Network 20q HW7

ZHANG Xiao Research Intern in IBM CRL

September 28, 2017

1 Time-Dependent Pricing

In this question, we need to calculate the optimal reward for the data delay transmission.

The proportions of delay transmission data is:

$$w_A(p) = 1 - e^{-\frac{p}{4}} \tag{1}$$

$$w_B(p) = 1 - e^{-\frac{p}{2}} \tag{2}$$

So the volume of transmitted data for A is:

$$v_{A,day} * w_A(p) = 8 * (1 - e^{-\frac{p}{4}}) \tag{3}$$

The volume of transmitted data for B is:

$$v_{B,day} * w_B(p) = 6 * (1 - e^{-\frac{p}{2}}) \tag{4}$$

The total cost for the reward is:

$$C_{reward} = p * (8 * (1 - e^{-\frac{p}{4}}) + 6 * (1 - e^{-\frac{p}{2}})) \tag{5}$$

At the same time, the exceeding costs for day time and night time are:

$$C_{night} = max\{5 + 8 * (1 - e^{-\frac{p}{4}}) + 6 * (1 - e^{-\frac{p}{2}}) - 10, 0\} \tag{6}$$

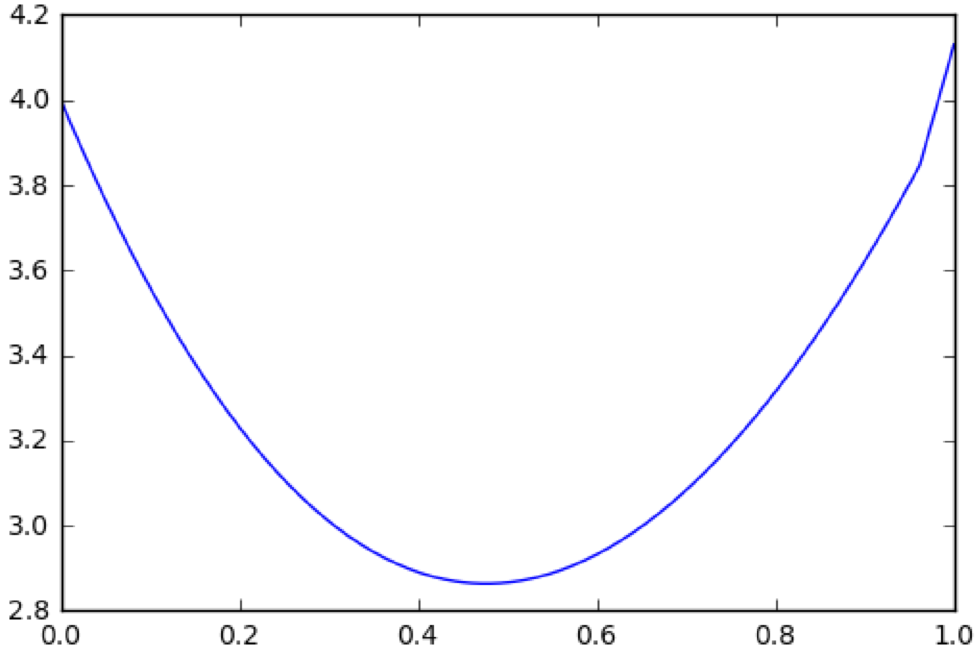
$$C_{day} = max\{4 - 8 * (1 - e^{-\frac{p}{4}}) - 6 * (1 - e^{-\frac{p}{2}}), 0\} \tag{7}$$

So the object function is:

$$C_{reward} + C_{day} + C_{night} \tag{8}$$

We need to extract the optimal p^*, to make the object function minimal.

With the source code in python, we could get the plot of cost v.s. the reward p, from the graph, we could conclude that, the function is a convex function.



And we could get the numerical solution by differentiate the cost function, by finding the first positive element, we could get the p^*.

$$p^* = 0.47614761476147616 \tag{9}$$

2 RIP

2.1 a

At the beginning, $t = 0$, each node could only see itself

Node	Destination	Cost	Next
A	A	0	A
B	B	0	B
C	C	0	C
D	D	0	D

(10)

When $t = 1$:

Node	Destination	Cost	Next
A	C	2	A
B	C	1	B
B	D	6	B
C	D	3	C
C	A	2	C
C	B	1	C
D	B	6	C
D	C	3	C

(11)

When $t = 2$:

Node	Destination	Cost	Next
A	B	3	C
A	C	2	A
A	D	5	C
B	A	3	C
B	C	1	B
B	D	4	C
C	A	2	C
C	B	1	C
C	D	3	C
D	A	5	C
D	B	4	C
D	C	3	C

(12)

2.2 b

While the link A to C failed, and B, C update the table, C would clean current path, and update its route with the link with B ,we could get:

Node	Destination	Cost	Next
B	A	3	C
C	A	4	B
D	A	5	C

(13)

Then B and D have to update

Node	Destination	Cost	Next
B	A	5	C
C	A	4	B
D	A	7	C

(14)

Then C would update again with C to B

Node	Destination	Cost	Next
B	A	5	C
C	A	6	B
D	A	7	C

(15)

The weight of path would increase to ∞

2.3 c

We could adopt the norms of many routing protocols that forbid the update coming back from last hop, which means do not allow update information coming back from a newly updated one.