Convex Optimization II

Lecture 10: TCP Forward Engineering Optimization-based Congestion Control

Hamed Shah-Mansouri

Department of Electrical Engineering

Sharif University of Technology

1400-2

References

[1] BA. Forouzan, *TCP/IP Protocol Suite*. McGraw-Hill Inc., 4th edition, Jun, 2002.

[2] S.H. Low and D.E. Lapsley, “Optimization flow control, I: Basic algorithm and convergence,” *IEEE/ACM Trans on Networking*, vol. 7, no. 6, pp. 861-874, Dec. 1999.

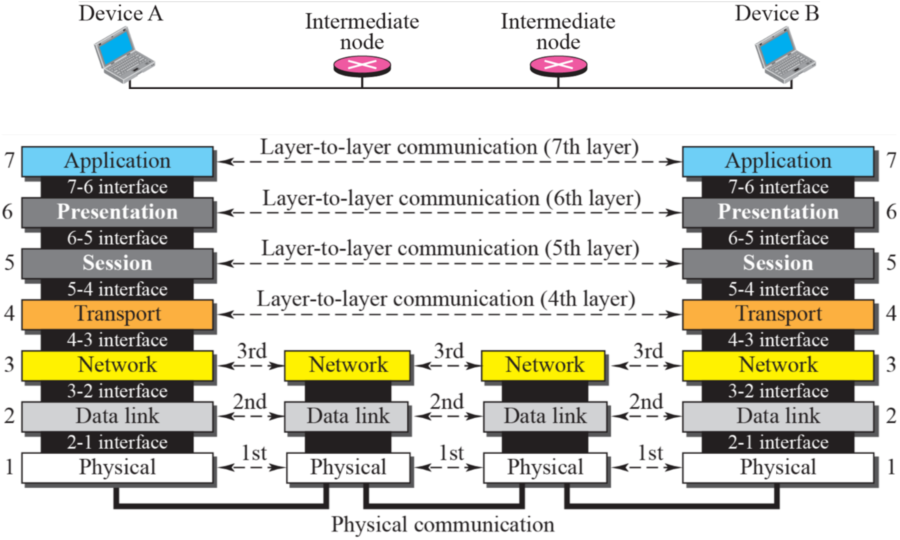
[3] F.P. Kelly, A. Maulloo, and D. Tan, “Rate control for communication networks: Shadow prices, proportional fairness, and stability,” *Journal of Operations Research Society*, vol. 49, no. 3, pp. 237-252, March 1998.

[4] S.H. Low, F. Paganini, and J. Doyle, “Internet congestion control,” *IEEE Control Systems Magazine*, Feb. 2002.

Motivation

- The work on forward engineering of transmission control protocol (TCP) and active queue management (AQM) in [4] has shown that most of the existing heuristic TCP protocols can be viewed as algorithms to solve network utility maximization problem for some choices of utility functions.
- It would be interesting if we could come up with new TCP protocols that solve network utility maximization problem for good choices of utility functions.
- The objective is to design distributed algorithms in which each source can perform computation to determine its sending rate based on some feedback (e.g., path delay, packet loss probability) from the network.
- How can we use convex optimization to address the above problem?
- In this lecture, we will formulate and solve a network utility maximization problem (with notations from [2]).
- Thanks to Prof. Behrouz A. Forouzan and Prof. Vincent Wong for the slides.

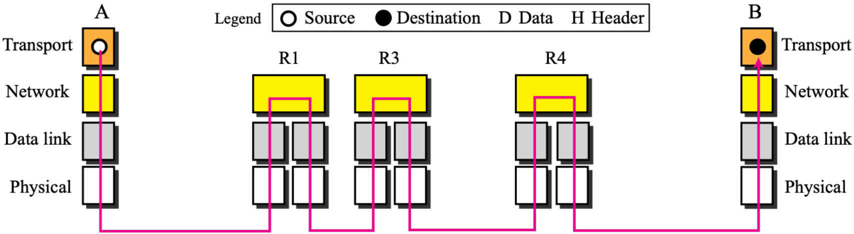
Network OSI Model

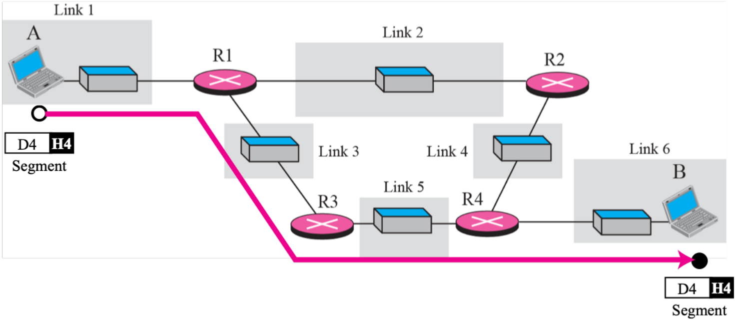


[1] BA. Forouzan, *TCP/IP Protocol Suite*. McGraw-Hill Inc., 4th edition, Jun, 2002.

