1 Routing games with tolls [10 points]

In our discussion of routing games in class, we talked about using tolls to induce independent selfish players to behave in a manner that is optimal for the system. To this end, we discussed the principle of marginal cost pricing. In this exercise, you’ll prove that marginal cost pricing really works.

Consider the model of routing games described in class. Assume that each user $i$ has traffic $r_i = 1$. We add a ‘toll’ to the original link cost functions $c_e(\cdot)$, setting the new cost function to be

$$c^*_e(x) = c_e(x) + (x - 1)(c_e(x) - c_e(x - 1))$$

Notice that the ‘toll’ on link $e$ captures the net latency that each player using the link adds to the other players using it. Prove that any optimal strategy for our original game is a Nash equilibrium for this new game (with link cost functions $c^*_e(\cdot)$).

Hint: Think potential function.

2 Nonatomic routing games [80 points]

The routing games we saw in class were “atomic”, in the sense that each job was a discrete entity, and could not be split up. In this exercise, we’ll introduce you to a class of nonatomic routing games, which you’ll analyze along the same lines as our discussion of “atomic” routing games in class. Be sure to think about how these results parallel/differ from the ones you’ve seen in class.

2.1 The model

A nonatomic routing game is specified as follows. You have a network described by the directed graph $G = (V,E)$, where $V$ is the set of vertices, $E$ is the set of links. Each link $e \in E$ is associated with a cost function $c_e : \mathbb{R}_+ \to \mathbb{R}_+$, such that $c_e(f_e)$ is the cost of carrying flow $f_e$ on link $e$. We assume the cost functions $c_e(\cdot)$ are nondecreasing and continuous. A set of commodities $I$ need to be routed through this network. Each commodity $i \in I$ is associated with a source node $s_i$, a destination node $d_i$, a total flow
volume \( r_i \), and a set \( P_i \) of paths connecting \( s_i \) to \( d_i \) that can be used by to route its flow. We assume that \( P_i \neq \emptyset \), and that \( P_i \cap P_j = \emptyset \) for \( i \neq j \).

Let \( P = \bigcup_{i \in I} P_i \) denote the set of all network paths. A flow vector \( f = (f_p, p \in P) \in \mathbb{R}_{+}^{|P|} \) specifies the amount of flow routed along each path. \( f \) is feasible if, for each commodity \( i \in I \), \( \sum_{p \in P_i} f_p = r_i \).

For a feasible flow vector \( f \), the flow on a link \( e \) is given by \( f_e = \sum_{p: e \in p} f_p \). The cost of a path \( p \) is defined as \( c_p(f) := \sum_{e \in p} c_e(f_e) \).

**Definition of equilibrium:** A feasible flow \( f \) is an equilibrium flow for the nonatomic game if, for each commodity \( i \in I \), and every pair of paths \( p, p' \in P_i \), with \( f_p > 0 \),

\[
    c_p(f) \leq c_{p'}(f).
\]

**Objective function:** The cost of a flow \( f \) is defined by

\[
    C(f) := \sum_{p \in P} f_p c_p(f) = \sum_{e \in E} f_e c_e(f_e).
\]

We will say a feasible flow is optimal if it minimizes \( C(f) \) over all feasible flows.

---

**Example - Braess's paradox:** To illustrate the model, let’s look at an example showing Braess’s paradox for nonatomic routing games. Consider the network shown on the left panel of Fig. 1. There is a single commodity, with source \( s \), destination \( d \), and flow volume \( r = 1 \) using the network. This flow volume may be split across two parallel paths, as shown. At equilibrium, convince yourself that the flow volume is split equally between the two paths, so that the cost of the equilibrium flow equals \( \frac{3}{2} \).

We now add the new zero cost link \( u \rightarrow v \) to the network; see the right panel of Fig. 1. There are now three paths between \( s \) and \( d \). The cost of the path \( s \rightarrow u \rightarrow v \rightarrow d \) is strictly smaller that that of the original two paths, as long as any of the flow volume is not routed through it. As a result, the unique equilibrium corresponds to routing all full flow volume along the path \( s \rightarrow u \rightarrow v \rightarrow d \). Therefore, the cost of the equilibrium flow equals 2, which exceeds the cost before the new link was added!
**Your task:** Now that we have described the model, your task will be to analyze it, answering the following questions (the same questions we considered in class when studying load balancing games and routing games).

1. Does an equilibrium always exist?

2. How efficient are the equilibria, i.e., what is the price of anarchy (PoA) and the price of stability (PoS)?

### 2.2 Part 1 - Existence of equilibria

For Part 1, proceed along these lines.

(a) **[10 points]** Interpret the definition of an equilibrium for our nonatomic routing game. Do you see any connections with the notion of a Nash equilibrium?

