Coalitional Negotiation Games with Emergent Communication

Xiaoyang Gao, Siqi Chen*, Lin Jie, Yang Yang, Haiying Wu, Jianye Hao

College of Intelligence and Computing, Tianjin University
siqichen@tju.edu.cn

Abstract

A group of agents must cooperate to accomplish their goals when tasks are complex or hard for a single agent. A good example is coalitional games, where a group of individuals form coalitions to produce jointly and share surpluses. In such coalitional negotiation games, how to strategically negotiate to reach agreements on gain allocation is however a key challenge, when the agents are independent and selfish. This work therefore employs deep RL to build autonomous agent called DALSL that can deal with arbitrary coalitional games without human input. Furthermore, DALSL agent is equipped with the ability of exchanging information between them through emergent communication. We have proved that the agent can successfully form a team, distribute the team's benefits fairly, and can effectively use the language channel to exchange specific information, thereby promoting the establishment of small coalition and shortening the negotiation process. DALSL agent can obtain higher payoff when negotiating with handcrafted agents and other RL-based agents; moreover, they can get higher payoff when the language channel is allowed.

Introduction

The research of single agent learning has made significant progress in various domains, including AlphaGo (Silver et al. 2016) for Go, Libratus (Moravcík et al. 2017) for Texas Hold'em, and Watson (Devarakonda and Tsou 2015) for medical diagnosis. However, in today's society, more and more tasks require teamwork to complete successfully and effectively, for example, scoring in a football match, search and rescue, and the parliamentary government system of the EU council (Leech 2002). Cooperation is thus an indispensable condition for the successful completion of these tasks. On the other side, participants may have different abilities and influence over outcomes, leading to their different interests. So there also exists a competitive relationship between them when participants are selfish and try to maximized their own payoff.

One such example is the weighted voting games (WVGs) – a typical problem in cooperative game theory. It reflects the minimum resources needed to complete a task given each agent has some resources. Agents that are selfish need to form coalitions with other participants and reach agreement on how to share the gains among coalition members, based on the the amount of resources they can contribute. As a powerful mechanism, negotiation is often used to facilitate conflict-resolving and achieve consensus between parties of different interests. This problem can therefore be abstracted into a negotiation problem about team formation, which is the research object of cooperative game theory (Shehory and Kraus 1998). Such games are called coalitional negotiation games(Zick et al. 2017), and can be naturally modeled by a multi-agent system.

The success of social tasks such as negotiations requires abilities in many aspects. In the negotiation, the influence of communication on cooperation and negotiation results is the same as or even greater than that of the proposal. Whether the agents can communicate during the negotiation, and how the communication between the agents will affect the negotiation process and results, is what we care about. In the weighted voting game, the formation of coalitions and the distribution of benefits require coordination among multiple agents, which provides a good environment for us to study the communication between multiple agents.

In this paper, deep reinforcement learning (RL) algorithm is used to construct a DALSL (Deep Attention LSTM SARSA(λ)) agent to negotiate with others to form coalitions. During the game, the agent negotiates with the payoff obtained in the game as the supervision signal for the success of the task. In addition, we designed specific language channels and language symbols that agents can use to study the influence of communication on agent negotiation. The results show that DALSL agents are better at negotiating than other agents. Moreover, when the game can use the language channel, communication can effectively promote the negotiation, and have a certain impact on the negotiation process and results. DALSL agents can effectively use the language channel, learn better strategies, and reach more favorable agreements in the game.

Preliminaries

Cooperative Game

According to Nash's definition of cooperative games, each player can form a small group with other players according

*Corresponding author.

to his own interests and cooperate with each other to seek a larger total payment. These small groups are called coalitions, and the coalition composed of all the players is called the grand coalition. Let the set of participants in the game is $N = \{1, 2, ..., n\}$, $|N| = n$ (only considering the limited number of participants), then for any $S \subseteq N$, we call S a coalition of N (Shoham and Leyton-Brown 2008). When $S = N$, it is called a grand coalition. For a given finite set of participants N, the characteristic function form of the cooperative game is an ordered pair (N, v), where the characteristic function v is the mapping from $2^N = \{S|S \subseteq N\}$ to the set of real numbers $\mathbb{R}$, that is, $(N, v) : 2^N \mapsto \mathbb{R}$, and $v(\phi) = 0$. $v(S)$ is called the characteristic function of coalition S, which represents the benefits that participants in the coalition can obtain from mutual cooperation. Therefore, the characteristic function of the cooperative game means that for each coalition S, $v(S)$ is specified to describe the total amount of transferable utility that the coalition S can obtain without the need for the participants outside of S.

