed an adaptive strategy switching mechanism for their autonomous negotiating agent framework. This mechanism could classify the opponent in a negotiation scenario and use the expert recommendation to select the coping strategy. Their results show that they can outperform most existing genius negotiators. Similarly, (Wu et al., 2021) proposed a negotiating agent framework that leverages Bayesian policy reuse in a negotiation. This framework could recognize the opponent and give a coping policy or build a new policy when facing an unseen opponent. These works focused on the strategy of coping with an opponent in a negotiation scenario, and they do not consider the RBAN case.

3 REPEATED-ENCOUNTER BILATERAL AUTOMATED NEGOTIATION

This work considers selecting a strategy from a strategy portfolio for an agent in RBAN. Each negotiation in RBAN is a bilateral automated negotiation that consists of a negotiation protocol, a negotiation scenario, and two negotiators. The negotiation protocol and scenario settings in this paper adopt the bilateral negotiation settings commonly used to evaluate negotiation strategies in literature (Ilany and Gal, 2016; Baarslag et al., 2013; Renting et al., 2020; Renting et al., 2022; Sengupta et al., 2021; Wu et al., 2021).

The negotiation protocol is the alternating offers protocol (AOP) (Rosenschein and Zlotkin, 1994), in which negotiators take turns to make an offer, accept an offer, or walk away. This continues until the deadline is reached or one negotiator agrees or walks away. The deadline can be measured in the number of rounds or real wall-time. The negotiation scenario includes a negotiation domain and two preference profiles. The domain is public information. A preference profile is unique and private information only known to its corresponding negotiator.

A domain D defines a set of issues $I = \{I_1,\ldots,I_i,\ldots,I_{n_{issues}}\}$ with possible values $V_{I_i} = \{v_1^{I_i},\ldots,v_j^{I_i},\ldots,v_{k_i}^{I_i}\}$, where n_{issues} is the number of issues, and k_i is the number of values for issue I_i. A set of values for each issue is referred to as an outcome ω. Ω is the set of all possible outcomes. A preference profile maps each outcome with a real value in $[0,1]$ usually in the form of a utility function. This paper adopts the linear additive utility function $U(\omega) = \sum_{i=1}^{n} w_{I_i} e_{I_i}(\omega[I_i])$ with a reservation value r, where w_{I_i} is the weight of issue I_i ($\sum_{i=1}^{n} w_i = 1$); $e_{I_i}(\cdot)$ is a function that maps the values of issue I_i to real numbers in $[0,1]$; a negotiator will obtain its reservation value if no agreement is reached.

RBAN is a sequence of negotiations with the same opponent under AOP. A negotiation in RBAN could be denoted as a function $\pi(\theta,S)$ of a strategy θ and a scenario $S = (\Omega, U_{n1}, U_{n2}, r_{n1}, r_{n2})$. The strategy selecting problem for a negotiation scenario S_i in a RBAN negotiation sequence $\Pi_q =< \pi_1,\ldots,\pi_i,\ldots,\pi_q >$:

$$\text{argmax}_{\theta_j\in\Theta}\{U_{our}^{i}(\omega_j)\,|\,\omega_j^i \leftarrow \pi_i(\theta_j,S_i),\ \Pi_i\}$$

where $\Pi_i = <\pi_1,\pi_2,\ldots,\pi_{i-1}>$ is a subset of the Π_q, meaning the negotiations before π_i; $\Theta = \{\theta_1,\theta_2,\ldots,\theta_{n_s}\}$ is the strategy space of an agent; and, θ is a strategy configuration that contains a set of numerical or categorical parameters.

4 AGGREGATION CNN SURROGATE MODEL

Figure 1 shows the structure of the proposed ACSM. An ACSM contacts several pre-trained CNN-components with an input layer in parallel and compacts their outputs with an aggregation layer. The output of the aggregation layer is the estimated agreement utility.

The *input layer* includes a discrete outcome-utilities matrix $\mathbb{U}$, reservation values $[r_{n_1}, r_{n_2}]$, and a strategy configuration θ. The strategy configuration θ could be real numbers representing real-valued strategies or one-hot encoded vectors representing categorical strategies. The discrete outcome-utilities matrix $\mathbb{U}(m\times m)$ is calculated from the outcome utilities. Each element $\mathbb{U}_{j,k(1\le j\le m,1\le k\le m)}$ in $\mathbb{U}$ indicates the number of outcomes in the corresponding utility bins and could be calculated using Equation 1. Figure 2 shows an example of transforming the outcome space to $\mathbb{U}(5\times 5)$. Mapping the outcome utility distribution to a size-fixed matrix that indicates the number of outcomes falling within utility bins of predefined makes the CNN-components applicable to domains of different sizes and reduces the computing cost of convolution.

A *CNN-component*, denoted $C_{A_i}(\cdot)$ in Equation 2, is already trained by the negotiation history data with an opponent A_i before integrating into ACSM. Each CNN-component is trained with a unique opponent agent. The output of a CNN-component $C_{A_i}(\cdot)$ is the estimated agreement utility of applying the input strategy on the input negotiation scenario when against the opponent agent A_i. In the training phase, labels are the real obtained agreement utilities of applying negotiation strategies on scenarios against the opponent agent A_i.

$$\hat{u}_{\theta_j}^{A_i} = C_{A_i}(\mathbb{U},\ r_{own}, r_{A_i}, \theta_j), j = 1,\ldots,n_s \qquad (2)$$

The *aggregation layer* is a sigmoid-activated neuron, ensuring the output is scaled to $[0,1]$. Its output is positively correlated to the weighted summation of the outputs of the aggregated CNN-components (Equation 3). Its weights are the online trainable parameters, which make a trade-off between the aggregated CNN-Components. ACSM is expected to be able to fit different opponent agents by adjusting the weights.

