15-381 Spring 06 Assignment 4: Game Theory and Auctions

Questions to Vaibhav Mehta(vaibhav@cs.cmu.edu)

Out: 3/08/06 Due: 3/30/06

Name: ____________________________ Andrew ID: ____________________________

Please turn in your answers on this assignment (extra copies can be obtained from the class web page). This written portion must be turned in at the beginning of class at 1:30pm on March 30. The code portion must be submitted electronically by 1:30pm on March 30. Please write your name and Andrew ID in the space provided on the first page, and write your Andrew ID in the space provided on each subsequent page. This is worth 5 points: if you do not write your name/Andrew ID in every space provided, you will lose 5 points.

Code submission. To submit your code, please copy all of the necessary files to the following directory:

`/afs/andrew.cmu.edu/course/15/381/hw4_submit_directory/yourandrewid`

replacing `yourandrewid` with your Andrew ID. **YOU MUST USE C++ for this assignment** since we will be running all the submissions in the same system to compare and rank the submission's outputs. However most of the code is already provided and you will only need to change a few functions with C-style code. All code will be tested on a Linux system, we will not accept Windows binaries. You must ensure that the code compiles and runs in the afs submission directory. Clearly document your program.

Late policy. Both your written work and code are due at 1:30pm on 3/30. Submitting your work late will affect its score as follows:

- If you submit it after 1:30pm on 3/30 but before 1:30pm on 3/31, it will receive 90% of its score.
- If you submit it after 1:30pm on 3/31 but before 1:30pm on 4/01, it will receive 50% of its score.
- If you submit it after 1:30pm on 4/01, it will receive no score.

Collaboration policy. You are to complete this assignment individually. However, you are encouraged to discuss the general algorithms and ideas in the class in order to help each other answer homework questions. You are also welcome to give each other examples that are not on the assignment in order to demonstrate how to solve problems. But we require you to:

- not explicitly tell each other the answers

- not to copy answers
- not to allow your answers to be copied

In those cases where you work with one or more other people on the general discussion of the assignment and surrounding topics, we ask that you specifically record on the assignment the names of the people you were in discussion with (or "none" if you did not talk with anyone else). This is worth five points: for each problem, space has been provided for you to either write people's names or "none". If you leave any of these spaces blank, you will lose five points. This will help resolve the situation where a mistake in general discussion led to a replicated weird error among multiple solutions. This policy has been established in order to be fair to the rest of the students in the class. We will have a grading policy of watching for cheating and we will follow up if it is detected. For the programming part, you are supposed to write your own code for submission.

1 Pure Equilibrium (10 points)

References (names of people I talked with regarding this problem or "none"):

____

Find the equilibria of the two games shown below.

- Game Matrix 1:

	U	V	W
X	40,40	20,60	0,20
Y	60,20	0,0	0,0
Z	20,0	0,0	0,0

Equilibrium are

- *(X, V) with payoff (20, 60)*
- *(Y, U) with payoff (60, 20)*
- *(Z, W) with payoff (0, 0)*

- Game Matrix 2:

	U	V	W
X	20,40	40,20	20,0
Y	40,20	0,20	0,0
Z	0,20	0,0	20,40

Equilibrium are

- *(Y, U) with payoff (40, 20)*
- *(Z, W) with payoff (20, 40)*

2 Mixed Equilibrium

References (names of people I talked with regarding this problem or “none”):

____

Consider the two-player, *zero-sum* game shown in Figure 1. The players use a mixed strategy if

- Player A chooses strategy A_1 with probability p (and, accordingly, strategy A_2 with probability $1-p$)
- Player B chooses strategy B_i with probability q_i (with $\sum_{i=1}^{4} q_i = 1$).

	B_1	B_2	B_3	B_4
A_1	4	3	1	3
A_2	0	5	2	1

Figure 1: Example game.

2.1 (4 points)

Assume that Player A uses mixed strategy p. Write the expected payoff to Player A for each of Player B’s pure strategies B_1, B_2, B_3, B_4 as a function of p.

