
- You must write up your solutions by yourself, without any collaborators or external references.
- Unless otherwise stated, you may collaborate with other students and consult external references, but may not take written/typed/recorded notes from these interactions.
- You must list all collaborators and external references consulted.
- Some problems will be marked as no collaboration problems. This is to make sure you have experience solving a problem start-to-finish by yourself in preparation for the midterms/final.
- Please upload each problem as a separate file via CS Dropbox.

**Problem 1: Price of Anarchy/Stability in Truthful Auctions (10 points, no collaboration)**

Pick an $n$, and provide a vector of values $v_1, \ldots, v_n$ such that the following two properties hold:

- The price of anarchy of the second-price auction when bidders have these valuations is $\infty$. That is, the welfare of the optimal allocation is non-zero, but there exists a Nash equilibrium of the second-price auction that achieves zero welfare.
- The price of stability of the second-price auction when bidders have these valuations is 1. That is, there exists a Nash equilibrium of the second-price auction that achieves the optimum welfare.

**Problem 2: Bigger Badder Braess’ Paradox (10 points, no collaboration)**

As a function of $\epsilon$, draw/describe two networks $G_\epsilon$ and $G'_\epsilon$ (and cost functions $c_e(\cdot)$ for each edge, and $\vec{r}$, where $r_{ab}$ units of traffic want to travel from $a$ to $b$ for all nodes $a, b$) such that:

1. The nodes in $G_\epsilon$ and $G'_\epsilon$ are the same.
2. All edges in $G_\epsilon$ are present in $G'_\epsilon$, and have the same cost function (but $G'_\epsilon$ may have additional edges). All edge cost functions are continuous, monotone non-decreasing, and non-negative.
3. The unique equilibrium flow in $G_\epsilon$ has total cost at most $1 + \epsilon$.

4. The unique equilibrium flow in $G'_\epsilon$ has total cost 2.

**Problem 3: As easy as pie (15 points).**

Consider the following procedure to partition a round cake (the unit circle, points $(x, y)$ such that $x^2 + y^2 \leq 1$) among three players: Alice, Barbara and Carol. Assume that each player has a valuation function $v_i(\cdot)$ that satisfies $v_i(S) = 0$ whenever $S$ has zero area (i.e. if $S$ is a line, then $v_i(S) = 0$).

- Alice positions three knives on the cake, like the hands of a clock with the tips touching the center and the handle touching the edge. So each knife $i$ has an initial angle $\theta_i$.

- She then rotates them clockwise (increasing each $\theta_i$) in the following manner: at time 0, knife $i$ is at angle $\theta_i$. At time 1, knife $i$ is at angle $\theta_{i+1}$ (and knife 3 is at $\theta_1$). At time $t \in (0, 1)$, knife $i$ is at angle $\theta(i, t) \in (\theta_i, \theta_{i+1})$, and $\theta(i, t)$ is a continuous function of $t$. Because of our assumptions on each $v_i$, all players’ values for each of the three slices are also continuous in $t$.

- Starting at time 0, Alice begins rotating the knives until either Barbara yells stop, or we reach time 1.

- When Alice stops rotating the knives, all three simultaneously cut, and the cake is split into three pieces. Carol chooses one of the three slices, then Barbara selects one of the two remaining slices, leaving the last slice to Alice.

For each player (Alice, Barbara, Carol), provide a strategy ensuring that, no matter how the others play, she prefers her slice at least as much as any other.

**Problem 4: Growth Hacking (15 points)**

In this programming task, your team will play the role of a no-name tech startup in Silicon Valley, desperate to implement your grand vision of a toothbrush sharing economy (or whatever else your startup does). Of course, the most fashionable way to spread the message nowadays is to pay for clicks: participate in auctions for search engine ad slots. The setup is as follows.

Each day $t$, there will be a new keyword up for auction. On that day (and no earlier), your marketing research team will tell you that you should value a click for this keyword at $v_t$ dollars, which you can think of as a real number drawn from an equal revenue curve. More clearly, $v_t$ will always be at least 1, and at most 1000. $v_t$ will be $> x$ with probability $1/x$ for all $x \in [1, 1000)$, and equal to 1000 with probability $1/1000$. All values are drawn independent for each team and independently for each day.

There are 10 ad positions up for auction. The first three have click-through rates of 50%, 15%, and 10%. The next seven have identical rates of 5%. Your value for an ad slot with click-through rate $c$ is $c \cdot v_t$. The auctioneer runs a **generalized second-price (GSP) auction**:

- Each startup submits a sealed bid $b_i \geq 0$.

- The top 10 bidders get the top 10 ad slots in order, with ties broken randomly.
• The winner of the top slot pays the second-highest bid per click, the winner of the second-highest slot pays the third-highest bid per click, and so on. Of course, teams which do not win a slot pay nothing.

This process is repeated for $T = 1000$ days. Initially, each team has a budget of $B = 500$ dollars, and payments are deducted from this budget. Of course, a team is not allowed to bid more than its current balance. (Any bid that does so will be set to 0, and incur a deduction.) Your objective is to maximize the sum of values obtained over all days, plus your remaining budget. Formally, if on day $t$ you win a slot with click-through rate $c_t$ and pay $p_t$, your payoff at the end of the game is $500 + \sum_t v_t \cdot c_t - p_t$.

**Written Submission**

One member of each group should submit a writeup, addressing the following questions (the second bullet is a concrete math question, for the others you should make a strong argument, but there is no right or wrong answer). Like PS3, a couple convincing paragraphs can suffice, and your writeup should absolutely be no longer than three pages.

- Describe and justify your team’s strategy.
- What is the expected value of $v_t$? Make sure to account for the truncation.
- If there were no budget constraint, how much do you think would get spent each round, in total (i.e. summing the payments of all teams in a given round)?
- How (if at all) does the budget constraint affect your program’s behavior?

As in the previous programming assignment, your grade will be based on a combination of correctness and analysis; while bonus/participation points will be available for teams that perform exceptionally well in the game.

**Technical Requirements**

Your class should implement the Bidder interface, which requires the following methods:

- **public double bid(double v):** called once during each day. $v$ will be your value on this keyword. You should return the bid made on this day’s keyword.

- **public void results(int k, double price, double[] bids):** called after each day’s auction is complete, with information about the auction’s results. $k$ will be the position you won: 1 for the top slot through 10 for the tenth slot, or $-1$ if you were not one of the top 10 bidders. $price$ is the GSP price you paid. $bids$ is the sorted (and thus anonymized) array of all bids for this round.

Unlike the previous programming assignment, please name your submission PS5_problem5.java. A simple example is provided in PS5_problem5.java.

---

1 If your writeup is longer than three pages, you won’t be penalized, but graders may ignore everything beyond the first three pages.