$$\begin{aligned}\hat{u}_\theta &= \text{ACSM}(T)\\ &= \text{sigmoid}(\sum_{i=1}^{n} w_{A_i} C_{A_i}(T)) \propto \sum_{i=1}^{n} w_{A_i} C_{A_i}(T)\end{aligned} \qquad (3)$$

where $T = (\mathbb{U}, r_{n_1}, r_{n_2}, \theta_l)$; $C_{A_i}(\cdot)$ denotes a CNN-component trained with an opponent agent A_i; n is

$$
\begin{aligned}
\mathbb{U}_{j,k} &= \left|\{\omega_i \in \Omega \; if \; b^{lower}_{n_1,j} \leq U_{n_1}(\omega_i) \leq b^{upper}_{n_1,j} \; and \; b^{lower}_{n_2,k} \leq U_{n_2}(\omega_i) \leq b^{upper}_{n_2,k}\}\right| \\
b^{lower}_{n_1,j} &= (j-1) \times \frac{1}{m},\; b^{upper}_{n_1,j} = j \times \frac{1}{m},\; b^{lower}_{n_2,k} = (k-1) \times \frac{1}{m},\; b^{upper}_{n_2,k} = k \times \frac{1}{m} \\
(j &= 1,2,\ldots,m;\; k = 1,2,\ldots,m)
\end{aligned}
\tag{1}
$$

where $\mathbb{U}_{j,k(1\leq j\leq m, 1\leq k\leq m)}$ is an element in $\mathbb{U}$ shows the number of outcomes in the corresponding utility bin.

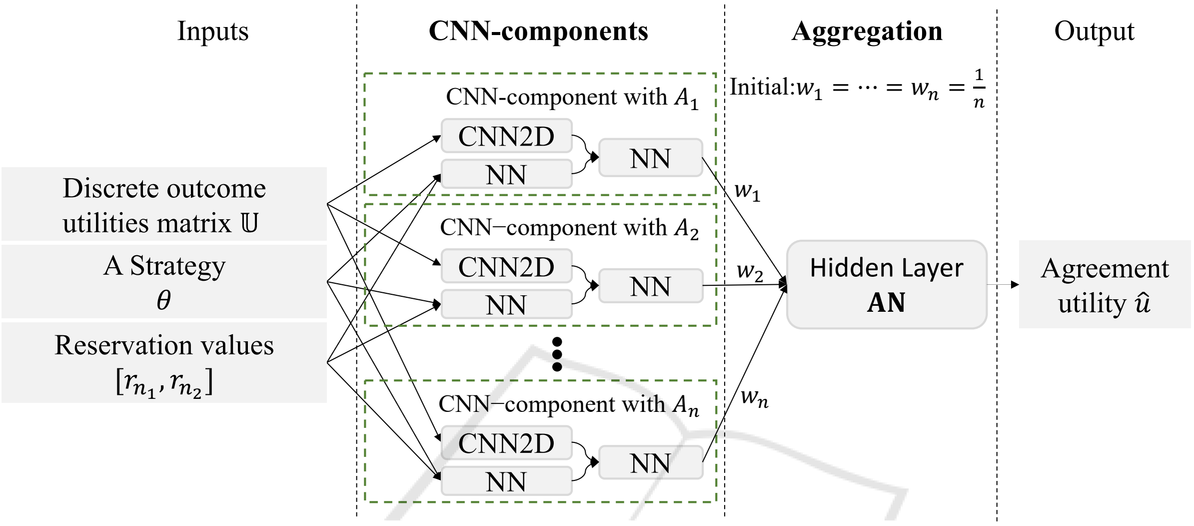

Figure 1: Structure of the proposed ACSM. CNN-component with A_i means it is trained with the negotiation data against the opponent A_i.

the number of aggregated CNN-components.

5 STRATEGY SELECTING MECHANISM OF FINE-TUNING ACSM

Figure 3 demonstrates a strategy selecting mechanism of ACSM with fine-tuning (F-ACSM) for RBAM. There are three primary procedures in the mechanism: initialize ACSM, select a strategy using ACSM with Monte-Carlo method, and fine-tune the surrogate model after each negotiation.

Initialize ACSM. Selecting the strategy that can perform averagely best before getting any information about the facing opponent is rational; hence, the weights of the aggregation layer are initialized to $\frac{1}{n}$ (n is the number of CNN-Components). The output of initial ACSM is positively correlated with the mean value of all component outputs (see Equation 4).

$$
\begin{aligned}
\hat{u}_\theta &= \mathrm{ACSM}_0\left(C_{A_1}(T), C_{A_2}(T), \ldots, C_{A_n}(T)\right) \\
&= \mathrm{sigmoid}\left(\frac{1}{n}\sum_{i=1}^{n} C_{A_i}(T)\right) \propto \frac{1}{n}\sum_{i=1}^{n} C_{A_i}(T)
\end{aligned}
\tag{4}
$$

Select a strategy with ACSM. The mechanism predicts the performance of all possible strategies with ACSM and selects the one with best prediction. To predict the performance of strategies for a given negotiation scenario, ACSM needs the reservation values and $\mathbb{U}$ as inputs, whereas the opponent's reservation value and utility function are private information. One feasible way of overcoming the private information is to sample the unknown part and select the strategy that performs better on the samples, i.e., Monte Carlo method (Figure 4). In this multiple issue linear additive utility case, a sampling assigns a weight w_{I_i} and generates a mapping function $e_{I_i}(v^{I_i}_j), v^{I_i}_j \in V_{I_i}$ for each issue $I_i \in I$ under restrictions (Table 1). The $e_{I_i}(v^{I_i}_j)$ maps a random number to each possible issue value $v^{I_i}_j \in V_{I_i}$. The first proposal restriction assumes an opponent would like to propose the bid that maximizes its utility at the first step (Baarslag et al., 2012; Baarslag et al., 2013; Ya'akov Gal and Ilany, 2015). After sampling, each strategy $\theta_k \in \Theta$ is evaluated using ACSM on all the samples. The average output on the samples is seen as the predicted performance of a strategy.

