Application in Network Optimization

Danny H.K.Tsang

Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology
Email:eetsang@ust.hk

Lecture Outline

- In networking, it is desirable to devise some techniques to help in the systematic design of systems
- Mathematical optimization provides such tools
- Goal: Given a set of variables, and a measure of effectiveness, how to find the most effective solution – the combination of system variables that leads to the best possible effectiveness.

Topic

1. Network Flow Problem (Optimal Routing)
2. Bandwidth Sharing in a Network

Network Flow Problem (Optimal Routing)

- Minimize a *linear cost* subject to *linear constraints*
- Let a network be represented as a directed graph $G(N, L)$, where N is the set of nodes (routers) and L is the set of directed links;
- Any link $l \in L$ has a head node $h(l)$ and a tail node $t(l)$, link is directed from the head to the tail;
- There are K demands that are to be routed on this network
  - Each demand is associated with an ordered pair of nodes (n_1, n_2);
  - Note that n_1 is the source of demand and n_2 is the destination;
  - Demands are numbered $1, 2, ..., k, ..., K$, nodes are numbered $1, 2, ..., i, ..., N$, links are numbered $1, 2, ..., l, ..., L$.

Network Flow Problem (Optimal Routing)

- Assume that traffic corresponding to a demand can be split arbitrarily across multiple paths between the source and destination. With arbitrary splitting allowed, it is possible that every link in L carries some part of a demand $d(k)$, $1 \le k \le K$.
- Define a *flow vector* $\mathbf{x}(k)$ corresponding to the kth demand, with $x(k)_l$ being the amount of the kth demand carried on link l.

Basic Definition

- The topology of a network can be summarized using its *node-link incidence matrix* $\mathbf{A}$;
- Note that A is a $N \times L$ incidence matrix, with a row for each node and a column for each link;
- Let $\mathbf{A}_{i,l}$ represent the (i,l)th element of $\mathbf{A}$, $\mathbf{A}_{i,.}$ represent the ith row of $\mathbf{A}$ and $\mathbf{A}_{.,l}$ represent the lth column of $\mathbf{A}$;
- Then the column corresponding to link l has the following entries:

$$\mathbf{A}_{i,l} = \begin{cases} +1, \text{ if } i \text{ is the head node of link } l \\ -1, \text{ if } i \text{ is the tail node of link } l \\ 0, \text{ otherwise} \end{cases}$$

- Let $s(k)$ and $t(k)$ be the source and destination nodes of demand k;
- Flow conservation equations are as follows:

$$\mathbf{A}_{i,.}\mathbf{x}(k) = \begin{cases} d(k), \text{ if } i = s(k) \\ -d(k), \text{ if } i = t(k) \\ 0, \quad \text{otherwise} \end{cases}$$

Basic Definition (Con't)

- If we now consider all the rows of $\mathbf{A}$ together, then we have the following compact equation:

$$\mathbf{A}\mathbf{x}(k) = \mathbf{v}(k)$$

where $\mathbf{v}(k)$ is an $N \times 1$ vector with the following entries:

$$\mathbf{v}(k)_i = \begin{cases} d(k), \text{ if } i = s(k) \\ -d(k), \text{ if } i = t(k) \\ 0, \quad \text{otherwise} \end{cases}$$

- We can see $\mathbf{v}(k)$ is a vector specifying the amount of net outgoing demand k from each node in the network.

Basic Definition (Con't)

- Now consider the matrix

$$\mathbb{A} = \begin{bmatrix} \mathbf{A} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{A} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & \mathbf{A} \end{bmatrix}$$

- There are K block elements in each row and K block elements in each column. $\mathbf{A}$ is the familiar node-like incidence matrix with dimension $N \times L$. 0 is also a matrix of dimension $N \times L$. Hence, $\mathbb{A}$ is a matrix of dimension $KN \times KL$.

Problem Formulation

- With these definitions, consider the equation

$$\begin{bmatrix} \mathbf{A} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{A} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \vdots \\ \mathbf{x}(K) \end{bmatrix} = \begin{bmatrix} \mathbf{v}(1) \\ \mathbf{v}(2) \\ \vdots \\ \mathbf{v}(K) \end{bmatrix}$$

- The above equation is the compact flow conservation equation we were looking for. Clearly, it is nothing but K equations of the form $\mathbf{Ax}(k) = \mathbf{v}(k)$, with $1 \leq k \leq K$.

Capacity Constraints

- For a feasible routing, the sum of all flows on a link should stay below the link capacity. Suppose that $\mathbf{C}$ denotes the column vector of link capacities, with C_l being the capacity of link l. Then, for a feasible routing, we have

$$\mathbf{x}(1) + \mathbf{x}(2) + ... + \mathbf{x}(K) \leq \mathbf{C}$$

Capacity Constraints (Con't)

- The vector of spare capacities, denoted by $\mathbf{z}$, is given by

$$\mathbf{z} = \mathbf{C} - (\mathbf{x}(1) + \mathbf{x}(2) + ... + \mathbf{x}(K))$$

- Let $z := min_{l \in L} z_l$ be the *smallest* spare capacity corresponding to a given feasible routing. Then the following inequality holds:

$$\mathbf{x}(1) + \mathbf{x}(2) + ... + \mathbf{x}(K) \leq \mathbf{C} - z\mathbf{1}$$

where $\mathbf{1}$ is a column vector of L elements, all of which a 1.