WVG is a typical model of cooperative game. In the WVG, we are given a set of participants $N = \{1, 2, ..., n\}$, each participant $i \in N$ corresponding a weight w_i. When the sum of the weights of the coalition S exceeds the given threshold Q, the members of the coalition can obtain payoff. And the coalition S is called the winning coalition, otherwise, it is called the losing coalition.

Multi-agent Reinforcement Learning

In the multi-agent system, each agent obtains reward value by interacting with the environment to learn and improve its own strategy, so as to obtain the optimal strategy in the environment, which is called Multi-Agent Reinforcement Learning (MARL). In single-agent RL, the environment where the agent is located is stable, but in MARL, the environment is complex and dynamic, which brings great difficulties to the learning process. MARL is a Markov game, which combines the stage strategies of each state to form an agent's strategy in a dynamic environment and constantly interacts with the environment to update the game rewards in each state.

A multi-agent Markov game can be expressed as (n, γ, S, A, T, R), where n represents the number of agents, S represents the state space, $A = A_1 \times \cdots \times A_n$ represents joint action space, and $A_i \in A$ represents the action space of the ith agent. $T : S \times A_1 \times \cdots \times A_n \times S \to [0, 1]$ represents the state transition probability, $R_i : S \times A_1 \times \cdots \times A_n \times S \to \mathbb{R}$ represents the return value obtained by the ith agent in the current state and joint action, γ represents the cumulative reward discount coefficient. Markov games also have Markov properties, the next state and reward are only related to the current state and the current joint action. In multi-agent reinforcement learning, each agent independently learns the optimal behavior strategy $\pi_i : S \to A_i$ according to its goal – maximizing its own long-term discounted return.

Stateactionrewardstateaction (SARSA) is a traditional RL method, representing the current state s, action a, reward r and the next step s' and a' respectively. We not only need to know the current s, a, r, but also the next step s' and a'. The SARSA(λ) is an improved version of SARSA. The main difference between the two is that after each take action gets

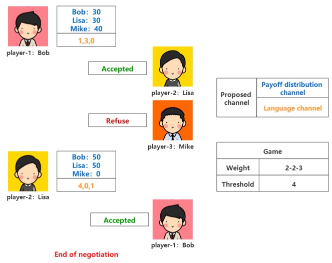


Figure 1: An overview of the negotiation process of RL-based agents.

the reward, SARSA only updates the previous step $Q(s, a)$, and SARSA(λ) updates the step before the reward. The closer the step to the reward, the more important the step is (the attenuation is controlled by the parameter λ).

Coalitional Negotiation Game

To reflect negotiation dynamics of various cooperative tasks in practice, this work propose a coalitional negotiation game. This game is based on the scenarios of weighted voting game. Following the traditional negotiation protocol (Rubinstein 1982), each round of coalitional negotiation game is divided into two stages: proposal and response stage.

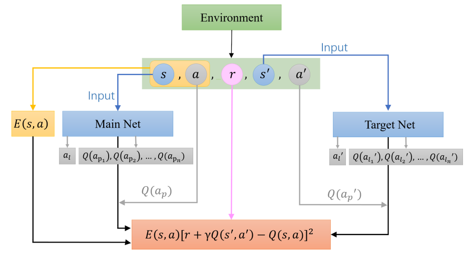
In the proposal stage, the player randomly selected by the system first proposes the coalition proposal with the payoff distribution between those potential members. The proposed coalition scheme must include the proposer and the total weight of the coalition members exceeds the threshold Q. In addition to the coalition proposal, players are allowed to send a language message, which we will explain later. Note that all information in the game like weights, exchanged proposals and language messages of all players, the thresholds, weights, and proposer numbers are made public.

The second stage is the response stage. After receiving the proposal, each member of the coalition can choose to agree or disagree according to his consideration on the situation. Only when all coalition members reach agreement, the proposal can be approved, and the total payoff will be distributed according to the payoff distribution scheme, and the game is over. Otherwise, the proposal is rejected, and no payoff is given. The game will continue until any agreement is settled or the maximum round is reached.