Fine-tune ACSM. The proposed fine-tuning approach adjusts the weights of the aggregation layer of

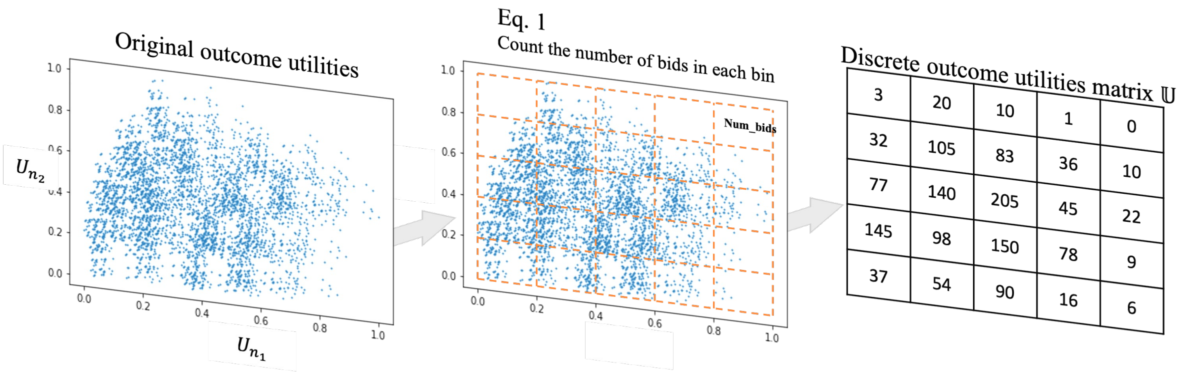

Figure 2: Example of transforming the original outcome utilities to $\mathbb{U}$ (5×5), where each index on the axis correspond to a utility range of the negotiator. The integers filled in the matrix indicate the number of outcomes in the corresponding utility bin.

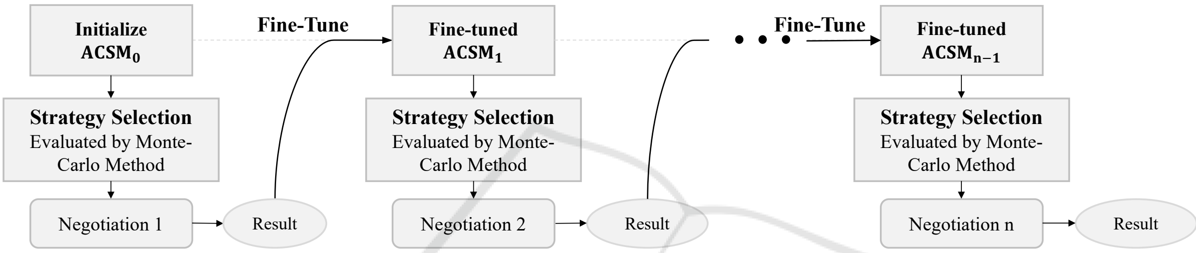

Figure 3: Procedures of the strategy selecting mechanism of F-ACSM model in a RBAN session with n negotiation scenarios.

ACSM after each negotiation (Algorithm 1). First, the Hardheaded opponent model (HOM) (Van Krimpen et al., 2013) estimates the opponent utility function $\hat{U}^k_{opp}$ by feeding with the opponent's bidding history BH_k. Then, a $\hat{\mathbb{U}}_k$ is calculated with the $\hat{U}^k_{opp}$ as Equation 1. The estimated opponent reservation value $\hat{r}^k_{opp}$ is the minimum estimated utility in the opponent's bidding history BH_k calculated by the $\hat{U}^k_{opp}$. Finally, the back-propagation optimizer fine-tunes the aggregation layer of ACSM by using $\hat{\mathbb{U}}_k$ and $\hat{r}^k_{opp}$ as inputs, and the actual utility obtained in π_k as the expected output. Suppose the diversity of the CNN-components of an ACSM is enough. In that case, the behavior pattern of the facing opponent agent must be similar to one or a combination of the training agents. Therefore, adjusting the aggregation layer weights of the ACSM could adapt it to the opponent agent even unknown.

6 EXPERIMENTS

Two experiments of selecting a strategy for a time-dependent agent were performed to evaluate ACSM and F-ACSM, respectively. One experiment of single negotiations compared ACSM with an expert-feature-based neural network surrogate model (NNSM) implemented by ourselves using the same features as in (Ilany and Gal, 2016), showing the capability of the CNN-feature-based surrogate model. Another one of RBAN compared ACSM-only with F-ACSM, presenting the effect of the fine-tuning approach. Both of them applied diverse negotiation scenarios and opponent agents. All experiments were performed on NegMAS of Python (Mohammad et al., 2020).

6.1 Experimental Setup

We evaluated the proposed methods by selecting a parameter for a time-dependent agent using only a time-dependent strategy. This time-dependent strategy is generally adopted by many advanced agents (Ya'akov Gal and Ilany, 2015) and can noticeably affect negotiation results. The time-dependent strategy (Faratin et al., 1998) follows a function: $U_t = 1 - \left(\frac{t}{T}\right)^e$, where T is the maximum negotiation time, and e controls the concession pattern. The lower value of e means that the concession is faster at the start, slower at the end, and vice versa. Usually, e is set in $[0.1, 5.0]$. We limit the strategy space to range $e \in [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.4, 1.6, 1.8, 2.0, 2.5, 3.0, 4.0, 5.0]$

The scenarios for evaluation are from 12 domains of ANAC 2013 (Table 2). Those scenarios are generated uniformly at random, covering multiple situa-

Table 1: Restrictions when sampling the opponent preference profile.

