Optimal bidimensional multi-armed bandit auction for multi-unit procurement

IE 613: Course Project

submitted by

Gampa Varun (150100089), Maitreya Verma (15D100016), Mayuri Bakshi (17I190012)

Under the guidance of

Prof. Manjesh Hanawal

Inter Disciplinary Programme

in

Industrial Engineering and Operations Research

Indian Institute of Technology Bombay

Powai, Mumbai 400076

Contents

Abstract

The multi-armed bandit (MAB) problem is a widely studied problem in machine learning literature in the context of online learning. In this work, our focus is on a specific class of problems namely stochastic MAB problems in an auction setup where the auctioneer gains stochastic rewards from buying items from a set of heterogeneous agents. Each agent offers a cost and a quantity, which influences the reward of the auctioneer along with the unique quality of the agent from which he chooses to buy. In particular, we emphasize stochastic MAB problems with strategic agents in a bi-dimensional mechanism of allocation.

In this project, we will try and employ different algorithms of learning with suitable modifications in the stochastic setting and study the bounds on the regrets obtained in each case. We extend the setting to the adversarial environment and analyze the regret bounds in this case. These settings have a variety of applications involving mechanism based allocations where learning from past actions is crucial.

Chapter 1

Introduction

Auctions have been used since antiquity for the sale of various goods. Auctions enable auctioneer to meet his different goals varying from welfare maximization or utility maximization to revenue maximization or cost minimization. Under auction theory, it is assumed that the players are symmetric. This means that they can be distinguished only by privately held types which can be their costs, valuations, or capacities. However the classical auction theory does not consider the experience of an auctioneer resulting from the consumption of the commodity or service. This experience is something that is not known to the auctioneer and he has to learn the type of the agents over time.

An auction based mechanism is one in which every participating player submits his bid for an object being sold. Depending on all the bids submitted by the agents and the allocation strategy of the auctioneer, the object is given to one of the players. In repeated auctions, along with the object, a reward is associated with each iteration which the auctioneer can use to modify his allocation strategy. Game-theoretic auction models are used to study various strategies in an auction. These models are a mathematical game represented by a set of players, a set of actions (strategies) available to each player, and a payoff vector corresponding to each combination of strategies. Generally, the players are the buyer(s) and the seller(s). These mechanisms are used to allocate resources and goods to various agents in multiple competitive settings. There is a strong assumption here that all the players are anonymous, that is the allocation of goods or resources depends only the submitted bids of players and their inherent quality. For different purposes various mechanisms have been designed.

In this work, a bi-dimensional auction is constructed with the purpose of maximising the utility of the auctioneer while maintaining the dominant strategy of the players to report their true evaluation. While the second price auction is a mechanism in which the dominant strategy for a player is to report their true valuation it does not maximize the auctioneer's utility rather Vickrey auctions maximises social welfare: that is, the winner of the auction is the person who values the item the most. Hence we are interested in allocating resources or goods in such a manner, that based on the qualities of players and the payment made to

them for their services or resources the profit gained by the auctioneer is maximised . But when inherent qualities of the players are unknown stochastic MAB setting is used to glean insights into players by performing the auction mechanism repeatedly to allocate resources to the player who would return the highest profit.

1.1 Motivating Example

A motivating example for designing such mechanisms comes from the problem of choosing an optimum service provider. Consider a hospital (auctioneer) interested in procuring a large number of units of a single generic drug from various pharmaceuticals who can supply limited quantities at different production costs. The quality of the procured generic drug from a supplier can depend on several parameters such as methodology used in preparation and other parameters which are inherent to the supplier. In this example and several other real world scenarios, there is an inherent heterogeneity amongst services or items procured from different agents. Therefore, we can attribute to every agent an inherent quality which is a measure of the perceived experience or reward. Thus, in order to maximize her utility, the auctioneer needs to minimize her payments at the same time ensure a required quality of service. If the qualities from different agents are observed repeatedly, the auctioneer can learn the quality of the agents for future optimization.

Motivated by examples such as above we analyse the 2D-UCB algorithm which learns the qualities of the agents, elicits true costs and capacities from the agents, and maximizes the expected utility of the auctioneer. While the above mechanism converges to allocation principle followed with the full information setting, we show that the regret is also sub linear in time or number resources and further give a lower bound on the regret which limits the rate at which any mechanism can converge. So the contents of the work is as follows:

1. Simulate the results of the previous work in this area and use variations of UCB algorithm to achieve maximum utility quicker.
2. Extend the stochastice MAB setting to adversarial case when the qualities of agents are changing in each round.
3. Show that the 2D-UCB algorithm has the same regret as in the case of UCB algorithm and hence report their regret upper bounds.