Transport Layer

End–to-end Communication

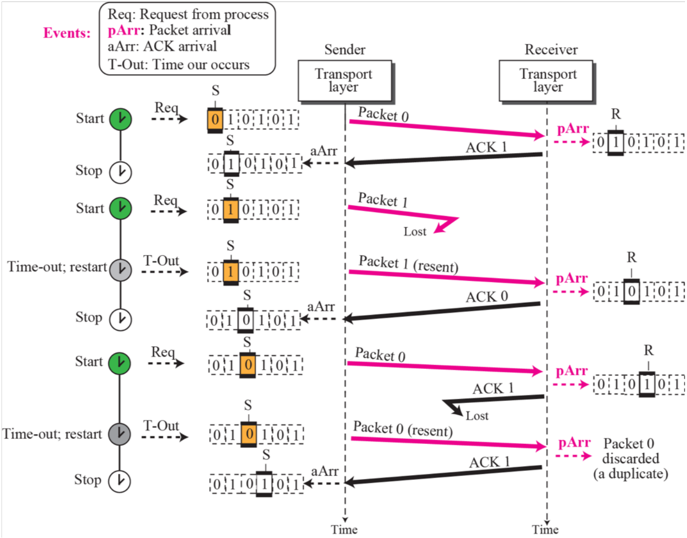




[1] BA. Forouzan, *TCP/IP Protocol Suite*. McGraw-Hill Inc., 4th edition, Jun, 2002.

Transmission Control Protocol (TCP)

Congestion Control



[1] BA. Forouzan, *TCP/IP Protocol Suite*. McGraw-Hill Inc., 4th edition, Jun, 2002.

Problem Formulation

- Consider a network that consists of a set $\mathcal{L} = \{1, \ldots, L\}$ of unidirectional links of capacity c_l, for each $l \in \mathcal{L}$.

- The network is shared by a set $\mathcal{S} = \{1, \ldots, S\}$ of sources.

- The path $\mathcal{L}(s) \subseteq \mathcal{L}$ is a set of links that source s uses along its routing path.

- For each link l, we define $\mathcal{S}(l) = \{s \in \mathcal{S} \mid l \in \mathcal{L}(s)\}$ as the set of sources that use link l.

- Note that link $l \in \mathcal{L}(s)$ if and only if source $s \in \mathcal{S}(l)$.

- Source s attains a utility $U_s(x_s)$ when it transmits at rate x_s that satisfies $m_s \leq x_s \leq M_s$.

- Utility function U_s is assumed to be increasing and strictly concave.

Problem Formulation

- Our objective is to choose source rates $\mathbf{x} = (x_s, s \in \mathcal{S})$ so as to solve the following network utility maximization problem

$$\begin{array}{ll} \underset{x_s \in I_s,\ s \in \mathcal{S}}{\text{maximize}} & \displaystyle\sum_{s \in \mathcal{S}} U_s(x_s) \\ \text{subject to} & \displaystyle\sum_{s \in \mathcal{S}(l)} x_s \leq c_l, \quad l = 1, \ldots, L. \end{array} \tag{1}$$

  For source s, the range or interval $I_s = [m_s, M_s]$.

- Problem (1) is called the primal problem.
- Constraint in problem (1) says that the aggregate source rate at any link l cannot exceed the capacity.
- Problem (1) has a *unique* optimal solution. Q: Why?

Problem Formulation

- The formulated network utility maximization problem is a *convex* optimization problem with strictly concave objective and linear inequality constraints.
- What is the difficulty of solving problem (1)?
- In networks, we would like to solve problem (1) in a distributed fashion.
  - The objective function is separable in x_s
  - The source rates x_s are coupled by the constraint in problem (1).
- Q: Any suggestion on how we can tackle this problem?

Lagrangian and Dual Function

- Remember from the lecture on convex optimization, sometimes it might be easier to solve the dual problem instead of the original primal problem.

