CS 684 Algorithmic Game Theory	**Scribe: Gei-Tai Lin** (gal24)
Instructor: Eva Tardos	**September 23, 2005**

1 Network Design with Discrete Potential Games

Recall that users chose a path P $s_i - t_i$ in a graph G. The delay incurred had to do with the number of users on every edge. $l_e(x)$ = delay on edge if x users use it. Today we will explore network design with potential functions. In talking about networks we will think in terms of the cost instead of delay. The cost of traveling across a path is additive, akin to delay in the routing problem. We can think of the cost as some sort of maintenance cost to regularly send packets.

In our network design, $l_e(x)$ is the cost of sharing that edge e for each user. Therefore the cost of a total path is $\Sigma_e l_e(x_e)$ where x_e is the number of users sharing edge e. In our network all our users will pay the same amount to cross a given edge. Because users pay the same amount, we have fair cost sharing. We define $c_e(x)$ as being the total cost for all users, and $l_e(x) = c_e(x)/x$ all int x ≥ 1

We will assume $l_e(x)$ is monotone decreasing (non-increasing), such that as we increase the number of people using an edge, the cost incurred by each user can only decrease. We will also assume that $c_e(x)$ is monotone increasing, such that the total cost cannot decrease if we add more users. See Figure 1.

2 Previous Lecture

Recall from the previous lecture that a deterministic Nash exists within our routing game. Formally:

Theorem 1 *There exists a selection of path $P_1..P_k$ that is a Nash equilibrium*

For this theorem, we considered the potential function: $\phi = \Sigma_e \Sigma_{i=1}^{x_e} l_e(i)$ In the same way, even though we now have costs on the edges instead of delay, a deterministic Nash still exists within our network design.

2.1 Comparing the quality of Nash and Opt

Now that we know that a deterministic Nash exists in our network, we want to compare the quality of the Nash to the quality of the optimal solution. We will define quality to be the total cost for these network designs.

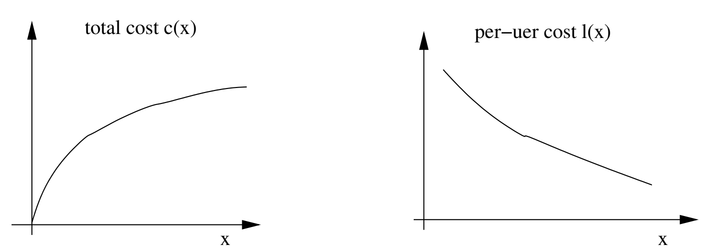

Figure 1: Graph of total cost for x users (on left), and graph of per-user cost as a function of the number of users (on right).

$$\Sigma_e c_e(x) = \Sigma_e x_e l_e(x)$$

This is our natural measure of quality. Notice that this is the same expression as total delay in our routing problem. We consider the Nash to minimizing $\phi(x)$, and will compare it to some other possible c(x) (and implicitly, opt).

We would like to argue that $C(x) \leq \phi(x)$. For an individual edge,

$$\phi_e(x_e) = l_e(1) + l_e(2) + l_e(3) + ... + l_e(x_e)$$

$$c_e(x) = x_e l_e(x_e)$$

From Figure 2 below , we can see clearly that as $l_e(x)$ monotone decreases, $\phi_e(x_e) \geq c_e(x_e)$, and summing over all these edges would give

$$C(x) \leq \phi(x).$$

What we want to claim is that the functions are approximately equal. To do this, we will have to bound $\phi(x)$

$$C(x) \leq \phi(x) \leq \alpha C(x)$$

Where α is some factor. We can turn to our fixed cost example to gain some intuition into what α is. For a fixed cost c_e the potential is

$$\phi = c_e + c_e/2 + c_e/3 + ... + c_e/x_e$$
$$= c_e(1 + 1/2 + 1/3 + 1/x_e)$$
$$\sim c_e ln(x_e)$$

To be precise α is somewhere between $[ln(x_e), ln(x_e + 1)]$

Theorem 2 *$C(x) \leq \phi(x) \leq ln(k+1)C(x)$ for all solutions x, where k is the number of users*

Proof Because l is monotone decreasing from our assumption in section 1, we saw that C $\leq \phi$. This satisfies the first inequality. For the 2nd inequality, we can compare $l_e(1)+ \ldots +l_e(e)$ to $x_e l_e(x_e)$

$$x_e l_e(x_e) = c_e(x)$$

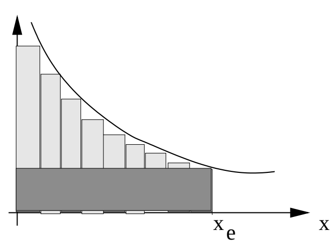

Figure 2: The total cost $x_e l_e(x_e)$ is the area of the small box. The potential function is the sum of the strips.

For any $i \leq x_e$ we have that $l_e(i) = c_e(i)/i \leq c_e(x_e)/i$, so we get

$$\phi_e(x_e) = \Sigma_{i=1}^{x_e} l_e(i) \leq c_e(x_e)\Sigma_{i=1}^{x_e} 1/i \leq c_e(x) ln(x_e + 1) \leq c_e(x_e) ln(k+1)$$

Recall that the left hand side of the inequality is the definition of ϕ, so if we take our conclusion for a single edge above and take the summation over all edges we find that

$$\phi(x) \leq ln(k+1)C(x)$$

Theorem 3 *The Nash minimizing ϕ has total cost $\leq$ ln(k+1)*Opt*

Proof We have already shown that $C \leq \phi \leq \alpha$C. This implies the solution minimizing ϕ is within α factor of minimizing C

2.2 Example of a Fixed Cost Bad Nash

Suppose we have 6 different sources $s_1 s_2 s_3 ... s_6$ that all want to reach their sink points $t_1 t_2 t_3 ... t_6$ that are located at the same point t. Each link from s_i to t_i costs 1/i respectively. Alternatively, there is a point O that leads to the sink point with cost 1+ϵ, and all the sources are linked to O with cost 0. In this case, Opt is obviously to have everyone go on the 1+ϵ link, and the cost per user would be (1+ϵ)/6. However, user 6 isn't happy, because (1+ϵ)/6 is greater than what he could've gotten on his own, which was 1/6. So he switches. This causes a cascading effect whereby all the users switch back to their primary link to the sink and we get a Nash costing 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6. We could increase the number of users in the construction and choose to achieve a bad Nash with exactly the cost we came up with in our proof above.

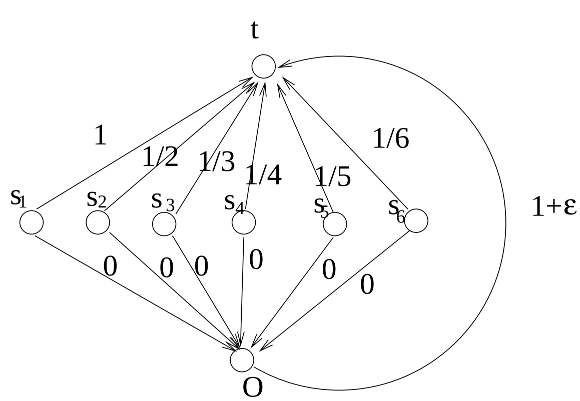

Figure 3: Example with worth case Nash/Opt ratio.