Chapter 2

Notations and Prerequisites

2.1 Notations

In the setup considered in [1], an auction is modelled where the auctioneer is buying L units of an item from a number of heterogeneous agents. For better understanding, some notations are introduced:

- $N = \{1, \ldots, n\}$: a set of n agents
- $q_i \in [0, 1]$: quality of ith agent, $q = (q_i)_{i=1}^n$ is the vector of qualities
- $c_i \in [\underline{c_i}, \overline{c_i}]$: true cost o $i$thf agent, $c = (c_i)_{i=1}^n$ is the vector of costs
- $k_i \in [\underline{k_i}, \overline{k_i}]$: true capacity of ith agent, $k = (k_i)_{i=1}^n$ is the vector of capacities
- $\hat{c_i}, \hat{q_i}$: reported values of the cost and quantity of the ith agent
- $b_i = (\hat{c_i}, \hat{q_i})$: bid of the ith agent
- $x = (x_1, x_2, \ldots, x_n)$: allocation, *i.e.*, the total number of units procured from the agents
- $t = (t_1, t_2, \ldots, t_n)$: payments made to the agents

It is assumed that the costs and capacities are independently distributed and their joint density function is given by $f(c_i, k_i)$ and their distribution function is given by $F(c_i, k_i)$. Some other important assumptions are as follows:

- The reward function is linear for the auctioneer and an expected reward of Rq_i is obtained on procuring an unit from agent i where $R \in \mathbb{R}$ is a constant.

- Any agent is not allowed to over-report his capacity since if chosen, he may fail to deliver that many items. In contrast to this, under-reporting can be done since it cannot be detected.

Given all these notations and assumptions, the auctioneer wants to design such a mechanism that maximizes his reward. It can be observed that the mechanism where the agents report their true values will enable the auctioneer to gain maximum reward.

2.2 Preliminaries

Definition 1. ***(Bayesian Incentive Compatible)*** *A mechanism is called Bayesian Incentive Compatible (BIC) if reporting truthfully gives an agent highest expected utility when the other agents are truthful, with the expectation taken over type profiles of other agents.*

Mathematically, $\forall i \in N, \forall \hat{c_i}, c_i \in [\underline{c_i}, \overline{c_i}], \forall \hat{k_i} \in [\underline{k_i}, \overline{k_i}]$,

$$U_i(c_i, k_i, c_i, k_i, q) \geq U_i(\hat{c_i}, \hat{k_i}, c_i, k_i, q),$$

where $U_i(\hat{c_i}, \hat{k_i}, c_i, k_i, q) = -\mathbb{E}_{b_{-i}}[c_i x_i(\hat{c_i}, \hat{k_i}; q) + t_i(\hat{c_i}, \hat{k_i}; q)]$

Definition 2. ***(Dominant Strategy Incentive Compatible)*** *A mechanism is called Dominant Strategy Incentive Compatible (DSIC) if reporting truthfully gives every agent highest utility irrespective of the bids of the other agents*

Mathematically, $\forall i \in N, \forall \hat{c_i}, c_i \in [\underline{c_i}, \overline{c_i}], \forall \hat{ki} \in [\underline{k_i}, \overline{k_i}], \forall \hat{c_{-i}}, \forall \hat{k_{-i}}$,

$$u_i(c_i, \hat{c_{-i}}, k_i, \hat{k_{-i}}, c, k; q) \geq u_i(\hat{c_i}, \hat{c_{-i}}, \hat{k_i}, \hat{k_{-i}}, c, k; q),$$

where, $u_i(\hat{c_i}, \hat{c_{-i}}, \hat{k_i}, \hat{k_{-i}}, c, k; q) = -c_i x_i(\hat{c}, \hat{k}; q) + t_i(\hat{c}, \hat{k}; q)$ is the utility when the true bid profile is c, k and agent i reports $\hat{c_{-i}}, \hat{k_{-i}}$.

Definition 3. ***(Individually Rational)*** *A mechanism is called Individually Rational (IR) if no agent derives negative utility by participating in the mechanism.*

Mathematically, $\forall i \in N, \forall c_i \in [\underline{c_i}, \overline{c_i}], \forall k_i \in [\underline{k_i}, \overline{k_i}]$,

$$U_i(c_i, k_i, c, k; q) \geq 0$$

Definition 4. ***(Optimal Mechanism)*** *A mechanism $(M) = (x, t)$ which procures at most L items is called optimal if it maximizes the expected utility of the auctioneer subject to BIC and IR. The expectation is taken over type profiles of all the agents.*

Definition 5. ***(Regularity)*** *For a virtual cost function defined as*

$$H_i(c_i, k_i) := c_i + \frac{F_i(c_i|k_i)}{f_i(c_i|k_i)}$$

a type distribution is regular if $\forall i \in N$, H_i *is non-decreasing in* c_i *and non-increasing in* k_i.

Definition 6. ***(Reward Realization)*** *A reward realization* s *is an* $n \times L$ *table where the* (i, j) *entry represents an independent realization drawn from the true quality of* i*th agent when procuring the* j*th unit from him.*

Definition 7. ***(Stochastic BIC Mechanism)*** *We say that a mechanism* $\mathcal{M} = (x, t)$ *is Stochastic BIC if truth telling by any agent* i *results in highest expected utility when expectation is taken over reward realizations and type profiles of other agents.*

Mathematically, $\forall \hat{c_i} \in [\underline{c_i}, \overline{c_i}], \forall k_i \in [\underline{k_i}, \overline{k_i}]$,

$$\mathbb{E}_s[U_i(c_i, k_i, c_i, k_i; s)] \geq \mathbb{E}_s[Ui(\hat{c_i}, \hat{k_i}, c_i, k_i; s)]$$

Definition 8. ***(Well-Behaved Allocation Rule)*** *An allocation rule* x *is called a Well-Behaved Allocation if:*