- The Lagrangian of problem (1) is

$$
\begin{aligned}
L(\mathbf{x}, \mathbf{p}) &= \sum_{s \in \mathcal{S}} U_s(x_s) - \sum_{l \in \mathcal{L}} p_l \left(\sum_{s \in \mathcal{S}(l)} x_s - c_l \right) \\
&= \sum_{s \in \mathcal{S}} \left(U_s(x_s) - x_s \sum_{l \in \mathcal{L}(s)} p_l \right) + \sum_{l \in \mathcal{L}} p_l c_l
\end{aligned}
$$

  where $\mathbf{p} = (p_1, \ldots, p_L)$.

- From the lecture on convex optimization, we can write the objective function of the dual problem (i.e., dual function $D(\mathbf{p})$) as

$$
\underset{x_s \in I_s,\ s \in \mathcal{S}}{\text{maximize}} \ L(\mathbf{x}, \mathbf{p}).
$$

  This requires maximizing the Lagrangian over $x_s \in I_s$.

Decompose $D(\mathbf{p})$ into S Separable Subproblems

- Since the first term is separable in x_s, we have (with abuse of notations)

$$\max_{x_s \in I_s,\ s \in \mathcal{S}} \sum_{s \in \mathcal{S}} \left(U_s(x_s) - x_s \sum_{l \in \mathcal{L}(s)} p_l \right) = \sum_{s \in \mathcal{S}} \max_{x_s \in I_s} \left(U_s(x_s) - x_s \sum_{l \in \mathcal{L}(s)} p_l \right)$$

- Therefore, the task of maximizing the Lagrangian over $(x_1, x_2, \ldots, x_S)$ can be reduced to several tasks of maximizing some objective functions over x_s for each source s.

Dual Function and Dual Problem

- In particular, we can write the dual function as

$$D(\mathbf{p}) = \underset{x_s \in I_s,\ s \in \mathcal{S}}{\text{maximize}}\ L(\mathbf{x}, \mathbf{p}) = \sum_{s \in \mathcal{S}} B_s(p^s) + \sum_{l \in \mathcal{L}} p_l c_l \tag{2}$$

where

$$B_s(p^s) = \underset{x_s \in I_s}{\text{maximize}}\ U_s(x_s) - x_s p^s \tag{3}$$

$$p^s = \sum_{l \in \mathcal{L}(s)} p_l \tag{4}$$

- The first term of the dual objective function $D(\mathbf{p})$ is decomposed into S separable subproblems in form of (3).

- Question: What are (3) and (4)?

- The dual problem is

$$\begin{array}{ll} \text{minimize} & D(\mathbf{p}) \\ \text{subject to} & \mathbf{p} \succeq 0 \end{array} \tag{5}$$

Interpretation of Lagrange Multipliers $\mathbf{p}$

- Let p_l denote the congestion price per unit bandwidth (indicating congestion measure) of link l.
- Q: What is the interpretation of p^s?
- Q: Can source s solve $B_s(p^s)$ locally?

Dual Problem

- So, it seems that we actually *can* solve the dual problem in a distributed fashion (Recall that we *cannot* solve the primal problem in a distributed manner).
- Q: Why is it good enough to solve the dual problem?
- Q: Does strong duality hold in this problem?
  - Problem (1) is a strictly concave optimization problem.
  - Slater's condition and complementary slackness are satisfied.
- Since strong duality holds, we can solve problem (5) instead of problem (1).

Dual Problem

- There is still one more question, solving the dual problem will give us only the optimal price values. How can we obtain the optimal data rates?
- Let $\mathbf{p}^*$ denote the optimal solution of problem (5), then $\mathbf{x}^*$ would be simply obtained for each individual source s by solving local problem (3).
- As long as we can obtain $\mathbf{p}^*$ in a distributed fashion, we are done with distributive solving of problem (1).
- So, let us continue on solving the dual problem.

Solving Dual Problem

- To be able to solve problem (5), we first need to obtain $B_s(p^s)$.
- We define $x_s(p^s)$ as the unique maximizer in (3).

$$x_s(p^s) = \arg\max_{x_s \in I_s} \; U_s(x_s) - x_s p^s$$

$$x_s(p^s) = \left[U_s'^{-1}(p^s)\right]_{m_s}^{M_s} \tag{6}$$

- In that case, we have

$$B_s(p^s) = U_s(x_s(p^s)) - x_s(p^s)p^s.$$

- Since we follow the notations from [2], we abuse the notation and use $x_s(.)$ both as a function of scalar price $p \in \mathcal{R}_+$ and of vector price $\mathbf{p} \in \mathcal{R}_+^L$. That is,

$$x_s(\mathbf{p}) = x_s(p^s) = x_s\left(\sum_{l \in \mathcal{L}(s)} p_l\right).$$

Gradient Projection Algorithm

- It is clear that problem (5) is a convex minimization problem over non-negative orthant.