(b) **[20 points]** Prove that an equilibrium always exists.

**Hint:** Define $\Phi(f) = \sum_{e \in E} h_e(f_e)$, where $h_e(x) = \int_0^x c_e(y)dy$. Think about the problem of minimizing $\Phi(\cdot)$ over all feasible flows.

(c) **[Challenging, 5 points extra credit]** Prove that if $f, f'$ are equilibrium flows, then $c_e(f_e) = c_e(f'_e)$ for all $e \in E$. Use this to conclude that $C(f) = C(f')$, i.e., all equilibrium flows have the same cost.

**Hint:** You will need to use the fact that $h_e(\cdot)$ as defined before is convex, which also makes $\Phi(\cdot)$ convex.

You can also use the fact without proof that a feasible flow is an equilibrium flow only if it minimizes $\Phi(f)$ over all feasible flows.

### 2.3 Part 2 - Efficiency of equilibria

In Part 1, we saw that not only do equilibria exist for nonatomic routing games, all equilibria also have the same cost. This suggests that $\text{PoA} = \text{PoS}$ for this class of games, and PoA can be conveniently defined as

$$\text{PoA} = \frac{C(f^E)}{C(f^*)},$$

where $f^E$ is an equilibrium flow, and $f^*$ is an optimal flow. In Part 2, we’ll study this PoA.

Consider a non-empty set $C$ of link cost functions (these functions are non-negative, non-decreasing, and continuous). We define the *Pigou bound* $\alpha(C)$ as

$$\alpha(C) = \sup_{c \in C} \sup_{r \geq 0} \frac{r \ c(r)}{x \ c(x) + (r - x) \ c(r)}$$

with the understanding that $0/0 = 1$.

It turns out that for the class of games with link cost functions in $C$,

$$\text{PoA} \leq \alpha(C).$$

In other words, if we fix a class $C$ of link cost functions, the Pigou bound is an upper bound on the PoA. You’ll prove here that this upper bound is tight. Amazingly, very simple examples of nonatomic routing games suffice to show this.
Figure 2: Simple example with 2 nodes, 2 links, and a single commodity with flow volume $r_1$. The single commodity has 2 parallel paths. The link cost functions $c_1(\cdot), c_2(\cdot) \in C$.

(a) **[15 points]** Prove that if $C$ is the class of affine cost functions (i.e., cost functions of the form $ax + b$, where $a, b \geq 0$), then $\alpha(C) = \frac{4}{3}$.

Note that this means that for nonatomic routing games with affine cost functions, the PoA $\leq \frac{4}{3}$.

(b) **[15 points]** Prove that for the game illustrated on the right panel of Fig. 1 (with the $u \rightarrow v$ link included), the PoA $= \frac{4}{3}$.

Note that this gives us an example of a game with affine cost functions that achieves the bound of part (a).

(c) **[20 points]** Consider now a general class of link cost functions $\mathcal{C}$ that includes all constant link cost functions. Restricting yourself to simple nonatomic routing games of the form described in Fig. 2, prove that the PoA can be made arbitrarily close to $\alpha(C)$.

Note that this means the bound PoA $\leq \alpha(C)$ is tight!

(d) **[Challenging, 5 points extra credit]** Can you relax the assumption that $C$ includes all constant cost functions to prove the result of part (c)?

(e) **[Challenging, 5 points extra credit]** Prove the upper bound, i.e., prove that $\text{PoA} \leq \alpha(C)$ for all games with link cost functions in $C$.

**Hint:** You might find it useful to prove the following lemma first.

A feasible flow $f$ is an equilibrium if and only if

$$\sum_{e \in E} f_e c_e(f_e) \leq \sum_{e \in E} f'_e c_e(f_e)$$

for every feasible flow $f'$.

3 **Tightness of PoA bound in load balancing games [10 points]**

In class, we looked at a load balancing game with $m$ servers and $n$ jobs. We showed that if each job can be run on every server ($S_i = S$ for all jobs $i$), and the performance objective is to minimize the maximum load, i.e., $C(A) = \max_j L_j$, then the price of anarchy (PoA) is bounded as follows.

$$\text{PoA} \leq 2 - \frac{2}{m+1}$$

Is this bound tight? If so, provide a family of examples that have a PoA that matches this bound for each $m$. If not, prove that for some $m$, no load balancing game with $m$ servers has a PoA that matches this bound.
4 PoA analysis for GSP [22 points extra credit]

Recall the ad auction model that was introduced in class. We consider an auction with \( n \) advertisers and \( n \) slots. The game here is between the \( n \) advertisers who each have their own valuation \( v_i \) for a click. The strategy for each advertiser is a bid \( b_i \in [0, \infty) \). There are \( n \) slots and based on the bids, we decide where to allocate each advertiser. In the simplest model, the \( k \)-th slot contains \( \alpha_k \) clicks, where \( \alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_n \).