1. *Allocation to any agent* i *for the unit being allocated in round* j , x_i^j , *for any reward realization* s *depends only on the agents bids and the reward realization of* j *units that are procured by the auctioneer so far and is non decreasing in terms of costs.*
2. *For the unit being allocated in round* j *and for any three distinct agents* $\{\alpha, \beta, \gamma\}$ *such that* j*th round unit is allocated to* β. *A change of bid by agent should not transfer allocation of* j*th round unit from* β *to* γ *if other quantities are fixed till* j *units.*
3. *For all reward realizations* s, $x_i(c_i, k_i; s)$ *is non-decreasing with increase in capacity* k_i

Chapter 3

Literature Review

Online Learning has raised the stakes of how any bi-dimensional auctions are conducted strategically these days with not just game theoretic rules applying to it but in also expediting the process of learning from those auctions and incorporating the same in our game's setting. Hence, with online learning, we run back and forth between a learning set up and with whatever information that we acquire upto that point, we try and use that information to aim to reach to our original game's equilibrium whose structure is known but the hidden parameters are being tried to be learned in every round.

Online learning has been a boon to both a seller as well as a bidder where from the seller's perspective, although the bi-dimensional mechanism design considered in [1] supports the seller owing to its important features of Bayesian Incentive Compatibility (BIC) where none of the bidders have any extra incentive in deviating from their true preferences which in our case is $2-$ dimensional *i.e.,* their true valuations of cost and capacity given other agents are truthful and Individual Rationality (IR), where no agent derives negative utility by participating in the mechanism. There were many significant results in single parameter domains, however, the multiple parameter domain was unexplored until recently.

A learning algorithm can be potentially manipulated by a strategic agent so as to increase utility. This problem is addressed using MAB mechanism design theory. Most of the literature in this space considers strategic agents with single dimensional private information and seeks to maximize social welfare. Our work, on the other hand, seeks to maximize the expected utility of the auctioneer. Since the true valuation of the product to be sold is unknown to the auctioneer, he aims to learn this valuation during the series of rounds to eventually allocate as per the allocation mechanism when the true valuations were reported by the sellers. [1] uses Algorithm 2D-OPT whose pseudocode is given as:

3.1 2D-OPT

3.1.1 Algorithm

Algorithm 1: 2D-OPT Mechanism

Input: $\forall i$, Bids $b_i = (\hat{c}_i\, \hat{k}_i)$, reward parameter R
Output: An optimal, DSIC, IR Mechanism $\mathcal{M} = (x, t)$

Allocation is given by $x = \text{ALLOC}(N, \hat{c}, \hat{k}, q, L)$
for $i \in N$ && $x_i \neq 0$ **do**
  $G_i := Rq_i - H_i(b_i)$
  $y = \text{ALLOC}(N \setminus \{i\}, \hat{c}_{-i}, (\hat{k}_{-i} - x_{-i}), q_{-i}, x_i)$
  Payment to i, $t_i = \sum_{k \in N \setminus \{i\}} y_k \max(G_i^{-1}(Rq_k - H_k(b_k)), \bar{c}_i) + \big(x_i - \sum_k y_k\big)\bar{c}_i$
end

Subroutine: $\text{ALLOC}(N^\tau, c^\tau, k^\tau, q^\tau, L^\tau)$

Input: $\langle N^\tau, c^\tau, k^\tau, q^\tau, L^\tau \rangle$ where
$N^\tau =:$ Set of agents,
$c^\tau =:$ Bid vector of costs,
$k^\tau =:$ Bid vector of capacities,
$q^\tau =:$ Vector of qualities,
$L^\tau =:$ Total number of units being allocated.
Output: Vector x of units allocated to each agent.

for $\kappa \in N^\tau$ **do**
  $H_\kappa(c_\kappa^\tau, k_\kappa^\tau) = c_\kappa^\tau + \frac{F_\kappa(c_\kappa^\tau | k_\kappa^\tau)}{f_\kappa(c_\kappa^\tau | k_\kappa)}$
  $G_\kappa := Rq_\kappa^\tau - H_\kappa(c_\kappa^\tau, k_\kappa^\tau)$
end
$(a_1, a_2, \ldots) =$ Sorted indices of agents in N^τ in non-increasing order of G_κ
$x = 0$
$L^{(1)} = L^\tau$
for $1 \leq \eta \leq |N^\tau|$ && $G_{a_\eta} \geq 0$ **do**
  $x_{a_\eta} = \max(k_{a_\eta}^\tau, L^{(\eta)})$
  $L^{(\eta+1)} = L^{(\eta)} - x_{a_\eta}$
end

3.1.2 Performance of 2D-OPT

Since in Algorithm [12], the valuations of the sellers are assumed to be reported truly, the allocation rule given is well behaved. Under this allocation mechanism, maximum possible units are allocated to the agents in decreasing order of Gs, which in turn maximizes the reward of the auctioneer. This allocation mechanism turns out to be valid because of the linear structure of G. In [1], the above stated allocation mechanism is proved to be optimal,

IR and BIC compliant using some strong concepts from game theory.

3.2 2D-UCB

In the Algorithm [12], we had assumed that the agents report their true valuations. We will now see the case where this assumption is relaxed. We move to a more general setting where the true valuations of the agent are taken to be unknown.