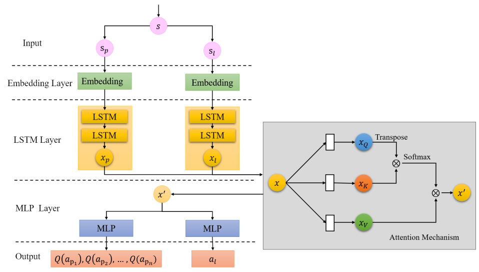
Sharing the information and goals between each other through communication is essential to successfully complete the negotiation. In order to study communication on negotiation, two information channels can be used by players. One is the traditional proposal channel, which is used to directly transmit the potential coalition and distribution of payoff. To be exact, the chosen player $i (i \in S)$ proposes a coalition $S (S \in N)$, and $w(S) = \sum_{i \in S} w_i \geq Q$. The payoff distribution scheme $\vec{p}$ satisfies: $\vec{p} \in \mathbb{R}^n_+$, $supp(\vec{p}) = S$, and $\sum_{i \in S} p_i = r$. The benefit r obtained by agent i in the negotiation is related to its own income p_i in the payoff distribution scheme $\vec{p}$, the discount factor γ and the rounds m used to reach the negotiation, $r_i = p_i \times \gamma^m$.

Another one for communicating information is the language channel. Players in the game can use the language channel to transmit character string to communicate in the form of cheap talk(Dautenhahn 2007). Cheap talk has two key attributes, one is that it is non-binding, and the other is its unverifiable nature, which means that the information sent and received through the language channel will not be enforced and may be a lie. In the negotiation game we specify that the number of symbols that can be used by the agent in the language channel is 5 (characters 0 to 4), and the length of the language message that the agent can generate is 3. Such a setting not only ensures that agents can communicate with enough language symbols, but also avoids the problem that too many characters may make it difficult for agents to learn and use. The process of negotiation between agents is shown in Fig.1.

The experiments are conducted in an online negotiation environment ONECG (Chen et al. 2019), which supports well the underlying coalitional negotiation games. In the negotiation environment, we can easily configure the coalition game specifications, and quickly develop a new agent for negotiation through well-defined APIs. Moreover, when negotiating, agents can not only exchange offers, but also communicate in natural language.



(a) Overall structure diagram.



(b) The structure of network in Figure (a).

Figure 2: Algorithm Structure

Agent Architecture and Learning

We propose a DALSL method for constructing coalitional negotiation game agents, and employ MARL method to train it. Each agent uses SARSA(λ) (Rummery and Niranjan 1994) to learn a strategy independently. The neural network – Long Short Term Memory Network (LSTM) is used to learn Q function in SARSA(λ) to reduce the difficult state space to a manageable number of features and increase the agent's ability to deal with long-term memory, because the negotiation information of the previous round is also very important for the future moves. At the same time, we have added attention mechanism for the agent to further improve the negotiating ability of the agent. The weights of the network are optimized by Adam optimizer with default parameters. The algorithm structure is shown in Fig.2.

The initial input required for model training is the basic configuration parameters of WVGs, including the weight $\overrightarrow{w} = \{w_1, w_2, ..., w_n\} \in \mathbb{R}^n$ of the agents and the threshold Q needed to reach the coalition. In the model, the action set of the proposed agent consists of two parts. The first part is the payoff distribution action set. The action set of the payoff distribution part is n-tuple $\overrightarrow{p} = \{p_1, p_2, ..., p_n\}$, where $\overrightarrow{p} \in \mathbb{N}_0$ and $\sum_{i \in S} p_i = r$. According to the game setting, the coalition S is composed of agents whose payoff in the payoff distribution is not zero, that is to say, $S = \{i | p_i > 0\}$. The second part is the language message action, which is a sequence of prescribed language symbols (y_1, y_2, y_3), where $y_i \in \{0, 1, 2, 3, 4\}$.

The input s obtained by LSTM at each time step t can be divided into two parts, one is the payoff distribution s_p of the agent, and the other is the language messages s_l of the language channel. First, we use the embedding table to convert the two inputs into dense vectors. Although both inputs are numbers, the meaning of payoff distribution and language messages is not the same, so the two inputs correspond to two embedding tables respectively. The LSTM is then used to encode each input sequence, and each input sequence still corresponds to an LSTM followed by the attention mechanism, and the result is two fixed-size vectors. Connect the obtained two vectors and pass through the multi-layer perceptron to get the action state value of the agent at the timestep t for the agent to select the action.

Each DALSL agent is independent, and trained to obtain a higher reward value as the only goal. Each agent learns to map the observation results of the environmental state to action decisions. Agents use RL method to update their strategies by interacting with other agents in the environment, and eventually learn an appropriate behavior strategy.