Name of restriction	Equation
Each weight range	$0 < w_{I_i} < 1$
Total weights	$\sum_{I_i \in I} w_{I_i} = 1$
Value mapping function	$0 \leq e_{I_i}(v_j^{I_i}) \leq 1$
First proposal	$e_{I_i}(v_j^{I_i}) = 1 \; if \; v_j^{I_i} \; in \; \omega_{first}, \; v_j^{I_i} \in V_{I_i}, \; I_i \in I$
Reservation value	$r_{opp} = [0.0, 1.0]$

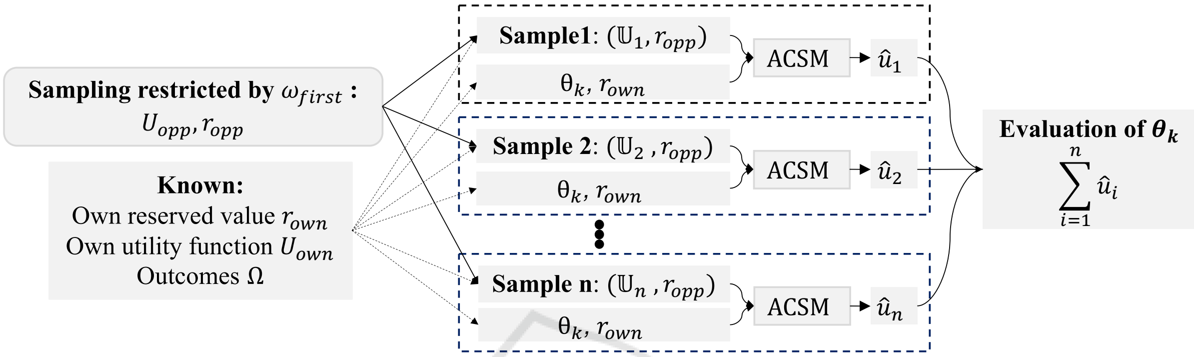

Figure 4: Evaluation of $\theta_k \in \Theta$ in a negotiation scenario. ω_{first} denotes the first bid from the opponent.

tions. The average conflict level is 0.501 with a standard deviation of 0.132. The conflict levels of 94.4% scenarios are located in $[0.237, 0.765]$.

Opponent agents for evaluation are ten different agents (Table 3) including four ANAC champions (Baarslag et al., 2012; Fujita et al., 2013; Mori and Ito, 2017; Aydoğan et al., 2020) and six basic agents (Faratin et al., 1998).

Comparing the strategy-selecting mechanism applying the initial ACSM with one applying the expert-feature-based NNSM is a feasible way of illustrating the effects of the proposed CNN-based feature-extracting approach. The performance metrics are the agreement utility, social welfare, and agreement ratio. Their values are the average of ten times repeated for a negotiation setting. The expert features referred to (Ilany and Gal, 2016) are presented in Table 4. We tested ACSM using three different numbers of Monte Carlo samples: 10, 20, and 30. We found that the differences between 20 and 30 were minimal, so we ultimately chose to use 20 Monte Carlo samples. We set the shape of $\mathbb{U}$ to 100×100, as this size was found to be a good balance between training time and performance when compared to 10×10 and 1000×1000, which were all tested.

Performing the strategy selecting mechanisms applying fine-tuned ACSM and the initial ACSM on RBAN sessions could present the performance of ACSM with and without fine-tuning. One RBAN session included 50 scenarios randomly sampled from the 600 scenarios. 60 RBAN sessions were sampled for testing, avoiding the randomness of one RBAN session. The experiments with ten different opponent agents were performed to demonstrate the efficiency of fine-tuning against different opponent agents. The average agreement utility of 20 times repeats of a session was used as the performance metric. We set the learning rate for fine-tuning to 0.01, as this value was found to perform the best among 0.001, 0.01, and 0.1, which were all tested.

6.2 Training

Six CNN-components are integrated into the ACSM, each trained with a unique opponent agent to predict the agreement utility for a scenario against that agent. The output of the initial ACSM equals the average value of the CNN-components. The baseline method (NNSM) is trained with the scenario's average agreement utility of the same opponent agents used for training CNN-components.

The applied architectures of CNN-component (*Figure 5. a*) and NNSM (*Figure 5. b*) are selected from ten different architectures designed intuitionally. Interestingly, we found that a down-sampling layer of Conv2D (stride = 2) outperforms the one of pooling in learning scenario features.

The 3000 scenarios of training, uniformly randomly sampled from the 12 domains, are ensured to be different from those for evaluation. The evaluation used both the basic agents and ANAC champions, while the training used only six basic agents (Ta-

Algorithm 1: Fine-tuning after a negotiation π_k. Ω_k is the outcome space, θ_k is the used strategy, ω_k is the agreement outcome, if no ω_k then $U^k_{own}(\omega_k) = 0$, BH_k is the opponent bidding history, α is the learning rate, U^k_{own} is own utility function and r^k_{own} is own reservation value, HOM is the hardheaded opponent model.

Require: $\Omega_k, \theta_k, \omega_k, BH_k, \alpha, U^k_{own}, r^k_{own}, \text{ACSM}_{k-1}, \text{HOM}$
Ensure: ACSM_k
$\hat{U}^k_{opp} \leftarrow \text{HOM}(BH_k)$
$\hat{\mathbb{U}}_k \leftarrow \text{Equation1}\left(\hat{U}^k_{opp}, U^k_{own}\right)$
$\hat{r}^k_{opp} \leftarrow \min\left(\left\{\hat{U}^k_{opp}(\omega_i) \middle| \omega_i \in BH_k\right\}\right)$
$\text{ACSM}_k \leftarrow \text{ACSM}_{k-1} - \alpha * \Delta\left(U^k_{own}(\omega_k),\ \text{ACSM}_{k-1}\left(\hat{\mathbb{U}}_k, r^k_{own}, \hat{r}^k_{opp}, \theta_k\right)\right)$

Table 2: Domain information of the experiments.