3.2.1 Algorithm

Before stating the actual algorithm for 2D-UCB, [1] states two preliminary algorithms, [11] and [3] to simplify the computations

Algorithm 2: Self-resampling Procedure

Input: bid $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, parameter $\mu \in (0, 1)$
Output: (α_i, β_i) such that $\overline{c}_i \geq \alpha_i \geq \beta_i \geq \hat{c}_i$

with probability $(1 - \mu)$
  $\alpha_i \leftarrow \hat{c}_i$, $\beta_i \leftarrow \hat{c}_i$
with probability μ
  Pick $\hat{c}'_i \in [\hat{c}_i, \overline{c}_i]$ uniformly at random.
  $\alpha_i \leftarrow recursive(\hat{c}'_i)$, $\beta_i \leftarrow \hat{c}'_i$

function Recursive($\hat{c}_i$)
  with probability $(1 - \mu)$
    return $\hat{c}_i$
  with probability μ
    Pick $\hat{c}'_i \in [\hat{c}_i, \overline{c}_i]$ uniformly at random.
    return Recursive($\hat{c}'_i$)

Algorithm [11] gives out the estimates of the true cost which is later used in the computation of allocation and payments. Under this algorithm, the input bids are modified through a recursive function. After that, the allocation and payment rule is designed over these modified bids. In the modified bids, α_i is the modified cost borne for the agent i and β_i is a variable that influences the payment done to the agent i.

Algorithm 3: Mechanism Transformation

Input: $\forall i$, bids $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, allocation rule x
Output: Allocation rule $\tilde{x}$ and the payment rule $\tilde{t}$

Obtain modified bids as $(\alpha, \beta) = ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), (\alpha_2(\hat{c}_2), \beta_2(\hat{c}_2)), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$

Allocate according to $\tilde{x}(\hat{c}, \hat{k}) = x(\alpha(\hat{c}), \hat{k})$

Make payment to each agent i, $\tilde{t}_i(\hat{c}, \hat{k}) = \hat{c}_i \tilde{x}_i(\hat{c}, \hat{k}) + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} \frac{x_i(\alpha(\hat{c}), \hat{k})}{\mathcal{F}_i'(\beta_i(\hat{c}_i), \hat{c}_i)}, & \text{if} \beta_i(\hat{c}_i) > \hat{c}_i \\ 0, & \text{otherwise.} \end{cases}$$

The algorithm [3] outputs the transformed allocation and the payment that induces the allocation rule which is stochastic BIC and IR compliant. Now, we quote the 2D-UCB Algorithm:

Algorithm 4: 2D-UCB Mechanism

Input: $\forall i \in N$, bids $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, Reward parameter R
Output: A mechanism $\mathcal{M} = (x, t)$

$\forall i \in N$, $\hat{q}_i^+ = 1$, $\hat{q}_i^- = 0$, $n_i = 1$

Obtain modified bids as (α, β)

$= ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$ using alg:resampling

Allocate one unit to all agents and estimate empirical quality $\hat{q}$

$\hat{q}_i = \tilde{q}_i(i)/n_i$, $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{1}{2n_i} ln(t)}$

for $t = n$ *to* L **do**

  Compute $H_i = \alpha_i + \frac{F_i(\alpha_i|\hat{k}_i)}{f_i(\alpha_i|\hat{k}_i)}$

  Let $i =_{\{js.t.k_j > n_j\}} R\hat{q}_j^+ - H_j$ and $\hat{G}_i = R\hat{q}_i^+ - H_i$

  if $\hat{G}_j > 0$ **then**

    Procure the unit from agent i and update $\hat{q}_i$

    $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{2}{n_i} ln(t)}$

  end

  else

    break \\ Don't allocate future units to anyone

  end

end

Make payment to each agent i, $\tilde{T}_i = \hat{c}_i n_i + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} n_i(\overline{c}_i - \hat{c}_i), & \text{if} \beta_i > \hat{c}_i \\ 0, & \text{otherwise.} \end{cases}$$

Under this mechanism, the unit is allocated to an agent with highest value of $\hat{G}_i$, the expected value of G_i. This G_i can be seen as the gain obtained from the seller i after

ordering q_i units from him and paying him cost H_i. We can see that

If the allocation rule is well-behaved it is seen that using the modified cost the overall mechanism turns out to be stochastic BIC and IR. We note that after modifying the bids there is a constant added to the reward parameter. The value of $\hat{G}_i$ depends on the qualities estimated only from the information in previous trials. So it is in our best interest to choose the agent with maximum expected value of $\hat{G}_i$ as many times as possible. Hence, to analyze its performance we define the regret for the problem as

$$R_q(k, H, \mathcal{M}) = max_{i \in N} \sum_t \mathbb{E}(R * q_i - H_i) - \sum_t \mathbb{E}(R * q_{I_t} - H_{I_t})$$

. But this is valid only when the number of rounds or the number of items to procure is less than the capacity of the player with the highest expected value of $\hat{G}_i$. Factoring in the capacities and assuming services or resources are allocated to n players when there is full information, the regret changes to,

$$\begin{aligned} R_q(K, H, \mathcal{M}) =& (\max_{i_1 \in N}(k_{i_1} * \mathbb{E}(R * q_{i_1} - H_{i_1})) + \max_{i_2 \in N - i_2}(k_{i_2} * \mathbb{E}(R * q_{i_2} - H_{i_2})) + \dots \\ &+ \max_{i_k \in N - i_1, i_2, \dots, i_{k-1}}(k_{i_1} * \mathbb{E}(R * q_{i_k} - H_{i_k}))) - \sum_t \mathbb{E}(R * q_{I_t} - H_{I_t}) \end{aligned} \tag{3.1}$$

such that $k_{i_1} + k_{i_2} + k_{i_3} + ..k_k \geq T$ and $k_{i_1} + k_{i_2} + k_{i_3} + ..k_{k-1} < T$