Experiment

Experimental setup

Our experiment was based on a WVG, with three agents participating in each game. In the experiment, the maximum negotiation round of each negotiation is up to 20, and the discount factor is 0.98 (often used in negotiation to reflect the value of time by reduction of the utility over time)(Chen and Weiss 2014). Experiment configurations are based on a Gaussion distribution over underlying weighted voting games. Each sample generated by this distribution is indicated by the weight vector $\overrightarrow{w}$ and the threshold Q. The sampled set are divided them into training set and testing set. In each group of experiments, we train the agent for 100,000 games, and then evaluate the agent on the test set.

Because designing an agent for every coalitional negotiation game is difficult and time-consuming, We have designed two types of three general negotiation agents as benchmarks. One is based on RL types of agents, proposed by Yoram Bachrach et al. (Bachrach et al. 2020). It is the first time that a deep RL agent is used to solve the team formation negotiation problem in cooperative game theory. We implemented it and named it Bachrach agent. This agent focuses on solving the team formation problem in the negotiation and ignores the communication problem in the negotiation. We improves the Bachrach agent so that it can use the language channel and use the specified language symbols in the negotiation. The improved agent is named Bachrach$^+$ agent.

The other is a handcraft type of agent. Since the handcrafted agent cannot use the language channel for negotiation, its proposed strategy only includes the payoff distribution scheme. We use two baselines to handcraft agents, one is the weighted-proportional agent and the other is the Shapley-proportional agent. Two handcrafted agents randomly select a compliant coalition S. For a given game $(\overrightarrow{w}, Q, r)$, the payoff distribution of the coalition members of the weight-proportional agent is $p_i = (w_i / \sum_{j \in S} w_j) \times r$ (where w_i is the weight of the agent i), and the payoff distribution of the coalition members of the Shapley-proportional agent is $p_i = (\phi_i / \sum_{j \in S} \phi_j) \times r$ (where ϕ_i is the Shapley value of agent i). For the acceptance strategy, we compare the payoff offered to the handcrafted agent in the proposal with the payoff p_i that the handcrafted agent would expect to get in the team. $g_i = r_i - p_i$ represent the difference between the quotation received by the handcrafted agent and the payoff it expects to obtain. A positive value of g_i means that the handcrafted agent is getting more payoff than they expected, and a negative value of g_i means that the handcrafted agent is getting less payoff than they expected. The probability of a handcrafted agent accepting an offer is $\sigma(c \cdot g_i)$, where σ represents the logical function $\sigma(x) = \frac{1}{1+e^{-x}}$, where c is a constant that controls the range of g_i between -1 and 1.

Comparison with other agents

We used two DALSL agents to negotiate with three baseline agents separately (language channels can be used). The results showed that the average payoff obtained by the DALSL agent is 15.93% higher than the Bachrach$^+$ agent, 22.89% higher than the Shapley-proportional agent, and 68.83% higher than the weighted-proportional agent. We carried out the Mann-Whitney U test on the experimental results, and the test results showed that the difference was significant at the $p < 0.005$ level. It can be seen that the DALSL agent has gained a clear advantage in the negotiation.

Moreover, Shapley-proportional agent performance is significantly better than the weighted-proportional agent. By analyzing the negotiation process, it can be found that the probability that the agreement proposed by the weighted-proportional agent was significantly higher than that proposed by the Shapley-proportional agent. When the weighted-proportional agent negotiates with DALSL agents, the probability that the agreement reached is proposed by the weighted-proportional agent is 53.25%. When the Shapley-proportional agent negotiates with DALSL agents, the probability that the agreement reached is proposed by the Shapley-proportional agent is only 37.75%.

The proposal of the weighted-proportional agent is more likely to be accepted, but the payoff is less. In the game where the weight $\overrightarrow{w} = (1, 1, 2)$ and the threshold $Q = 3$, the proposal proposed by the weighted-proportional agent may be $(20, 20, 50)$, $(33, 0, 66)$, $(0, 33, 66)$. The proposal proposed by the Shapley-proportional agent may be $(16, 16, 67)$, $(0, 20, 80)$, $(20, 0, 80)$. It can be found that when the weighting-proportional agent i's weight is $w_i = 2$, its proposal is easier to accept because it "sacrifices" its own payoff. When the weighting-proportional agent i's weight is $w_i = 1$, its proposal may be rejected because its own income is slightly higher. The DALSL agent accepts the proposal of the weighted-proportional agent when it "sacrifices" its own payoff, and rejects the proposal when it is not enough. This leads to the high acceptance rate of the weighted-proportion agent's proposal but the low payoff.

