Exercise 5: An Auctioning Agent for the Pickup and Delivery Problem

Group № 38: Andrej Janchevski 309143, Orazio Rillo 313423

November 22, 2020

1 Bidding strategy

The computation of our final price for the auctioned task consists of several steps.

1.1 Computing a tentative bid based on self-interest

Initially, we compute a *tentative bid*, dependent only on our estimated *marginal cost*. In order to compute this value, we keep track of the cost given by the SLS solution through an online computation of the optimal plan with the algorithm submitted for the centralized exercise. After that, we compute the proportion between the *marginal cost* that the auctioned task would incur on our plan (call it c) and the total cost of our current plan; call this proportion r. We then use this coefficient, together with the adjustable parameters m and M, to compute the gain that we wish to get by winning the currently auctioned task with respect to c. The two mentioned parameters represent the minimum (m) and the maximum (M) additive relative gain we request. Finally:

$$g = (1 + M - m)^r + m.$$

We used an exponential function to compute the relative gain in order to discourage our agent from winning tasks with very high marginal cost and, in general, to not be too conservative and exploit the possibility of winning auctions with higher bids. This translates into being *risk seeking*. In simple terms, the policy about "bad" tasks is to either win them with a very high marginal gain or rather lose them. About the "good" ones we try to bid towards the minimum gain we can accept. Our tentative bid is then proportional to the *marginal cost* and g is indeed the proportionality coefficient.

1.2 Utilizing the feedback from opponent's previous bids

We use the information coming from our opponent's previous bids as follows. We keep track of all opponent's bids by storing them in a list. When we get the result of an auction, we update the mentioned list, compute the mean bid of the opponent (μ) and its standard deviation (σ). These are used in order to obtain a 95% CI of the opponent's bids ($\mu \pm 2\sigma$), assuming an underlying Gaussian distribution. We then use this CI to estimate a lower bound for the opponent's bid for a task t. This is done by using a new metric we introduced and called *dissimilarity*. The dissimilarity of t for a player p is:

$$d(t,p) = \frac{1}{2} \min_{c \in C(p)} \frac{dist(c, pickupCity(t))}{D} + \frac{1}{2} \min_{c \in C(p) \cup \{pickupCity(t)\}} \frac{dist(c, deliveryCity(t))}{D},$$

where $C(p)$ is the set of the cities that p has to necessarily visit in order to pickup and deliver his tasks and D is the diameter of the topology graph. The interpretation we give to the introduced metric is the following: if task t's *dissimilarity* is low (i.e. close to 0), t is supposed to be a very "good" task for the opponent, so we expect a bid very close to the lower bound of the CI; on the contrary, if the *dissimilarity* is high (i.e. close to 1), we expect the opponent to bid very close to the upper bound of the confidence interval since it is very likely that this task will be a "bad" one for the opponent. Refer to section 1.3 for an explanation on how to use the *dissimilarity* to compute a lower bound for the opponent's bid.

Moreover, we use the information about the tasks won by the opponent to compute a lower bound for the cost of its plan. We do it again using the previously mentioned implementation of the SLS algorithm. It will result in a lower bound since the opponent's vehicles' home cities are unknown at auction time. In fact, in the computation of the adversary's optimal solution, we do not consider the cost of the path from vehicles' home cities to the first task's pickup city.

1.3 Using the task distribution to speculate about the future

During every auction round, we use the task distribution to compute an expectation of the dissimilarity of the task that will be auctioned in the next round. We compute the expected future dissimilarity of the task t for the player p as below. Assuming that p won the auction for t:

$$\mathbb{E}_{td}[d(t,p)] = \sum_{c_P \in C} \sum_{c_D \in C \setminus \{c_P\}} d(task(c_P, c_D), p) \cdot \mathbb{P}_{td}(c_P, c_D),$$

where td is the task distribution provided by Logist and C is the set of all cities in the topology.

This expected future dissimilarity is used to weigh the choice of the lower bound for the opponent. In fact, we compute for each task t its *discounted dissimilarity* as:

$$dd(t,p) = (1-w) \cdot d(t,p) + w \cdot \mathbb{E}_{td}[d(t,p)],$$

where w is the discount factor of the future. Finally, this coefficient is the one that we use to linearly select the value of the lower bound for the opponent's bid within the confidence interval we estimated.

1.4 Combining all together

Based on the values of the *tentative bid*, the *marginal cost* and the estimated lower bound for the opponent bid ($\mathbb{E}[b]$) we mentioned in the previous sections we perform the appropriate "business practice" to adjust the actual bid we will submit.

When the *tentative bid* is lower than $\mathbb{E}[b]$, we have room to increase our bid and still win the auction. On the contrary, if the *tentative bid* is higher than $\mathbb{E}[b]$, we offer a carefully picked discount computed by discriminating multiple cases based on the value of the *marginal cost*.