But a much simpler form can be followed by using the regret decomposition lemma. Using that we get,

$$R_q(k, H, \mathcal{M}) = \sum_{i_j \in N} \mathbb{E}(N_{i_j}(T)) * \Delta_{i_j}$$

Where N_{i_j} is given by,

$$N_{i_j} = \sum_t^T 1_{I_t = i_j and N_{i_j}(t-1) < k_{i_j}}$$

Therefore, we can find upper bound on $\mathbb{E}(N_{i_j}(T))$. And if without loss of generality we assume the $i_1, i_2 \dots i_n$ have expected value of $\hat{G}$ then Δ_{i_j} is $\mathbb{E}(\hat{G_{i_1}} - \hat{G_{i_j}})$ for k_1 number of rounds and $i_j \in N$, and Δ_{i_j} is $\mathbb{E}(\hat{G_{i_2}} - \hat{G_{i_j}})$ for k_2 number of rounds and $i_j \in N - i_1$ and so on. Here we are using Δ_{i_j} is $\mathbb{E}(\hat{G_{i_1}} - \hat{G_{i_j}})$ and $i_j \in N$ for all time as this will be an upper bound else we can use Δ_{i_j} as weighted average of $\mathbb{E}(\hat{G_{i_k}} - \hat{G_{i_j}})$ for $k \in 1, 2 \dots n$ and the

capacities of $i_1, i_2, \ldots i_n$ as weights.

$$\begin{aligned}
\mathbb{E}(N_{i_j}(T)) &= \sum_{t}^{T} Pr(I_t = i_j, N_{i_j}(t-1) < k_{i_j}) \\
&= \sum_{t}^{T} Pr(I_t = i_j) * Pr(N_{i_j}(t-1) < k_{i_j} | I_t = i_j) \\
&\leq \sum_{t}^{T} Pr(I_t = i_j) \\
&\leq 6 * log(T)/\Delta_{i_j}^2 + \pi^2/3 + 1
\end{aligned}$$

Hence the upper bound is,

$$6 * log(T) \sum_{\Delta_{i_j}} 1/\Delta_{i_j} + |N| * (\pi^2/3 + 1)$$

Chapter 4

Stochastic Setting

4.1 Intuition

In this stochastic setting we have a slew of M.A.B algorithms for determining the best agent in each round. We have employed Kl-UCB, Thompson sampling and MOSS algorithm to choose the best agent in round t using estimate of the agents' qualities from reward realizations till $t-1$ rounds. Clearly as these algorithms also have a sub-linear regret in the standard stochastic setting even here the regret will be sub-linear.

4.1.1 2D - MOSS algorithm

We know that (Mini-max optimal strategy for stochastic case) MOSS algorithm has a regret bound of the order of $\sqrt{nT}$ where T is the number of rounds instead of $logT$ which is the case with UCB. Therefore we used the MOSS algorithm to select best agent whose capacity is not yet reached. The optimal agent in round t is chosen as,

$$i_t = \underset{i}{argmax} \hat{q}_i(t-1) + \sqrt{\frac{4}{N_i(t-1)} log^+\left(\frac{T}{nN_i(t-1)}\right)}$$

Where ,

$$log^+(x) = max\left\{1, x\right\}$$

We have pitted this algorithm against the 2D- UCB algorithm for the same mean qualities of the agents where qualities for reward generation come from the Bernoulli distribution. While 2D-UCB showed lesser regret initially it was observed that the rate of decrease of regret was higher in case of 2D-MOSS compared to 2D-UCB, though this depends largely on the qualities chosen also.

Algorithm 5: 2D-MOSS Algorithm

Input: $\forall i \in N$, bids $\hat{c}_i \in [\underline{c}_i, \bar{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, Reward parameter R
Output: A mechanism $\mathcal{M} = (x, t)$

$\forall i \in N$, $\hat{q}_i^+ = 1$, $\hat{q}_i^- = 0$, $n_i = 1$
Obtain modified bids as (α, β)
$= ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$ using alg:resampling
Allocate one unit to all agents and estimate empirical quality $\hat{q}$
$\hat{q}_i = \tilde{q}_i(i)/n_i$, $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{4}{N_i(t-1)} log^+(\frac{T}{nN_i(t-1)})}$
for $t = n$ *to* L **do**
    Compute $H_i = \alpha_i + \frac{F_i(\alpha_i|\hat{k}_i)}{f_i(\alpha_i|\hat{k}_i)}$
    Let $i =_{\{js.t.k_j > n_j\}} R\hat{q}_j^+ - H_j$ and $\hat{G}_i = R\hat{q}_i^+ - H_i$
    if $\hat{G}_j > 0$ **then**
        Procure the unit from agent i and update $\hat{q}_i$
        $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{2}{n_i} ln(t)}$
    end
    else
        break \\ Don't allocate future units to anyone
    end
end
Make payment to each agent i, $\tilde{T}_i = \hat{c}_i n_i + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} n_i(\bar{c}_i - \hat{c}_i), \text{ if} \beta_i > \hat{c}_i \\ 0, \text{ otherwise.} \end{cases}$$