The game proceeds as follows:

- Each advertiser submits a bid \( b_i \geq 0 \), which is his declared value for a click. Let \( b = (b_1, b_2, \ldots, b_n) \) denote the vector of all bids.

- The advertisers are sorted by their bids (ties are broken arbitrarily). Call \( \pi_k(b) \) the advertiser with the \( k \)-th highest bid in \( b \). For example, \( \pi_1(b) \) would denote the advertiser with the highest bid, and \( \pi_n(b) \) would denote the advertiser with the lowest bid.

- Advertiser \( \pi_k(b) \) is placed on slot \( k \) and therefore receives \( \alpha_k \) clicks.

- For each click, advertiser \( \pi_k(b) \) pays \( b_{\pi_{k+1}(b)} \), which is the next highest bid (assume that the lowest bidding advertiser pays nothing, essentially getting the last slot for free).

The vector \( \pi(b) \) is a permutation that indicates to which slot each player is assigned – it is completely determined by the vector of bids, \( b \). The utility of advertiser \( i \) when his ad is occupying slot \( j \) is given by

\[
U_i(b) = \alpha_j(v_i - b_{\pi_j+1}(b)).
\]

The “social welfare” of this game is the total value the bidders get from playing it, which is

\[
W(b) = \sum_j \alpha_j v_{\pi_j(b)}.
\]

**Price of anarchy:** We will measure the efficiency of this auction by the “price of anarchy”, which, for this game, is defined as the ratio of the optimal social welfare to the social welfare of the worst Nash equilibrium of the game. Formally, let \( b^* \) denote the vector of bids that maximizes the social welfare \( W(b) \), and let \( NE \) denote the set of all Nash equilibrium bid vectors. Then,

\[
\text{PoA} = \frac{W(b^*)}{\min_{b \in NE} W(b)}.
\]

To keep things simple, in this problem, we make the following assumptions.

- We consider only pure Nash equilibria in the analysis of the price of anarchy. This is often called the pure price of anarchy or PPOA. There are results that bound the general price of anarchy too.

- We consider the complete information setting, where each bidder knows the true valuations of all the other bidders. There are results that apply to the more realistic Bayesian setting where bidders don’t know each other’s values, but have beliefs about them.

**Your task:** We are now ready to prove some results!

(a) **[2 points extra credit]** Show that the (pure) price of anarchy can be arbitrarily large. Specifically, consider the case when \( n = 2 \). For this case, given any \( r > 1 \), construct an example for which the (pure) price of anarchy is \( r \). For a hint, read on.
(b) It turns out that ‘bad’ equilibria like the ones in (a) are unnatural, in the sense that the equilibrium includes (weakly) dominated strategies, which could expose the advertiser to a risk of obtaining a negative utility. Let us define a bid \( b_i \) for an advertiser to be \textit{conservative}, if \( b_i \leq v_i \). The strategy space for a conservative advertiser \( i \) is therefore, \([0, v_i]\).

(i) [2 points credit] Identify the dominated strategy employed in the example you provided for (a), and explain the risk involved for the corresponding advertiser. Recall that a strategy \( b_i \) is dominated if there is some \( b_i' \) such that \( U_i(b_i, b_{-i}) \leq U_i(b_i', b_{-i}) \) for all \( b_{-i} \) and for at least one value of \( b_{-i} \), the inequality is strict.

(ii) [4 points extra credit] Show that non-conservative bids are always dominated strategies.

(iii) [10 points extra credit] When \( n = 2 \), show that if both advertisers are conservative, then the PPoA is exactly 1.25.

(iv) [4 points extra credit] Show that for a general \( n \), when all \( n \) advertisers are conservative, PPoA \( \leq 2 \).

\textbf{Hint:} You may want to show that any permutation corresponding to a conservative Nash equilibrium is \textit{weakly feasible}. A permutation \( \pi \) is said to be \textit{weakly feasible} if for each pair of slots \((i, j)\) that satisfy \( i < j \) but \( \pi_i > \pi_j \), \( \frac{\alpha_j}{\alpha_i} + \frac{\beta_i}{\beta_j} \geq 1 \). After this, restrict your attention to maximizing PPoA over only weakly feasible permutations. (\textbf{Hint:} Use induction.)

5 \hspace{1cm} \textbf{Clickmaniac}

Send your team information and the email address associated with the facebook account of one group member to \texttt{cms144.caltech@gmail.com} so that we can set up your campaigns.