- $B_1 : 4p$
- $B_2 : 5 - 2p$
- $B_3 : 2 - p$
- $B_4 : 1 + 2p$

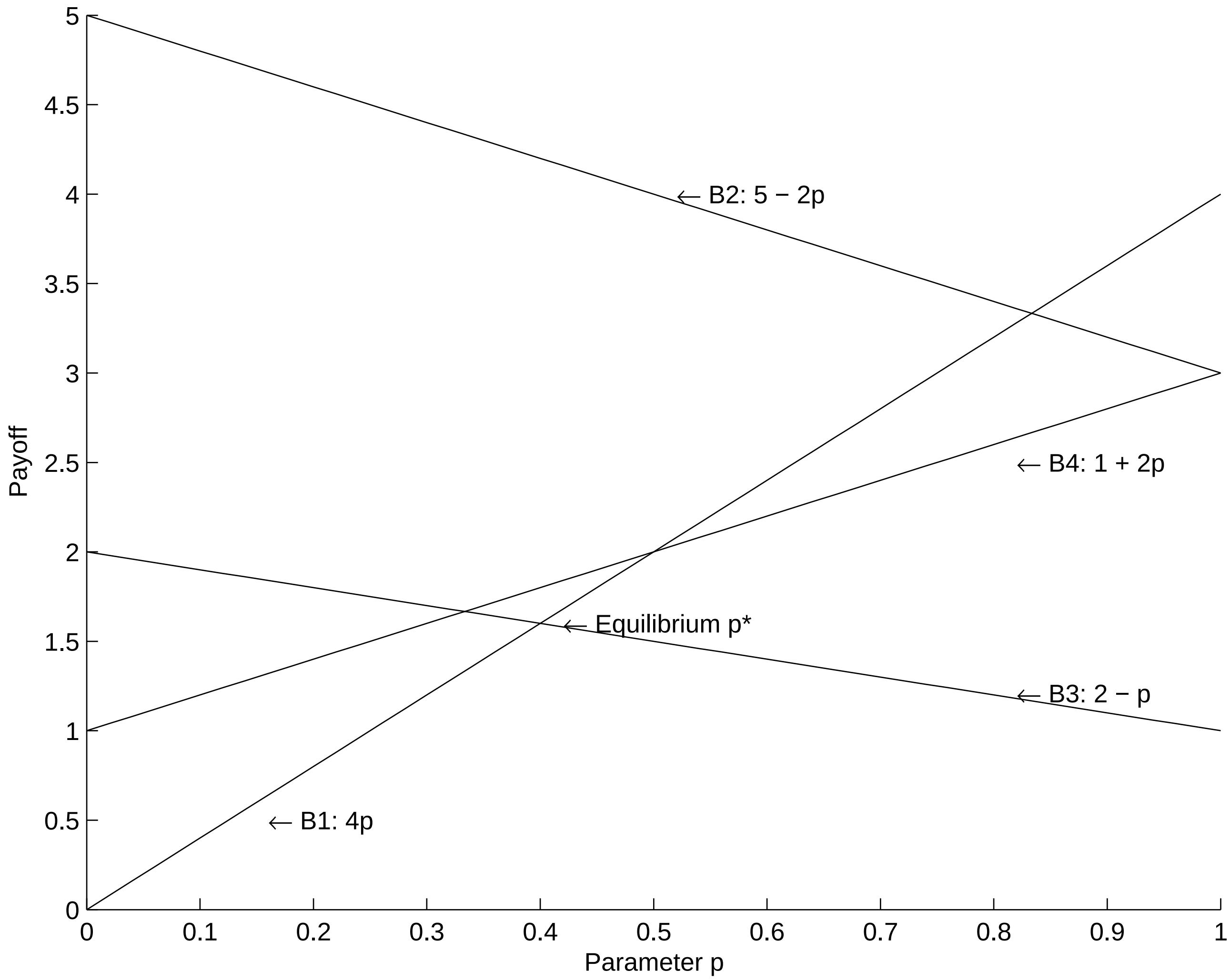

Figure 2: Plot of payoff vs parameter p

2.2 (8 points)

Plot the four resulting functions of p representing the expected payoffs derived in the previous question. Based on this plot, derive the value p^* of the equilibrium mixed strategy for Player A.