4.1.2 2D- KL-UCB Algorithm

The UCB algorithm uses the upper bound to select the best arm, but KL-UCB finds the mean of another distribution which maximises the divergence with the estimated distribution under a constraint. We know that KL-UCB has a better regret bound than UCB hence in the step of selecting the agent we employed KL-UCB on the estimated qualities, to find the most optimal agent at time t as,

$$i_t = \underset{i_j \in N}{argmax} \; \underset{p}{max} \left\{ p > \bar{q_{i_j}} | d(\bar{q_{i_j}}, p) \leq \frac{log(t) + clog(log(t))}{N_{i_j}(t)} \right\}$$

where $d(\bar{q_{i_j}}, p)$ is the KullbackLeibler divergence. Which is defined as

$$d(P, Q) = \sum_{x \in \mathcal{X}} P(x) log\Big(\frac{P(x)}{Q(x)}\Big)$$

When performed simulations using Bernoulli random variable to generate the quality for rewards we found that KL-UCB it has a sub linear regret, which is to be expected as

Algorithm 6: 2D-KL-UCB Algorithm

Input: $\forall i \in N$, bids $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, Reward parameter R
Output: A mechanism $\mathcal{M} = (x, t)$

$\forall i \in N$, $\hat{q}_i^+ = 1$, $\hat{q}_i^- = 0$, $n_i = 1$
Obtain modified bids as (α, β)
$= ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$ using alg:resampling
Allocate one unit to all agents and estimate empirical quality $\hat{q}$
$\hat{q}_i = \tilde{q}_i(i)/n_i$, $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{1}{2n_i} ln(t)}$
for $t = n$ *to* L **do**
    Compute $H_i = \alpha_i + \frac{F_i(\alpha_i|\hat{k}_i)}{f_i(\alpha_i|\hat{k}_i)}$
    Let $i = \underset{i_j \in N}{argmax} \underset{p}{max} \left\{ p > \bar{q_{i_j}} | d(\bar{q_{i_j}}, p) \leq \frac{log(t) + clog(log(t))}{N_{i_j}(t)} \right\}$
    if $\hat{G}_j > 0$ **then**
        Procure the unit from agent i and update $\hat{q}_i$
        $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{2}{n_i} ln(t)}$
    end
    else
        break \\ Don't allocate future units to anyone
    end
end
Make payment to each agent i, $\tilde{T}_i = \hat{c}_i n_i + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} n_i (\overline{c}_i - \hat{c}_i), & \text{if} \beta_i > \hat{c}_i \\ 0, & \text{otherwise.} \end{cases}$$

we shown earlier that traditional regret bounds work in this case also. Further KL-UCB outperformed the remaining algorithms too, which is to be expected considering KL-UCB has the lowest regret bound.

4.1.3 2D- Thompson Sampling

We have studied that Thompson's algorithm performs better than UCB even though the order of regret is the same for both the algorithms , that is

$$R(T) < \mathcal{O}(logT \sum_i \frac{1}{\Delta_i^2}), \Delta_i \neq 0$$

When running the simulations for the same set of qualities it was observed that both algorithms behaved similarly.

Algorithm 7: 2D-Thompson Sampling Algorithm

Input: $\forall i \in N$, bids $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, Reward parameter R

Output: A mechanism $\mathcal{M} = (x, t)$

$\forall i \in N$, $\hat{q}_i^+ = 1$, $\hat{q}_i^- = 0$, $n_i = 1$

Obtain modified bids as (α, β)

$= ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$ using alg:resampling

Allocate one unit to all agents and estimate empirical quality $\hat{q}$

Let S_i=1 $\forall$ i s.t. $\mathrm{q}_i = 1$

Let F_i=1 $\forall$ i s.t. $\mathrm{q}_i = 0$

for $t = n$ *to* L **do**

  Compute $H_i = \alpha_i + \frac{F_i(\alpha_i|\hat{k}_i)}{f_i(\alpha_i|\hat{k}_i)}$

  Let $i = arg\max_{is.t.\gamma_i > n_i} \beta(S_i + 1, F_i + 1)$ **if** $\hat{G}_j > 0$ **then**

    Procure the unit from agent i and update $\hat{q}_i$

    $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{2}{n_i} ln(t)}$

  end

  else

    break \\ Don't allocate future units to anyone

  end

end

Make payment to each agent i, $\tilde{T}_i = \hat{c}_i n_i + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} n_i(\overline{c}_i - \hat{c}_i), \text{ if} \beta_i > \hat{c}_i \\ 0, \text{ otherwise.} \end{cases}$$

Chapter 5

Adversarial Setting

5.1 Intuition

We have assumed that the inherent quality has a fixed mean in every round. But this need not be the case, so when this assumption is removed we treat the MAB problem in the adversarial setting. As only the quality of agent whose resource we use is obtained, EXP3 and EXP3-IX can be used to allocate resources to the agents. It is seen that when simulated with the reward parameter as a Bernoulli random variable both these algorithms converge to the optimal solution.