	Origin Domain	Domain Size		Origin Domain	Domain Size
D1	Lunch	3840	D2	Kitchen	15625
D3	House Keeping	384	D4	Fifty Fifty	11
D5	Defensive Charm	36	D6	Planes	27
D7	Outfit	128	D8	Wholesaler	56700
D9	Dog Choosing	270	D10	Animal	1152
D11	Nice or Die	3	D12	Smart Phone	12000

ble 3); thus, the methods were evaluated on negotiations distributed both homogeneously and heterogeneously with the training set.

The inputs of training CNN-components utilized the opponent's private information for the convergence. The output of training NNSM was the average agreement utility of the six training opponents on a scenario. The batch size is set to 200. All training processes are stopped within 100 steps, although the maximum number is set to 3000. Each training repeated five times with early stopping, and the model with the best validation was used for testing. The validation loss of the CNN-components of time-dependent agents are around 0.06, and those of tit-for-tat agents are around 0.13. The validation loss of NNSM are around 0.11.

6.3 Results

The experimental results of strategy selecting mechanisms applying initial ACSM and NNSM are discussed in *Section 6.3.1.*, comparing CNN-extracted features with expert features. A strategy-selecting mechanism is denoted as the surrogate model applied, i.e., ACSM or NNSM, simplifying the notation. *Section 6.3.2.* presents the differences between applying ACSM with and without fine-tuning against different opponent agents.

6.3.1 ACSM and NNSM

The results are demonstrated from two perspectives. One is the performance against each opponent over all the scenarios, showing the influence of the opponent agent; another is the performance in each domain against all the opponents, showing the influence of scenario size.

Table 5 shows the results against each opponent. Out of the ten opponent agents, ACSM performed not worse in seven in terms of the agreement utility, ACSM performed not worse in eight in terms of the social welfare, and ACSM performed not worse in nine in terms of the agreement rate. The differences between ACSM and NNSM regarding the agreement utility were marginal. One possible reason is that the experiments were to select the strategy parameter for a time-dependent agent, where stubborn strategies were easier to get higher agreement utilities in most cases; hence, both surrogate models learned to select the most stubborn strategy (i.e., $e = 5$) for most scenarios, resulting in the differences being marginal. Notably, ACSM outperformed NNSM regarding social welfare and agreement rate noticeably, although it was not trained for them. We found that ACSM will flex to less stubborn but more reasonable strategies when the scenario is not promising, probably by considering more scenario details, contributing to the higher agreement ratios under similar agreement utilities, thus promoting higher social welfare. Simultaneously, NNSM lost more details when selecting the strategy, resulting in a lower agreement ratio and social welfare.

According to all three performance metrics, the only one of the ten opponent agents that NNSM dominated ACSM was the time-dependent agent (e=1). Its agreement utility value in a scenario was close to all

Table 3: Opponent agents used in this experiment.

Type	Agent Name	Year of ANAC/Strategy
Testing	AgentK	2010
	Hardheaded	2011
	Atlas3	2015
	AgentGG	2019
Training and testing	Time dependent	$e = 0.1$
	Time dependent	$e = 1.0$
	Time dependent	$e = 5.0$
	Tit-For-Tat	$\delta = 1$
	Tit-For-Tat	$\delta = 2$
	Tit-For-Tat	$\delta = 3$

Table 4: Features used in features-based neural network surrogate model. U denotes own utility function.

Type	Description	Equation	Notation
Domain	Number of issues	$\lvert I \rvert$	
	Average number of values	$\lvert \frac{1}{\lvert I \rvert} \sum_{I \in I} \lvert V_I \rvert$	
	Number of outcomes	$\lvert \Omega \rvert$	
Preference	Standard deviation of weights	$\sqrt{\frac{1}{\lvert I \rvert} \sum_{I \in I} \left(w_I - \frac{1}{\lvert I \rvert} \right)^2}$	
	Average utility of Ω	$\frac{1}{\lvert \Omega \rvert} \sum_{\omega \in \Omega} U(\omega)$	$\bar{U}$
	Standard deviation utility of Ω	$\sqrt{\frac{1}{\lvert \Omega \rvert} \sum_{\omega \in \Omega} (U(\omega) - \bar{U})^2}$	
Reservation value	Reservation value	r	
	Percent of Ω above r	$\frac{1}{\lvert \Omega \rvert} \lvert \Omega^r \rvert,\ U(\Omega^r) > r$	
	Average utility of Ω^r	$\frac{1}{\lvert \Omega^r \rvert} \sum_{\omega \in \lvert \Omega^r \rvert} U(\omega)$	$\bar{U}^r$
	Standard deviation utility of Ω^r	$\sqrt{\frac{1}{\lvert \Omega^r \rvert} \sum_{\omega \in \Omega^r} (U(\omega) - \bar{U}^r)^2}$	
Opponent	The utility of the first bid from the opponent	$U(\omega_{first})$	

training agents' average agreement utility value, resulting that NNSM performs like predicting the agreement utility of the time-dependent agent (e=1), which could be one reason that NNSM dominated regarding the time-dependent agent (e=1). In contrast, when selecting a strategy with ACSM, a strategy is evaluated by all CNN-components; thus, the CNN-component that evaluates strategies in a more radical way, i.e., leaves more apparent gaps between the evaluation values assigned to strategies, would have a more significant impact on the final selection; consequently, a strategy would not be preferred even if only one CNN-component assigns it an evaluation value noticeably lower than the other strategies.