Based on the plot, the value of p^* *is* $\frac{2}{5}$.

2.3 (3 points)

By inspection of the plot, show that strategies B_2 and B_4 cannot be part of Player B's mixed strategy solution (in other words, $q_2 = q_4 = 0$ at the solution).

Player B would never choose B_2 and B_4. Based on the plot, we can see that for every value of p, there exists a strategy among B_1 or B_3 which has a better payoff than B_2 or B_4 for player B. So, he never needs to use strategies B_2 or B_4.

2.4 (5 points)

Consequently, Player B's mixed strategy solution can be represented by a single number q between 0 and 1: Player B plays strategy B_1 with probability q and B_3 with probability $1 - q$. Compute the optimal strategy q^* for Player B.

Solving for q, we get $q=\frac{1}{5}$

3 Responding to Intruder

References (names of people I talked with regarding this problem or "none"):

____

Suppose that n robots are moving through an area. An intruder is detected by all the robots simultaneously. The robots do not communicate with each other (in order to remain covert, for example). Each robot has two options: to report the intruder to the central computer or 2) to not report it (and hope that another robot does). All the robots are interested in having the intruder reported (because he may be harmful). This is modeled by assuming that, if *any* robot reports the intruder, then *all* the robots get a reward of v. However, there is a problem: For some reason, communicating back to the central computer is costly. This is modeled by saying that whichever robot decides to communicate incurs a cost of c. As a result, if a robot decides to report the intruder, its total payoff is $v - c$ (and the payoff to any other robot is v). We assume of course that $c < v$. Finally, if *none* of the robots report the intruder, they all get a payoff of 0.

This problem can be modeled as a game in which the players are the n robots; each robot has two possible strategies "Report" or "Ignore"; robot i gets the payoff:

- 0 if *none* of the robots report
- v if one of the other robots reports
- $v - c$ if i reports the intruder

3.1 (6 Points)

What are the pure strategy equilibria of this game, assuming that all the robots are exactly identical and do not communicate?

Pure strategy equilibrium: one robot reports, rest don't

3.2 (4 Points)

Why are these equilibria not very useful, given that the robots are identical and do not coordinate their decisions?

These equilibrium are not very useful because there is no communication. In case of no communication, a particular robot can not know if some other robot is reporting or not, and whether it should also report or not. If only one robot does the reporting always, other robots are useless.

3.3 (15 Points)

More useful are the mixed strategy equilibria for this game. That is, we now assume that each robot i decides to report the intruder with probability p_i. Assuming that all the robots are identical, we can assume that, at the equilibrium, they all follow the same mixed strategy, $p_i = p^*$. Compute the equilibrium mixed strategy p^*.

Hint: Write the expected payoff for one robot, and write that the payoff when all the other robots follow the strategy p^* is maximum (i.e., derivative with respect to the robot's mixed strategy parameter p is zero.)

Hint: We guarantee that this can be solved in a few lines if you remember that there are only three cases of interest in computing the expected payoff: 1) the robot reports 2) the robot does not report but (at least) one other robot does report 3) the robot does not report and nobody else does. If you find yourself filling up pages with complicated notations, you are definitely doing something wrong.

Let p be the probability that a particular robot X reports. Also, p^* is the probability of a robot reporting in equilibrium.

Expected payoff of robot X
$= P(X\ reports) \times (v - c) + P(X\ doesn't\ report) \times P(Atleast\ one\ other\ robot\ reports) \times v$
$= p \times (v - c) + (1 - p) \times (1 - (1 - p^*)^{n-1}) \times v$

Differentiating expectd payoff of X with respect to p, we get $p* = 1 - (\frac{c}{v})^{\frac{1}{n-1}}$

3.4 (5 Points)

What happens to the probability that a given robot reports the intruder, p^* as n increases? What happens to the total probability that *at least one* robot, out of the population of n robots, reports the intruder as n increases[1].