5.2 EXP3

Here we have used the standard EXP3 algorithm to select the agent from a probability distribution. Here the regret decomposition lemma cannot be used to transform the original regret but still the upper bound for EXP3 can be used. Indeed it is the case if we consider if the capacity of all players exceed the number of units to be allocated. While running the simulations for a fixed vector of qualities of agents in each round an agent is selected from a probability distribution , where probability of choosing i^{th} agent is,

$$P_{t,i} = \frac{exp(\eta \hat{S_{t-1,i}})}{\sum_j exp(\eta \hat{S_{t-1,j}})}$$

and $\hat{S_{t,i}}$ is updated by,

$$\hat{S_{t,i}} = \hat{S_{t-1,i}} + \frac{1_{\{I_t=i\}} X_t}{P_{t,j}}$$

Algorithm 8: 2D-Exp3 Algorithm

Input: $\forall i \in N$, bids $\hat{c}_i \in [\underline{c}_i, \overline{c}_i]$, $\hat{k}_i \in [\underline{k}_i, k_i]$, parameter $\mu \in (0,1)$, Reward parameter R
Output: A mechanism $\mathcal{M} = (x, t)$

$\forall i \in N$, $\hat{q}_i^+ = 1$, $\hat{q}_i^- = 0$, $n_i = 1$
Obtain modified bids as (α, β)
$= ((\alpha_1(\hat{c}_1), \beta_1(\hat{c}_1), \ldots, (\alpha_n(\hat{c}_n), \beta_n(\hat{c}_n))$ using alg:resampling
Allocate one unit to all agents and estimate empirical quality $\hat{q}$
$\hat{q}_i = \tilde{q}_i(i)/n_i$, $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{1}{2n_i} ln(t)}$
for $t = n$ *to* L **do**
    Compute $H_i = \alpha_i + \frac{F_i(\alpha_i|\hat{k}_i)}{f_i(\alpha_i|\hat{k}_i)}$
    Let $i = \underset{i_j \in N}{argmax} \underset{p}{max} \left\{ p > \bar{q_{i_j}} | d(\bar{q_{i_j}}, p) \leq \frac{log(t) + clog(log(t))}{N_{i_j}(t)} \right\}$
    if $\hat{G}_j > 0$ **then**
        Procure the unit from agent i and update $\hat{q}_i$
        $\hat{q}_i^+ = \hat{q}_i + \sqrt{\frac{2}{n_i} ln(t)}$
    end
    else
        break \\ Don't allocate future units to anyone
    end
end
Make payment to each agent i, $\tilde{T_i} = \hat{c}_i n_i + P_i$, where,

$$P_i = \begin{cases} \frac{1}{\mu} n_i(\overline{c}_i - \hat{c}_i), \text{ if} \beta_i > \hat{c}_i \\ 0, \text{ otherwise.} \end{cases}$$

We have run the simulation for this algorithm given a fixed mean qualities of the agents and observed that the algorithm is converging.

5.3 EXP3-IX

The EXP3 algorithm has a drawback that once an agent gets high enough probability of being chosen then algorithm would exploit that agent even if he fails to be the optimal agent after some rounds, this necessitates for a an inherent exploration term which is found in EXP3-ix algorithm where probability of choosing i^{th} agent is,

$$P_{t,i} = \frac{exp(\eta \hat{S_{t-1,i}})}{\sum_j exp(\eta \hat{S_{t-1,j}})}$$

and $\hat{S_{t,i}}$ is updated by,

$$\hat{S_{t,i}} = \hat{S_{t-1,i}} + \frac{1_{\{I_t=i\}} X_t}{P_{t,j} + \gamma}$$

The γ brings about the exploration phase by making sure that none of the probabilities blow up. For the agents with fixed means chosen for EXP3 it is observed that EXP3-ix performed much better.

