GENIUS Negotiation Tournament - Group 5

Dipankar Niranjan & Susobhan Ghosh

1 Introduction

Our Group5 agent aims to negotiate in any discrete domain setting, with any given number of adversaries (or negotiators). We have used Bayesian Modeling (Opponent Modeling Phase) followed by an Utility Minimax Heuristic (Optimization Phase) to propose bids to the other agents.

2 Opponent Modeling Phase

We have essentially used Bayesian modeling here, using the frequency of occurences as a metric to judge which bids are most valuable to others and which are not. The basic assumption for this to work is that, all the agents in the negotiation are rational. Irrational agents tend to destabilize our opponent modeling technique. Since it is logical for someone to bid high in initial rounds, and decrease their own utilites as rounds progress, this strategy easily maps out how their utility function stacks up. After determining their utility function, we just use our next phase to optimize over opponent's utilities.

Algorithm 1: Opponent Modeling

```
Data: lastBid, currentBid T
if T == 1 then
    for i = 1 to totalAgents without self do
        Initialize Utility Spaces
        Give equal weights to all issues and values
    end
else
    for i = 1 to totalAgents without self do
        find issues for which currentBid[i][issue] == lastBid[i][issue]
        for each issue in issues do
            weight[i][issue] = weight[i][issue] + ϵ
        end
        normalize weight[i] so that ∑_i weight[i] = 1
        increment value by 1 for all values occuring in currentBid
    end
end
```

3 Optimization and Bidding Phase

Here, after learning the opponent models, we optimize and try to find bids which give the max social welfare, with the constraint that our utility comes out to be the highest. So that means we optimize over the sum of utilities of all other agents, subject to the constraints that our utility is the highest. The optimization is given as:

$$
\begin{aligned}
\max_{bid} \sum_{i \neq self} utility(agent_i, bid) \\
\text{subject to} \\
utility(self, bid) > utility(agent_i, bid)\ \forall i \\
utility(agent_i, bid) > 0.1\ \forall i
\end{aligned} \tag{1}
$$

In the initial rounds till the discount factor evaluates to 0.9, we submit bids of constant value of 0.9. During this phase, if the offer on the table gives us a utility of greater than 0.9, we accept.

Also, we accept offers on the table, whenever we get the highest utility amongst all other agents (using our opponent modeling of their utility function).

In the last 5 rounds, we accept offers if we are getting at least 0.7, and linearly degrade to 0.6 in the deadline round. Below this, we don't accept offers and are happy to not reach an agreement.

4 Results

In our own experiments, with conflicting preference profiles, we tend to win against Agent1 (Pheonix), Agent2 (CUHK), Agent23 (RandomDance) and Agent24(JohnnyBlack) in tournaments, and the Distance to Nash is less than 0.1 in most of the cases when we reach agreements. In 1v1 situations, more often than not, we tend to get the Nash solution itself.

Also, our approach guarantees us to have the best utility, whenever all agents decide to agree. This is something we've tested rigorously over all the agents provided in the GENIUS platform, and have not found one single case where this hasn't happened.