The probability that a given robot reports the intruder decreases as n increases.

[1] If you replace the words "robots" by "people", "intruder" by "crime in progress", and "report intruder" by "call the police", you'll see why the answer to this question is quite disturbing.

Total probability that atleast one robot reports
$= 1 - (1 - p^*)^n$
$= 1 - (\frac{c}{v})^{\frac{n}{n-1}}$

As n increases, total probability of reporting also decreases!

4 Auctions(40 points)

References (names of people I talked with regarding this problem or "none"):

____

You are part of a large advertising firm. Your job is to show advertisements of your client companies on the front page of the newspaper *The XYZ Times*. The newspaper has two slots on the front page, which are auctioned to determine which advertisements to show in these slots and at what price. Note that the two slots are **identical**. You, on behalf of your firm, submit bids for these slots (b_1, b_2). b_1 is your bid for getting one of the slots. b_2 is your bid for getting the second slot, after having obtained one of the slots. For you, the value of having advertisements shown in the newspaper depends on factors like the clients whose advertisements are being shown on that day. You are given your value of showing advertisements as (v_1, v_2) on a particular day, where $v_1 > v_2 > 0$. v_1 is the value of getting one slot for advertisement. v_2 is the value of getting a second slot for advertisement, after having obtained one of the slots already. The winner of the auction is determined by selecting the two maximum bids submitted by all the bidders. A tie is broken by randomly selecting bidders as winners.

The question has a written component, followed by a programming section. You are given 10000 dollars to begin with. The auction will be repeated for 1000 days. On each day, you will be given:

- *my_money* - the total amount of money left with you.
- *my_values* - the value (v_1, v_2) of showing advertisements. Note that v_1 and v_2 are randomly chosen from a fixed set $[100, 200]$ on each day, with $v_1 \geq v_2$.
- *num_bidders* - The number of bidders participating in the auction.

At the end of each day, for each of your advertisement shown on the newspaper, you will be awarded a "profit" equal to your current value for that advertisement minus the price paid by you.

Answer a few written questions first. Each of the following three question introduces a new auction mechanism. An auction mechanism is a set of rules of the auction designed by the auctioneer to achieve a specific outcome. A mechanism is called as incentive compatible if the dominant strategy for the bidders is to submit their true value as their bid. You are supposed to reason out whether each of the given auction mechanisms is incentive compatible or not.

4.1 Discriminatory auction (5 points)

Suppose the price paid by the winner for showing an advertisement is equal to his own bid. (This is the First-price sealed-bid auction discussed in class). If your values are (v_1, v_2), is keeping your bids equal to your values a dominant strategy for you? Why or Why not?

If I bid equal to my values, my expected payoff is 0*. But, if I bid lower than my value, and win, I have a positive payoff. Hence, bidding equal to my value is NOT a dominant strategy.*

4.2 Uniform pricing auction (5 points)

Suppose the price paid for each slot is the same, and equal to the highest rejected bid among all the bids. (This is a multi-unit variant of the Second-price sealed-bid auction discussed in class). If your values for the slots are (v_1, v_2), is keeping your bids equal to your values a dominant strategy for you? Why or Why not?

Bidding equal to my value is NOT a dominating strategy for me. Suppose I bid equal to my value. Also suppose my bids are the highest (or second highest) and third highest bids in the auction. Then, I win one slot and pay price equal to the third highest bid which is also mine. So, lowering that bid would increase my payoff. Hence, bidding equal to my value is not a dominating strategy for me.