Table 6 classifies the results by domain, showing the performance with a different number of outcomes. Out of the 12 domains, ACSM performs not worse in nine in terms of the agreement utility, ACSM performs not worse in ten in terms of the social welfare, and ACSM performs not worse in ten in terms of the agreement rate. Notably, ACSM performed better in the cases when the number of outcomes was greater than 100, i.e., D1-3, D7-10, and D12, demonstrating that ACSM could understand the negotiation settings rich in information better than NNSM. In contrast, in domains D4-6 and D11 with less than 50 outcomes, ACSM performed the same with or slightly worse than NNSM. One reason could be that the outcome-utilities matrices of those domains are too sparse for CNN to extract useful information.

To summarize, the hypotheses that ACSM outperforms NNSM in terms of agreement utility by 0.027%, in terms of social welfare by 0.253%, and in terms of agreement ratio by 0.335% are confident at $\alpha = 0.1$ according to the Mann-Whitney U test ($p = 0.095$). These results indicate that a strategy selection mechanism using ACSM is able to select more advantageous strategies than one using NNSM by making more accurate evaluations.

6.3.2 F-ACSM and ACSM

This section presents the experimental results of performing strategy selecting mechanisms using ACSM

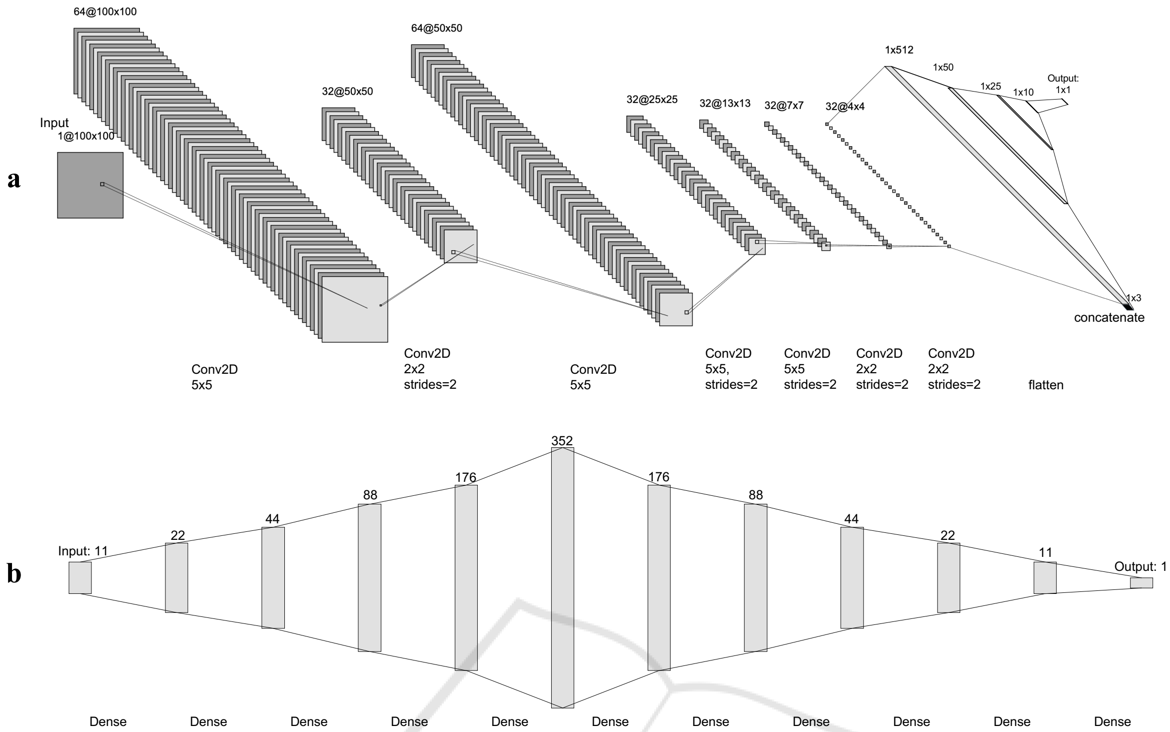

Figure 5: The schematic diagram of CNN-component (a) and NNSM (b).

Table 5: Results of ACSM and NNSM regarding each opponent. Hard denotes the HardHeaded agent. T4T denotes the tit-for-tat agent, Time denotes the time dependent agent, and the number below them means the strategy parameter. AU denotes average agreement utility; SW denotes social welfare; AR denotes agreement rate; AC denotes ACSM; NN denotes NNSM.

		Hard	AgentK	AgentGG	Atlas3	T4T 1	T4T 2	T4T 3	Time 0.1	Time 1	Time 5
AU	AC	**0.684**	**0.754**	**0.746**	0.844	**0.799**	0.808	**0.810**	**0.896**	0.831	**0.799**
	NN	**0.684**	**0.754**	0.737	**0.845**	0.798	**0.809**	**0.810**	0.894	**0.834**	0.794
SW	AC	**1.558**	**1.607**	**1.546**	**1.646**	**1.527**	**1.518**	**1.508**	**1.473**	1.492	**1.610**
	NN	1.554	1.602	1.526	**1.646**	1.525	1.516	1.503	1.468	**1.494**	1.606
AR	AC	**0.709**	**0.796**	**0.753**	**0.915**	**0.773**	**0.778**	**0.767**	**0.828**	0.750	**0.863**
	NN	0.706	0.792	0.738	0.910	0.772	0.775	0.760	0.827	**0.757**	0.852

with fine-tuning (F-ACSM) and one without fine-tuning (ACSM) on RBAN sessions. The results regarding each opponent agent are demonstrated, showing the performances of fine-tuning against different opponent agents. Only agreement utility is used as the performance metric, considering fine-tuning targets achieving a higher agreement utility.