In addition, we examine the flexibility of the DALSL agent. We used the weighted-proportional agent (Shapley-proportional agent) to train the DALSL agent, but used Shapley-proportional agent (weighted-proportional agent) in the evaluation process. This obstacle significantly reduces the average payoff obtained by the DALSL agent in negotiations. However, even in this case, the average payoff obtained by the DALSL agent is still 23.68% higher than the handcrafted agents ($p < 0.005$).

DALSL agent may be sensitive to the initial parameter configuration or the value of hyperparameters during learning. Therefore, we selected robust parameters for the DALSL agent after conducting multiple experiments. Even when the interference learning rate and hidden layer size reach $\pm 20\%$, the DALSL agent performs better than handcrafted agents in the same statistical sense.

The experimental results show that the DALSL agent can not only reach a reasonable negotiation agreement, but also superior to some other agents. Although the performance of the DALSL agent may be inferior to more carefully designed handcrafted agents, the DALSL agent has good generalization, and the generated strategies also have a certain degree of flexibility. Moreover, our method provides a way to construct automated and intelligent agents.

The impact of communication on negotiations

This section uses the DALSL agent to negotiate with the three baseline agents in the case of prohibiting the use of language channels, and contrasts with the content of the previous section to study the influence of communication on the negotiation process and results.

Negotiation between RL-based agents Compared with the experiment that can use the language channel, the payoff obtained by the DALSL agent decreased by 10.55%, and the Bachrach agent decreased by 8.69%. Although the payoff received by the DALSL agent has fallen more, it is still 13.59% higher than the Bachrach agent. The payoff obtained by the agent is not only related to the agreed payoff distribution, but also related to the discount factor and the rounds used to reach the agreement. As shown in Fig.3, when the language channel can be used, the number of rounds for the agent to reach the negotiation agreement is significantly less.

Continue to analyze and find that the number of rounds

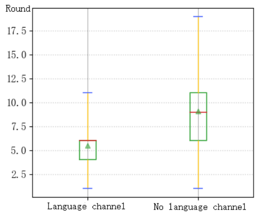
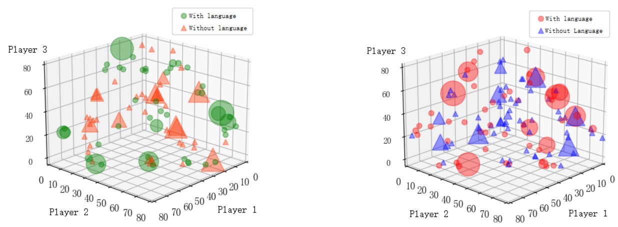


Figure 3: The rounds of agreements reached between RL-based agents.

is directly related to the coalition reached. In negotiations that can use language channels, the proportion of small coalitions reached as high as 89.50%, while in negotiations that cannot use language channels, the proportion of small alliances reached is only 65.75%. For an individual, since the number of agents distributing payoff in the small coalition is less than that in the ground coalition, there is a greater probability of obtaining higher payoff and also a greater risk of being excluded from the winning coalition.

There are two reasons for the above results. First, the coalition only needs the consent of the members who are included in the coalition,while the agents who are excluded from the coalition do not have the right to vote. Therefore, the achievement of a small coalition requires fewer members to agree. Second, in one stage of the game, agents generally earn more in the small coalition than in the ground coalition because the number of agents in the small coalition is less than the number in the ground coalition.



(a) Shapley-proportional agent (b) Weight-proportional agent

Figure 4: Scatter plot of payoff distribution reached between DALSL agents and handcrafted agents.

Negotiation between DALSL agents and handcrafted agent The average payoff obtained by agents in the negotiation of prohibiting the use of language channels, the payoff obtained by DALSL agents decreased by 10.52%, handcrafted agents decreased by 8.15%, while the payoff obtained by DALSL agents was still 39.93% higher than that of handcrafted agents

In the agreement reached, 500 samples were randomly sampled to count the payoff distribution. The results are shown in Fig.4. The reason for random sampling is to avoid too many selected data, which may cause confusion in the image and make it difficult to observe the results. The coordinate axis in Fig.4 represents the payoff obtained by negotiation participants. The circular points indicate that the negotiation is allowed to use language channels, and the triangular points indicate that the negotiation prohibits the use of language channels. The size of the dot shape indicates the number of times the agreement has been reached. Most of the circular points are located on the coordinate axis,which means that in negotiations with language channels, the winning coalitions are more likely to be minor coalitions. This result is consistent with the results obtained from the negotiation of three RL-based agents.