4.3 Vickrey Auction (10 points)

Suppose the price paid is according to Vickrey auction. A bidder who wins k slots pays the sum of the k highest rejected bids submitted by *other* bidders. If your values are (v_1, v_2), is keeping your bids equal to your values a dominant strategy for you? Why or Why not?
Hint: Consider separately the cases in which the other people's bids are such that you are winning no slot, one slot and two slots when your bids are (v_1, v_2)

Bidding equal to my values is a dominant strategy.
Suppose I bid equal to my value. I consider three cases in the auction: I am winning no slot, one slot, or two slots.

1. *Winning no slot: If I increase my bids, I can win a slot. But, the price I pay would be greater than my value. Hence, I would have a negative payoff. Decreasing my bids would keep my payoff as* 0 *only. Hence, bidding equal to my values is a dominant strategy in this case.*

2. *Winning one slot: Suppose I change my bids such that I am winning two slots now. Then, the payoff would decrease as I would pay a higher price than my value for the second slot. Suppose I change my bids such that I win no slot, then my payoff decreases too. Hence, bidding equal to my values is a dominant strategy in this case*

3. *Winning two slots: Increasing my bids would not affect my payoff. If I decrease my bids, I might loose one or both of the slots, and my payoff decreases. Hence, bidding equal to my value is a dominant strategy in this case.*

Another equally valid argument is to say that the price I pay is independent of my bids. Hence, decreasing any of my bid can never decrease the price I pay for a slot. Thus, decreasing any of my bid is never useful. Increasing my bid is not useful either, because I could increase my bid over other bids to win a slot but in that case, the price I pay for the slot will also be higher than my value resulting in a negative payoff.

4.4 Programming

Now, we make the auction a little more interesting. On each day, you will also be charged 2 dollars to participate in the auction. For the programming portion, you are free to use any bidding strategies that you find to work well in practice.

4.4.1 Discriminatory Pricing (10 points)

Suppose the price paid by the winner for showing an advertisement is equal to his own bid. Write your function to compute biddings for each day to maximize your profit.

4.4.2 Uniform Pricing (10 points)

Suppose the price paid for each slot is the same, and equal to the highest rejected bid among all the bids. Write your function to compute biddings for each day to maximize your profit.

For this portion of the assignment, you will write functions that, given the input values described above, compute the biddings for each day. At *http://www.cs.cmu.edu/~vaibhav/381/hw4*, you will find a tarred zip file with the code and a README, which has detailed instructions for you to follow. Two of the files are called template.h and template.cpp. Copy and rename those files as *username.cpp* and *username.h* where *username* is your Andrew User id. These files are also placed on the course homepage.

Inside these files, you will find the following function prototypes:

$$double\ disc\_bid\_slot1(\ double\ my\_value1,\ double\ my_value2,\ double\ my\_money,\ int\ num\_bidders\ );$$
$$double\ disc\_bid\_slot2(\ double\ my\_value1,\ double\ my_value2,\ double\ my\_money,\ int\ num\_bidders\ );$$
$$double\ uniform\_bid\_slot1(\ double\ my\_value1,\ double\ my_value2,\ double\ my\_money,\ int\ num\_bidders\ );$$
$$double\ uniform\_bid\_slot2(\ double\ my\_value1,\ double\ my_value2,\ double\ my\_money,\ int\ num\_bidders\ );$$

Change the name of the functions to be $username\_disc\_bid\_slot1$ and $username\_disc\_bid\_slot2$ and $username\_uniform\_bid\_slot1$ and $username\_uniform\_bid\_slot2$. Other than that, you need not change the function prototypes in this file. Some other minor changes are required and are described in the README. note that if you do not wish to participate in the auction, return NO_BID. You will begin the competition with 10,000 dollars.

When you submit your assignment, turn in only *username.c* **and** *username.h*. If your files are not named correctly, or if you have not named your functions correctly, your program will not be graded. In addition to your code, please submit a brief (about half a page) description of your strategy. We will hold an auction for the two auction mechanisms above and the winner (or winners) of each auction will be awarded 5 additional points. The winner will be determined by the agent with the highest wealth at the end of the 1000 days.