Ubinet, Distributed Optimization and Games 2015-2016	December 14, 2016

Lecture 3

Lecturer: Giovanni Neglia — *Scribe: Arthur Finkelstein, Remy Garcia*

NOTE: The content of these notes has not been formally reviewed by the lecturer. It is recommended that they are read critically.

1 Introduction

In the previous lesson, we looked into the example of an electrical network and showed that a simple local policy for an agent, in our case an electron, can lead to the optimization of a large distributed problem which is the minimization of energy loss in the network. In this lesson, we consider ourselves with a road traffic model to demonstrate that choosing local policies is not trivial and how potential wrong choices can lead to poor system behaviour. Moreover we solve the more general problem which arises from our road network and present how local policies in such cases should be correctly chosen.

2 Problem Definition

We consider an oriented graph (e.g. the road-traffic graph of a city) like in Figure 1. From now on we look at this problem as a routing problem.

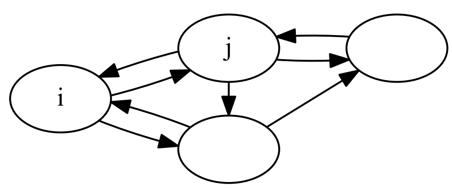

Figure 1: Oriented graph of a city

We have a set of directed links E which are the connections between the nodes of the graph and a set of *source-destination* pairs S, where we have $s \in S$ such that $s = (i, j)$ represents the agents wanting to travel from node i to j. Moreover we denote by R the set of all routes.

For each $s \in S$ we define $R(s)$ to be the set of routes connecting the *source-destination* pair and f_s to be the amount of traffic to route on s. Also $s(r)$ is the *source-destination* pair connected by $r \in R$. Since a given f_s can be split over various routes, we denote by x_r the amount of traffic of $s(r)$ that takes route r i.e. $\sum_{r \in R(s)} x_r = f_s \ \forall s$ with $x_r \geq 0$.

Finally, we define y_l to be the amount of traffic on link $l \in E$ over all routes, i.e. $y_l = \sum_{r|l \in r} x_r$. On each link l there exists a delay per unit of traffic which is a function of the traffic on that link that we denote by $D_l(y_l)$. The delay function is assumed to be convex, increasing and its first derivative to exist.

Thus the problem consists of minimizing the overall delay experienced by the agents.

The optimization problem can be modelled as follows:

$$\begin{aligned}
\text{minimize} \quad & \sum_{l \in E} y_l D_l(y_l) \\
\text{subject to} \quad & \sum_{r \in R(s)} x_r = f_s \ \forall s \in S \\
& y_l = \sum_{r | l \in r} x_r \ \forall l \in E \\
& x_r \geq 0 \ \forall r \in R
\end{aligned} \tag{1}$$

This problem happens to be convex.

Proof: We know that D_l is increasing and convex. This means that for all y_l we have $D_l'(y_l) \geq 0$ and $D_l''(y_l) \geq 0$.

We know that $y_l D_l(y_l)$ is convex iff its second derivative is nonnegative.

We start by computing the first derivative. $(y_l D_l(y_l))' = D_l(y_l) + y_l D_l'(y_l)$ where $D_l(y_l) \geq 0$ and $y_l D_l'(y_l) \geq 0$. Thus $(y_l D_l(y_l))'$ is nonnegative.

Then we compute the second derivative. $(y_l D_l(y_l))'' = D_l'(y_l) + y_l D_l''(y_l) + D_l'(y_l) = 2D_l'(y_l) + y_l D_l''(y_l)$ where $2D_l'(y_l) \geq 0$ and $y_l D_l''(y_l) \geq 0$. Thus $(y_l D_l(y_l))''$ is nonnegative.

Hence $y_l D_l(y_l)$ is convex and it implies that $\sum_l y_l D_l(y_l)$ is also convex and thus this problem is convex. $\square$

3 Naive solution

One naive solution for this problem would be that each person tries to minimize his own delay, and from this we could hope that the total delay would be minimized. Therefore we are going to check if this solution is indeed the optimal one and if it isn't how far off are we from the optimal.

3.1 A Specific exemple

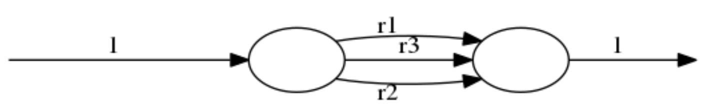

Figure 2: Our specific network

With this specific exemple each different edge has a specific delay equal to $D_i = \alpha_i \times y_i$.

From this function we can minimize the total delay which gives us $\sum_{i=1}^{3} (\alpha_i \times y_i) \times y_i = \sum_{i=1}^{3} \alpha_i \times y_i^2$, and we can see that this function is the same as the model we studied in the first lecture.