Compared with the negotiation that cannot use the language channel, in the negotiation that can use the language channel, the probability that the winning coalition is a small coalition composed of two DALSL agents has increased significantly by 12.36%. In contrast, the probability that the winning coalition is a small coalition composed of handcrafted agents and DALSL agents dropped by 10.17%. Due to the use of language channels in negotiations, small coalitions are easier to be formed, and at the same time, two handcrafted agents are more likely to be excluded from the winning coalition by the DALSL agent. This also leads to less payoff for handcrafted agents.

Even in a game where only two DALSL agents can use the language channel, the DALSL agent can still use the language channel to form a more favorable small coalition. At the same time, language channel makes DALSL agents form a more stable cooperative relationship, and promotes the cooperation between DALSL agents.

Conclusions

In reality, many tasks require cooperation and coordination. Therefore, the formation of teams and the communication between agents is always a hot topic in multi-agent systems. The work of this paper studies this topic.

The contributions of this paper include the following points. First, we propose a method – DALSL based on deep RL to build agents that can outperform some handcrafted agents and RL-based agents. Secondly, DALSL is completely empirically driven and can be well generalized without the need for manual adjustment and manual data. Finally, the use of language channels enables DALSL agents to achieve better performance in negotiations. In the negotiation, the use of language channels enables RL-based agents to exchange certain information, promote the achievement of small coalitions, and shorten the negotiation process. Moreover, communication promotes the cooperation between the DALSL agents, and makes DALSL agents form a more stable cooperative relationship.

Our experimental environment is based on WVG in cooperative game. In other games, how to form a team between multiple agents, how to communicate, and how the communication will affect the formation of the team and the outcome of the negotiation, are the future research direction that we are interested in.

References

Bachrach, Y.; Everett, R.; Hughes, E.; Lazaridou, A.; and Graepel, T. 2020. Negotiating Team Formation Using Deep

Reinforcement Learning. *Artificial Intelligence*, 288.

Chen, S.; Cui, Y.; Shang, C.; Hao, J.; and Weiss, G. 2019. ONECG: Online Negotiation Environment for Coalitional Games. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2348–2350. Montreal Canada: International Foundation for Autonomous Agents and Multiagent Systems.

Chen, S.; and Weiss, G. 2014. An Intelligent Agent for Bilateral Negotiation with Unknown Opponents in Continuous-Time Domains. *ACM Trans. Auton. Adapt. Syst.*, 9(3): 16:1–16:24.

Dautenhahn, K. 2007. Socially intelligent robots: dimensions of human–robot interaction. *Philosophical transactions of the royal society B: Biological sciences*, 362(1480): 679–704.

Devarakonda, M. V.; and Tsou, C. 2015. Automated Problem List Generation from Electronic Medical Records in IBM Watson. In Bonet, B.; and Koenig, S., eds., *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, 3942–3947. AAAI Press.

Leech, D. 2002. Designing the Voting System for the Council of the European Union. *Public Choice*.

Moravcík, M.; Schmid, M.; Burch, N.; Lisý, V.; Morrill, D.; Bard, N.; Davis, T.; Waugh, K.; Johanson, M.; and Bowling, M. H. 2017. DeepStack: Expert-Level Artificial Intelligence in No-Limit Poker. *CoRR*, abs/1701.01724.

Rubinstein, A. 1982. Perfect Equilibrium in a Bargaining Model. *Econometrica*, 50(1): 97–109.

Rummery, G. A.; and Niranjan, M. 1994. *On-line Q-learning using connectionist systems*, volume 37. University of Cambridge, Department of Engineering Cambridge, UK.

Shehory, O.; and Kraus, S. 1998. Methods for Task Allocation via Agent Coalition Formation. *Artif. Intell.*, 101(1-2): 165–200.

Shoham, Y.; and Leyton-Brown, K. 2008. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489.

Zick, Y.; Gal, K.; Bachrach, Y.; and Mash, M. 2017. How to Form Winning Coalitions in Mixed Human-Computer Settings. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 465–471. ijcai.org.