Random sampling is performed in all cases in which we need to increase or decrease a bid in order to introduce unpredictability.

2 Results

2.1 Experiment 1: Comparisons with dummy agents

2.1.1 Setting

For this experiment we wished to allow our agent to compete against several different opponents in an auction. The England topology and default distribution configurations were chosen as settings for the tournaments. The number of tasks for each tournament was varied to discover a possible difference in the final outcome. The maximum time available to compute each bid was set to 5 seconds, while the maximum time available to compute the final optimal plan was set to 40 seconds.

The list of all of the opponents which competed against our agent is the following:

A) A modification of the given dummy agent template that always bids its marginal cost;

B) The given dummy agent template, which on average increases its prices as the auction proceeds;

	10 Tasks			20 Tasks			40 Tasks		
Opponent	Round 1	Round 2	Total profit	Round 1	Round 2	Total profit	Round 1	Round 2	Total profit
A	7586 : 0	7871 : 0	15457 : 0	11305 : 0	18588 : 0	29893 : 0	12761 : 0	4988 : 0	17749 : 0
B	7404 : 0	3681 : 535	11085 : 535	16992 : 0	10816 : 0	27808 : 0	11873 : 0	37119 : 220	48992 : 220
C	16975 : 434	7220 : 1636	24195 : 2070	18501 : 536	12319 : 1636	30820 : 2172	17204 : 398	15319 : −6916	32523 : −6518
D	0 : −1055	0 : −2414	0 : −3469	0 : −1143	0 : −871	0 : −2014	−402 : −4255	−323 : 2054	−725 : −2201
E	5307 : 0	4102 : 0	9409 : 0	12014 : 0	3898 : 173	15912 : 173	3790 : 3992	19132 : 173	22922 : 4165
F	3122 : 0	891 : 392	4013 : 392	7411 : 77	−252 : 1751	7159 : 1828	15764 : 173	4094 : 2650	19858 : 2823

Table 1: Results obtained from tournaments run with varying number of tasks. In each result the scores represent the profit of the agents' final plans. The score of our main agent is always given on the left-hand side.

C) A modification of the given dummy agent template that on average decreases its prices in time;

D) A very competitive auction agent. Differs from the main implementation in that it does not utilize any information from its own marginal cost: it only seeks to win every task by estimating the lowest possible opponent bid value using all available information and then bids that price;

E) A self-interested auction agent. Differs from the main implementation in that it does not utilize any information from the future or the opponent's bids and only bids w.r.t. its cost;

F) An auction agent that uses most of the techniques from the main agent. Differs in that it does not utilize any information about the future and estimates the opponent's bid lower bound only using the very conservative and crude objective cost independent of the vehicle parameters.

Note that even though agents A, B and C are dummies the other 3 are fierce competitors that utilize non-trivial ideas which are now part of our final implementation.

2.1.2 Observations

The results from the tournament runs are presented in Table 1. Not so surprisingly, our agent obtains a clear victory when competing against the dummy agents A, B and C. The greedy tentative bid value combined with the idea of estimating the distribution of the past opponent bids coupled with the notion of dissimilarity to estimate a lower bound counters greatly any naive strategy.

Agent D is already a stronger opponent, on a different scale compared to the previous 3. The tactic to always bid lower than your opponent unsurprisingly works extremely well to steal every task and indeed this is what our main agent experienced. However, this strategy is too risky as there is no guarantee that a positive profit value will be obtained. As such, our final implementation humorously won always by default, as it did not incur such high losses unlike the opponent.

Most of our implementation ideas for the estimation of the lower bound for the opponent's bid prices was inspired by the goal to defeat self-interested opponents like agent E. Only taking into account self-convenience does prove to be very effective in conquering any naive agent, as in this manner the total profit from the auction is maximized. However, it turns out to be almost impossible for such an agent to defeat a more intelligent one that also considers the behaviour of its opponent.

Finally, the greatest rival was discovered to be agent F. Alike our final agent, it maximizes its profit when a task with a small estimated cost is auctioned and minimizes its losses when it is in a non-advantageous situation. Victory against this opponent in every round is not guaranteed. However, the disadvantage of this agent is the lack of consideration for the future and the too conservative lower bound for the opponent, restricting the selfishness of the bids. In addition, the cons of pure reliance on the SLS algorithm to estimate the lower bound are very evident: indeed, as the bidding time available is reduced, the agent's estimates become more inaccurate.

To finalize the discussion, we note that with respect to the choice of number of tasks to be auctioned, we do not observe any different outcome: in almost all runs the main agent emerges victorious, with a very similar profit margin w.r.t. each opponent. An expected outcome was also observed: with a higher number of tasks agent F loses with a larger margin, as its computations become heavier and thus more inaccurate, further proving its inferiority.