We can compare our network to an electrical network with three different resistances in parallel, and from our previous lecture we know that the difference of potential $\Delta V = I_1 R_1 = I_2 R_2 = I_3 R_3$ with $I_i = y_i$ and $R_i = \alpha_i$, so at the optimal solution y_i^* we have then $\alpha_1 \times y_1^* = \alpha_2 \times y_2^* = \alpha_3 \times y_3^*$ which is $D_1(y_1^*) = D_2(y_2^*) = D_3(y_3^*)$.

This result can be easily understood with the fact that each car at the beginning will take the path with the least delay and once this delay becomes bigger than the delay of either of the other path we will take this path, and so on, this will create a situation of equilibrium in regard to the delays of each route.

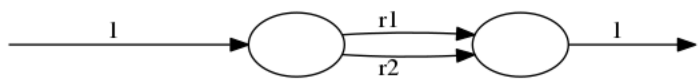

Figure 3: Our general network

3.2 A general exemple

We are now going to extend the previous exemple with delay functions that aren't linear, we have then $D_1(y_1) = 1$ and $D_2(y_2) = y_2^n$, because this function relates the most to how delay affects routing in a real world scenario. This exemple is known as the PIGOU exemple.

The comparison between the two delay function gives us the following graphic.

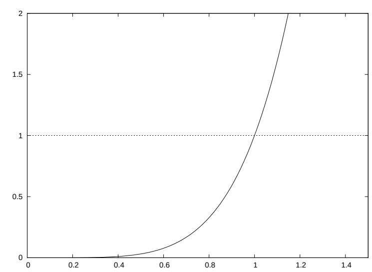

Figure 4: Our general network

We now put our naive solution in this example with each person minimizing his or her delay. We get $y_1^1 = 0$ and $y_2^1 = 1$ because the second link is always the most attractive with regard to the delay, $D_t(1) = 1 \times 0 + 1^n \times 1 = 1$.

We now try to solve the problem in the general case where $D_t(\alpha) = (1 - \alpha) \times 1 \times \alpha \times \alpha^n$. We know that to find the minimum we look for:

$$D_t'(\alpha) = 0 \iff -1 + (n+1) \times \alpha^n \iff \alpha^* = \frac{1}{\sqrt[n]{n+1}}$$

Thus we know that the optimal quantity of traffic is α^*. Now we will check that the delay we got from the naive solution is optimal or not.

The optimal delay is:

$$D_t(\alpha^*) = 1 - \alpha^* + {\alpha^*}^{n+1} = 1 - \frac{1}{\sqrt[n]{n+1}} + \frac{1}{\sqrt[n]{n+1}} * \frac{1}{n+1} = 1 - \frac{1}{\sqrt[n]{n+1}} * \frac{n}{n+1}$$

Thus we can deduce that for every $n > 0$ the delay is strictly less than 1. Therefore the naive solution is false but if this solution is close enough from the optimal one it would still be acceptable.

For this we compute the price of anarchy:

$$\frac{D_t(\alpha^1)}{D_t(\alpha^*)} = \frac{1}{1 - \frac{1}{\sqrt[n]{n+1}} * \frac{n}{n+1}} \to +\infty$$

Thus the naive solution gives us a delay that is arbitrarily bigger than the optimal one so our solution is unfortunately pretty poor.

The next step would be to apply our reasoning to the original problem.

4 Solving Problem (1) by the Lagrange multipliers method

We can observe that the objective function of our optimization problem (1) is convex. Indeed:

$$f(x) = y_l D(y_l) \Rightarrow f'(x) = y_l D'(y_l) + D(y_l) \Rightarrow f''(x) = y_l D''(y_l) + D'(y_l) + D'(y_l) \tag{2}$$

We know that $D'(y_l)$ and $D''(y_l)$ are nonnegative, on the hypothesis that $D(y_l)$ is convex. Thus each term of $f''(x)$ is nonnegative and the sum of nonnegative terms is nonnegative, hence the objective function is convex.
Another observation that we make is that each constraint is linear and that they define a set which is compact. Indeed, each variable x_r is bounded:

$$0 \leq x_r \leq \max_{s \in S} f_s \quad \forall r \in R \tag{3}$$

so the set defined is closed and bounded.

Considering the previous observations we can apply the *lagrange multipliers method* while using theorem (1) from the lesson 2. We define two set of multipliers:

1. one for each source-destination pair, λ_s, $\forall s \in S$ and
2. one for each link, ν_l, $\forall l \in E$.