Table 7 presents the results of the ACSM and F-ACSM methods against various opponent agents. According to the Wilcoxon signed ranks test with a confidence level of $\alpha = 0.05$, the results that are bolded in each column are statistically significantly greater than the other. Our analysis shows that F-ACSM consistently outperforms ACSM across a range of opponent agents, both for the training and testing cases. This indicates that the fine-tuning method is effective at allowing ACSM to adapt to the opponent agent being faced in the current scenario, even it that is not used in the training of the CNN components. In most cases, the differences between F-ACSM and ACSM are notable, with only two exceptions: the Hardheaded and Tit-for-Tat ($\delta = 3$) opponent agents. One potential reason for this is that the Hardheaded agent is particularly stubborn, making it difficult to improve the agreement utility through adjustments to the time-dependent strategy. Another possible reason is that the Tit-for-Tat ($\delta = 3$) agent is complex and highly dependent on the scenario, which may make it difficult to learn through fine-tuning, even though its CNN-component has a relatively lower validation accuracy

Table 6: Results of ACSM and NNSM in different domains. D1-12 refer to the domain in Table 2, respectively.

		D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12
AU	AC	**0.765**	**0.847**	**0.806**	**0.805**	0.784	0.746	**0.773**	**0.832**	**0.817**	**0.798**	0.781	**0.812**
	NN	0.763	0.844	0.800	**0.805**	**0.792**	**0.748**	0.769	0.830	0.816	0.797	**0.782**	0.808
SW	AC	**1.480**	**1.621**	**1.570**	**1.545**	**1.564**	1.478	**1.556**	**1.582**	**1.564**	**1.529**	**1.534**	1.560
	NN	1.478	1.612	1.561	1.543	1.562	**1.482**	1.539	1.576	1.558	1.525	**1.534**	**1.561**
AR	AC	0.739	**0.822**	**0.808**	**0.842**	**0.815**	0.706	**0.732**	**0.830**	**0.817**	**0.792**	**0.836**	**0.782**
	NN	**0.743**	0.812	0.800	0.838	0.809	**0.712**	0.722	0.824	0.813	0.788	**0.836**	0.770

Table 7: Average agreement utility of F-ACSM and ACSM with each opponent in the 60 RBAN sessions each including 50 scenarios. F-AC and AC denote F-ACSM and ACSM respectively.

	Hard	AgentK	AgentGG	Atlas3	T4T 1	T4T 2	T4T 3	Time 0.1	Time 1	Time 5
F-AC	**0.6887**	**0.7564**	**0.7464**	**0.8472**	**0.8058**	**0.8114**	0.8108	**0.8953**	**0.8366**	**0.8018**
AC	0.6885	0.7558	0.7455	0.8441	0.8048	0.8108	0.8107	0.8941	0.8353	0.8007

(0.17 compared to the average of 0.10).

Figure 6 demonstrates the changing of agreement utility as the number of scenarios increases, showing that F-ACSM performed better in most cases and on average. Especially, F-ACSM learned the testing-only ANAC agents not slower than the essential agents. We noticed some curves would go down, especially after 20 scenarios. One reason could be that, at first, the surrogate model could be successfully tuned easier when it is far from the truth (the optimal parameter configuration); meanwhile, after tuning, the misleading could cause more deterioration when it is near the truth. Another could be that using the estimated opponent's preference profile as inputs for fine-tuning can mislead the tuning sometimes, also describing when against the Hardheaded agent why F-ACSM deteriorated at first and recovered rapidly.

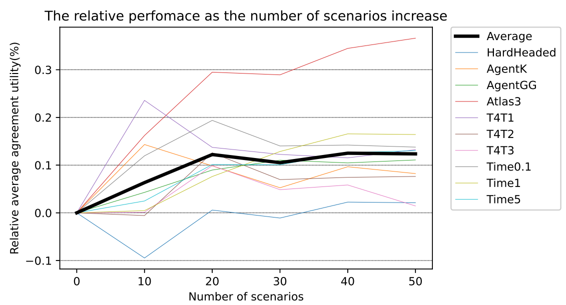

Figure 6: Relative performance between F-ACNM and ACNM with different opponents as the number of scenarios increases. The value is the average for every ten negotiation scenarios over the 60 sessions. The bold line presents the average value of all opponents. T4T denotes the tit-for-tat agent; Time stands for the time dependent Agent, the value after denotes the strategy parameter.

To summarize, F-ACSM achieved an average improvement of 0.13% over ACSM when tested against the 10 opponent agents. This result is statistically significant at the $\alpha = 0.05$ level according to a Mann-Whitney U test. It is expected that F-ACSM would only show slight improvements over ACSM when only selecting a concession speeds parameter for a basic time-dependent agent. Additionally, the changing of relative performance as the number of scenarios increases shows that fine-tuning could adapt ACSM to the opponent gradually in RBAN, although the process may be a tortuous ascent.

7 CONCLUSION AND FUTURE WORK

This paper presented an ACSM and fine-tuning approach for a strategy selecting mechanism applied to RBAN. The ACSM was characterized by using CNN to intelligently extract negotiation scenario features and aggregating different CNNs to ensure the diversity of extracted features. The fine-tuning approach was applied to adjust the weights of CNNs of the ACSM after each negotiation to adapt the ACSM to the facing opponent. The ACSM was higher than the NNSM in agreement utility, social welfare, and agreement ratio in the experimental results of single negotiations with selecting a parameter for a time-dependent agent. This indicated that the CNN-feature-based surrogate model is more promising than the existing expert-feature-based surrogate model. The F-ACSM was higher than the ACSM in agreement utility in the experimental results of RBAN, showing that the fine-tuning method is beneficial for adapting ACSM to the opponent.

In future studies, we will consider overcoming the negative effect of over-fitting (i.e., the deterioration caused by fine-tuning, especially in the late RBAN.).

Designing some early-stopping or dynamic learning rate rules for fine-tuning would be beneficial; however, the scenario uncertainties make it difficult to calculate the opponent model's current accuracy and weigh up its performance after tuning.