- If it is easy, we can solve problem (5) by finding the closed form solution of the KKT conditions.

- An alternative is to solve problem (5) iteratively, using gradient projection method:

$$p_l(t+1) = \left[p_l(t) - \gamma \frac{\partial D}{\partial p_l}(\mathbf{p}(t))\right]^+, \tag{7}$$

  where γ is the step size.

- From the objective function of problem (5), we have

$$\begin{aligned} D(\mathbf{p}) &= \sum_{s\in\mathcal{S}} \left(U_s(x_s(\mathbf{p})) - x_s(\mathbf{p})p^s\right) + \sum_{l\in\mathcal{L}} p_l c_l \\ &= \sum_{s\in\mathcal{S}} U_s(x_s(\mathbf{p}) - \sum_{l\in\mathcal{L}} p_l \sum_{s\in\mathcal{S}(l)} x_s(\mathbf{p}) + \sum_{l\in\mathcal{L}} p_l c_l \end{aligned}$$

GRADIENT PROJECTION ALGORITHM (CONT.)

- We can represent the gradient of the objective function $D(\mathbf{p})$ as

$$\nabla D(\mathbf{p}) = \left[\frac{\partial D}{\partial p_l}(\mathbf{p})\right]_l$$

where

$$\frac{\partial D}{\partial p_l}(\mathbf{p}) = -(x^l(\mathbf{p}) - c_l) \tag{8}$$

and

$$x^l(\mathbf{p}) = \sum_{s \in \mathcal{S}(l)} x_s(\mathbf{p})$$

denotes the aggregate data rate at link l.

- From (8), we update the congestion prices as:

$$p_l(t+1) = \left[p_l(t) + \gamma(x^l(\mathbf{p}(t)) - c_l)\right]^+, \qquad l \in \mathcal{L}. \tag{9}$$

REMARKS

- **Remark 1**: If the update equations in (9) converge, then vector $\mathbf{p}$ will converge to the optimal solution of problem (5).

- **Remark 2**: The update equation in (9) is consistent with the law of supply and demand :
  - If the demand $x^l(\mathbf{p}(t)) = \sum_{s \in \mathcal{S}(l)} x_s(\mathbf{p}(t))$ for bandwidth at link l exceeds the supply c_l, raise price $p_l(t)$.
  - Otherwise, reduce price.

- **Remark 3**: The update equation in (9) is completely distributed.

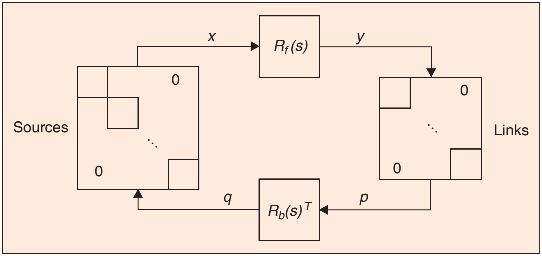
DISCUSSION

- Recall that due to strong duality, we could solve problem (5) instead of problem (1).

- Looking at update equations in (6) and (9), can we say that they let the network links l and the sources s act as processors in a distributed computation system to solve problem (5)?

Overall Structure

- The overall structure of the congestion control system:



- In each iteration, each source s individually solves (3) and communicates its result $x_s(p)$ to all links $l \in \mathcal{L}(s)$ on its path. Question: How do sources communicate with the links?

- Links l then update their prices p_l according to (9) and communicate the new prices to sources s, and the cycle repeats. Question: How do links communicate with the sources?

Link l's Algorithm

- Link l's algorithm:

At times $t = 1, 2, \ldots$, link l:
1. Receives rates $x_s(t)$ from all sources $s \in S(l)$ that go through link l.
2. Computes a new price

$$p_l(t+1) = [p_l(t) + \gamma(x^l(t) - c_l)]^+$$

where $x^l(t) = \sum_{s \in S(l)} x_s(t)$.
3. Communicates new price $p_l(t+1)$ to all sources $s \in S(l)$ that use link l.

- The link algorithm can be implemented as an active queue management (AQM) protocol.

SOURCE s'S ALGORITHM

- Source s's algorithm:

At times $t = 1, 2, \ldots$, source s:
1. Receives from the network the sum $p^s(t) = \sum_{l \in L(s)} p_l(t)$ of link prices in its path.
2. Chooses a new transmission rate $x_s(t+1)$ for the next period: [1]

$$x_s(t+1) = x_s(p^s(t))$$

where $x_s(\cdot)$ is given by (6).
3. Communicates new rate $x_s(t+1)$ to links $l \in L(s)$ in its path.

- The source algorithm can be implemented as a TCP protocol.