Chapter 6

Results

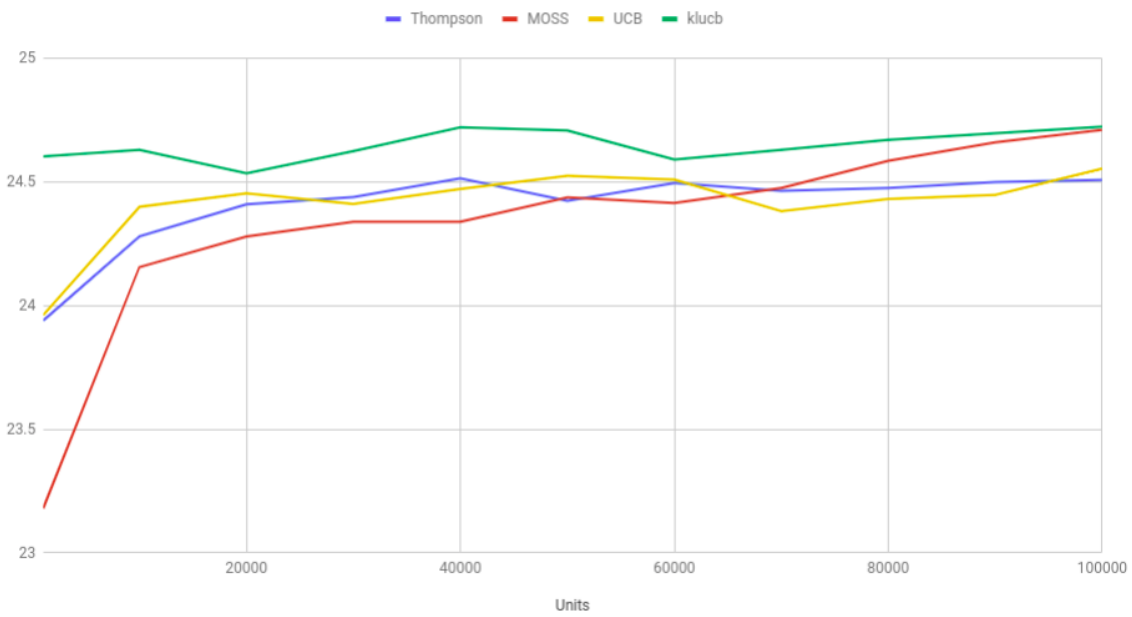

Figure 6.1: Comparing average reward per unit for various stochastic algorithms

The above values were obtained when different algorithms were run for the same random value of quality and different number of units. From this, we can observe that MOSS has a lower reward for lesser number of Units but the reward in Moss increases rapidly as the number of units increase and is equal maximum among the 4 stochastic algorithms used.

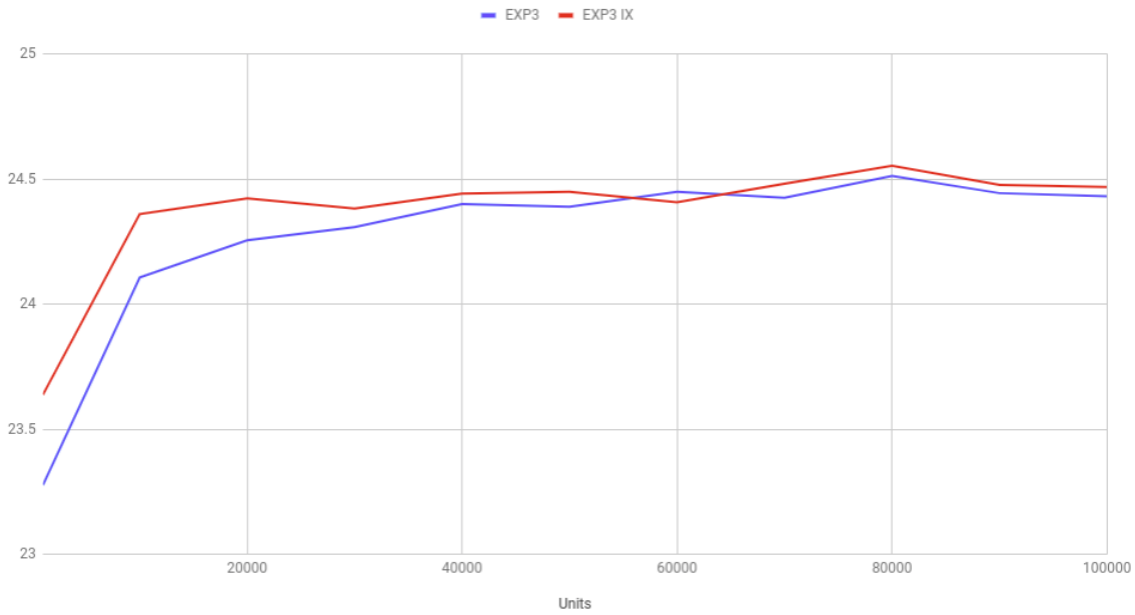

Figure 6.2: Comparing average reward per unit for various adversarial algorithms

The above values were obtained when the two adversarial algorithms were run for the same random value of quality and different number of units. EXP3 has a better performance initially but after a particular number of units, EXP3-IX has a slightly better performance than EXP3.

Chapter 7

Conclusion

Online learning in bi-dimensional mechanism is a learning process which is controlled strategically and not much research done to handle a generalised case of uncountable arms, some famous algorithms that perform well in their respective settings cannot be employed directly. We saw how the concept of penalising under-bidding and over-bidding was captured simultaneously through the definition of regret in auctions that aimed at maximizing expected net revenue. With some existing algorithms working as a base model, we tried to apply some better performing models in the stochastic as well as adversarial setting where Thomson Sampling seemed to perform the best empirically in comparison to UCB-ID and KL-UCB in the case of stochastic setting while Exp3ix-somewhat had a controlled variance throughout but a fluctuating behavior in comparison to the opponent's rule of playing the auctions.

Chapter 8

Future work

This field of applying online learning in bi-dimensional auction is a highly demanded combination. It is a relatively newer area and is open to solving the problem of maximizing the revenue of the bidder across various rounds subject to budget constraints while trying to learn the true valuation under a bi-dimensional mechanism. There is a great scope for algorithms that can handle uncountable arms without an assumption made on them in terms of the rewards that they generate and can be applied freely to scenarios like these. Along the lines of budget unrestricted scenario, we aim to develop theoretical bounds for the algorithms that we proposed for the respective settings and expand the horizon of research by incorporating the seller's perspective also.

Chapter 9

Acknowledgement

We would like to express our gratitude towards our professor in charge of this course, Prof. Manjesh Hanawal, for providing us with an opportunity to explore this area. In addition to this, we would like to thank the TA Arun Verma, for his valuable inputs and guidance throughout the project.

Bibliography

[1] Satyanath Bhat, Shweta Jain, Sujit Gujar, and Yadati Narahari. An optimal bidimensional multi-armed bandit auction for multi-unit procurement. *Annals of Mathematics and Artificial Intelligence*, 85(1):1–19, 2019.