ACKNOWLEDGEMENT

This study was supported by JSPS KAKENHI Grant Numbers 22H03641, 19H04216 and JST FOREST (Fusion Oriented REsearch for disruptive Science and Technology) Grant Number JPMJFR216S.

REFERENCES

Aydogan, R., Baarslag, T., Fujita, K., and Jonker, C. (2022). 13th Automated Negotiating Agents Competition (ANAC2022). https://web.tuat.ac.jp/ katfuji/ANAC2022/genius.html.

Aydoğan, R., Baarslag, T., Fujita, K., Mell, J., Gratch, J., de Jonge, D., Mohammad, Y., Nakadai, S., Morinaga, S., Osawa, H., Aranha, C., and Jonker, C. M. (2020). Challenges and main results of the automated negotiating agents competition (ANAC) 2019. In Bassiliades, N., Chalkiadakis, G., and de Jonge, D., editors, *Multi-Agent Systems and Agreement Technologies*, pages 366–381, Cham. Springer International Publishing.

Baarslag, T., Fujita, K., Gerding, E. H., Hindriks, K., Ito, T., Jennings, N. R., Jonker, C., Kraus, S., Lin, R., Robu, V., and Williams, C. R. (2013). Evaluating practical negotiating agents: Results and analysis of the 2011 international competition. *Artificial Intelligence*, 198:73–103.

Baarslag, T., Hindriks, K., Jonker, C., Kraus, S., and Lin, R. (2012). The first automated negotiating agents competition (ANAC2010). In Ito, T., Zhang, M., Robu, V., Fatima, S., and Matsuo, T., editors, *New Trends in Agent-Based Complex Automated Negotiations*, pages 113–135. Springer Berlin Heidelberg, Berlin, Heidelberg.

Chkoniya, V. and Mateus, A. (2019). Digital category management: How technology can enable the supplier-retailer relationship. In *Smart Marketing With the Internet of Things*, pages 139–163. IGI Global.

Faratin, P., Sierra, C., and Jennings, N. R. (1998). Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(3):159–182.

Fujita, K. (2014). Automated strategy adaptation for multi-times bilateral closed negotiations. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems*, AAMAS'14, page 1509–1510, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Fujita, K. (2018). Compromising adjustment strategy based on tki conflict mode for multi-times bilateral closed negotiations. *Computational Intelligence*, 34(1):85–103.

Fujita, K., Ito, T., Baarslag, T., Hindriks, K., Jonker, C., Kraus, S., and Lin, R. (2013). The second automated negotiating agents competition (ANAC2011). In Ito, T., Zhang, M., Robu, V., and Matsuo, T., editors, *Complex Automated Negotiations: Theories, Models, and Software Competitions*, pages 183–197. Springer Berlin Heidelberg, Berlin, Heidelberg.

Güneş, T. D., Arditi, E., and Aydoğan, R. (2017). Collective voice of experts in multilateral negotiation. In An, B., Bazzan, A., Leite, J., Villata, S., and van der Torre, L., editors, *PRIMA 2017: Principles and Practice of Multi-Agent Systems*, pages 450–458, Cham. Springer International Publishing.

Ilany, L. and Gal, Y. (2016). Algorithm selection in bilateral negotiation. *Autonomous Agents and Multi-Agent Systems*, 30(4):697–723.

Kawata, R. and Fujita, K. (2020). Meta-strategy based on multi-armed bandit approach for multi-time negotiation. *IEICE Transactions on Information and Systems*, E103.D(12):2540–2548.

Mohammad, Y., Nakadai, S., and Greenwald, A. (2020). Negmas: A platform for automated negotiations. In *PRIMA 2020: Principles and Practice of Multi-Agent Systems: 23rd International Conference, Nagoya, Japan, November 18–20, 2020, Proceedings*, page 343–351, Berlin, Heidelberg. Springer-Verlag.

Mori, A. and Ito, T. (2017). Atlas3: A negotiating agent based on expecting lower limit of concession function. In Fujita, K., Bai, Q., Ito, T., Zhang, M., Ren, F., Aydoğan, R., and Hadfi, R., editors, *Modern Approaches to Agent-based Complex Automated Negotiation*, pages 169–173. Springer International Publishing, Cham.

Renting, B. M., Hoos, H. H., and Jonker, C. M. (2020). Automated configuration of negotiation strategies. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS'20, page 1116–1124, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Renting, B. M., Hoos, H. H., and Jonker, C. M. (2022). Automated configuration and usage of strategy portfolios for mixed-motive bargaining. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'22, page 1101–1109, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Rosenschein, J. S. and Zlotkin, G. (1994). *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. MIT Press, Cambridge, MA, USA.

Sengupta, A., Mohammad, Y., and Nakadai, S. (2021). An autonomous negotiating agent framework with reinforcement learning based strategies and adaptive strategy switching mechanism. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS'21, page 1163–1172, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Taiji, M. and Ikegami, T. (1999). Dynamics of internal models in game players. *Physica D: Nonlinear Phenomena*, 134(2):253–266.

Van Krimpen, T., Looije, D., and Hajizadeh, S. (2013). Hardheaded. In Ito, T., Zhang, M., Robu, V., and Matsuo, T., editors, *Complex Automated Negotiations: Theories, Models, and Software Competitions*, pages 223–227. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wu, L., Chen, S., Gao, X., Zheng, Y., and Hao, J. (2021). Detecting and learning against unknown opponents for automated negotiations. In Pham, D. N., Theeramunkong, T., Governatori, G., and Liu, F., editors, *PRICAI 2021: Trends in Artificial Intelligence*, pages 17–31, Cham. Springer International Publishing.

Ya'akov Gal, K. and Ilany, L. (2015). The fourth automated negotiation competition. In Fujita, K., Ito, T., Zhang, M., and Robu, V., editors, *Next Frontier in Agent-Based Complex Automated Negotiation*, pages 129–136. Springer Japan, Tokyo.