We can claim that $\mathbf{z}^* = \binom{\mathbf{x}^*}{\mathbf{y}^*}$ is a global minimum for this optimization problem if and only if there exist λ_s^*, $\forall s \in S$ and ν_l^*, $\forall l \in E$ such that:

1. $\mathbf{x}^*$ and $\mathbf{y}^*$ are feasible,
2. $\nabla_{\mathbf{x}} L(\mathbf{z}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)^T (\mathbf{z} - \mathbf{z}^*) \geq 0, \;\; \forall \mathbf{z} \in \{\binom{\mathbf{x}}{\mathbf{y}}, \; \mathbf{x} \geq 0\}$.

The Lagrangian function results:

$$L(z, \lambda, \nu) = \sum_{l \in E} y_l D(y_l) + \sum_{s \in S} \lambda_s \left(f_s - \sum_{r : s(r) = s} x_r \right) + \sum_{l \in E} \nu_l \left(\sum_{r : l \in r} x_r - y_l \right) \tag{4}$$

When we differentiate it, we obtain:

$$\frac{\partial L}{\partial y_{\bar{l}}} = D(y_{\bar{l}}) + y_{\bar{l}} D'(y_{\bar{l}}) - \nu_{\bar{l}} \qquad\qquad \frac{\partial L}{\partial x_{\bar{r}}} = -\lambda_{s(\bar{r})} + \sum_{l \in \bar{r}} \nu_l \tag{5}$$

For y_l:
We can take a point "above" or "below" and we get:

$$\frac{\delta L}{\delta y_l} \Delta y_l \geq 0$$

$$\frac{\delta L}{\delta y_l} - \Delta y_l \geq 0$$

from these two results we can see that $\frac{\delta L}{\delta y_l}$ is equal to zero.
For x_r:
We can take a point to the "right" but not on the "left" because of the domain, and we get:

$$\frac{\delta L}{\delta x_r} \geq 0$$

if there is traffic on that route $x_r = 0$ and if there is no traffic then $x_r \geq 0$. At the optimum, we have:

$$\lambda_s^*(r) \begin{cases} = \sum_{l \in r} \nu_l^* & \text{if } x_r > 0, \\ \leq \sum_{l \in r} \nu_l^* & \text{if } x_r = 0, \end{cases} \tag{6}$$

and the quantity ν_l results:

$$\nu_l^* = D(y_l) + y_l D'(y_l). \tag{7}$$

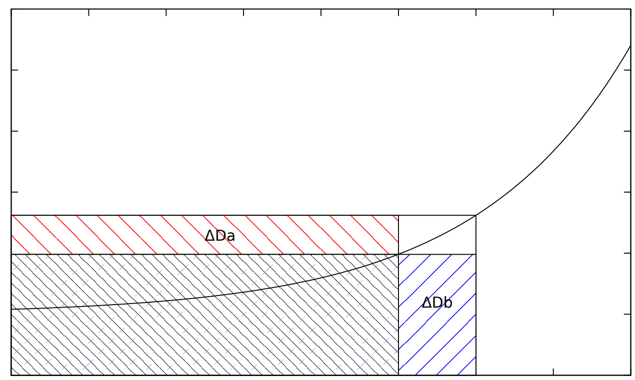

Figure 5: Additional term representation

We can interpret the result for the multiplier ν_l like the cost that the drivers traversing link l must pay. That cost is composed of the delay (i.e. $D(y_l)$) and the additional term $y_l D'(y_l)$. The additional term is useful when there is a change in traffic and represent the cost we put on a route to prevent agents from using it.

In Figure 5 we can see the total delay before Δy arrives on the black pattern. When computing the new delay, we need to consider the red pattern, ΔD_a and the blue pattern, ΔD_b. Where $\Delta D_a = y_l' D(y_l')\Delta y$ and $\Delta D_b = \Delta y D(y_l')$. Then we have $\Delta D = \Delta y(D(y_l') + y_l' D'(y_l'))$, the weight of the link. Furthermore, we can see $\lambda_{s(r)}$ the minimal cost available to source-destination pair $s(r)$.

With the previous observation, if we add tolls on the link, the drivers are encouraged to have a more desirable behaviour. Indeed we keep the example in the previous section and we add the toll $y_l D'(y_l)$ at the cost of each link. Each user experiences then two new costs for traveling the two paths. $C_{upper} = 1 + y_{upper} D'(y_{upper}) = 1$ and $C_{lower} = y_{lower} + y_{lower} D'(y_l) = 2y_{lower}$. Now the drivers choose the link where the total cost, delay plus toll, is minimum and the final solution is that half of the drivers choose the lower link and the others choose the upper link, that is the optimal solution for Problem (1).

References

[1] Frank Kelly and Elena Yudovina, Stochastic Networks. *Cambridge Press*, 2014.