Capacity Constraints (Con't)

- Let $\mathbf{I}$ be the $L \times L$ identity matrix and $\mathbb{I}$ be defined as

$$\mathbb{I} = [\ \mathbf{I} \quad \mathbf{I} \quad \ldots \quad \mathbf{I}\ ]$$

- There are K block elements in the matrix $\mathbb{I}$. Thus, the dimension of $\mathbb{I}$ is $L \times KL$. Then we have

$$[\ \mathbf{I} \quad \mathbf{I} \quad \ldots \quad \mathbf{I}\ ] \begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \vdots \\ \mathbf{x}(K) \end{bmatrix} + z\mathbf{1} \leq \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_L \end{bmatrix}$$

- As expected, the product of the $L \times KL$ matrix $\mathbb{I}$ and the $KL \times 1$ vector representing the flow vectors of all demands gives the $L \times 1$ vector of link capacities.

Objective Function

- Given a network and a set of demands, there are many feasible routings.
- To choose one routing from this set, the standard approach is to define an objective function and then choose the routing that optimizes the objective function.
- Define the objective function as the foregoing quantity z – the objective function is the smallest spare capacity resulting from a routing. Then optimal routing would be the one that maximizes the smallest spare capacity.
- In other words, we avoid routing that lead to a bottleneck link having very little spare capacity.
- This objective promotes a balanced utilization of capacity and does not create hot spots.

Objective Function (Con't)

- Optimization problem:

$$\max \quad z$$

subject to

$$\begin{bmatrix} \mathbf{A} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{A} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \vdots \\ \mathbf{x}(K) \end{bmatrix} = \begin{bmatrix} \mathbf{v}(1) \\ \mathbf{v}(2) \\ \vdots \\ \mathbf{v}(K) \end{bmatrix}$$

$$[\ \mathbf{I} \quad \mathbf{I} \quad \dots \quad \mathbf{I}\ ] \begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \vdots \\ \mathbf{x}(K) \end{bmatrix} + z \cdot \mathbf{1} \leq \mathbf{C}$$

$$x(k) \geq 0, \qquad 1 \leq k \leq K, \qquad z \geq 0$$

Objective Function (*Compact Form*)

- Optimization problem:

$$\max \quad z$$

subject to

$$\begin{bmatrix} \mathbb{A} & \mathbf{0} \\ \mathbb{I} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ z \end{bmatrix} \begin{matrix} = \\ \leq \end{matrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{C} \end{bmatrix}$$

$$\mathbf{x} \geq 0, \quad z \geq 0$$

On the left side, $\mathbf{0}$ is an all 0's vector of size $KN \times 1$, where 1 is an all-1's vector of size $L \times 1$.

- This final form of the optimization problem that defines the optimal routing – called *primal problem*.
- Optimal solution can be obtained by considering the *dual* of this problem.

Optimal Solution Procedure

- Think of the dual variables as row vectors:
- Let $\mathbf{u}(1)^{\mathbf{T}}$ be a $1 \times N$ vector of dual variables corresponding to the first N equality constraints, $\mathbf{u}(2)^{\mathbf{T}}$ be another $1 \times N$ vector of dual variables corresponding to the next N equality constraints, ... Then

$$\mathbf{u}^{\mathbf{T}} := [\mathbf{u}(1)^{\mathbf{T}}, \mathbf{u}(2)^{\mathbf{T}}, ..., \mathbf{u}(K)^{\mathbf{T}}]$$

  denote the $1 \times KN$ vector of dual variables in the first group.
- Clearly, each element of $\mathbf{u}$ corresponds to a (node, demand) pair. Similarly, let

$$\mathbf{y}^{\mathbf{T}} := [y_1, y_2, ..., y_L]$$

  denote the $1 \times L$ vector of dual variables corresponding to the group of L inequality constraints. Each element of $\mathbf{y}$ corresponds to a link.

Dual Problem

- Then the dual problem can be stated as follows:

$$\min \quad (\mathbf{u}^{\mathbf{T}}\mathbf{v} + \mathbf{y}^{\mathbf{T}}\mathbf{C})$$

subject to

$$\begin{bmatrix} \mathbf{u}^{\mathbf{T}} & \mathbf{y}^{\mathbf{T}} \end{bmatrix} \begin{bmatrix} \mathbb{A} & \mathbf{0} \\ \mathbb{I} & \mathbf{1} \end{bmatrix} \geq \begin{bmatrix} 0^{\mathbf{T}} & 1 \end{bmatrix}$$

$$\mathbf{y}^{\mathbf{T}} \geq 0$$

- Note that $0^{\mathbf{T}}$ is a $1 \times KL$ vector of all zeros.
- Optimal routing are shortest paths with y_i^* viewed as link cost *(see p.689 in "Communication Networking : An Analytical Approach, by A. Kumar, D. Manjunath, and J. Kuri, published by Morgan Kaufman, 2004").*

Detailed Solution Procedure

- Lagrange relaxation of the prior primal problem

$$L = z + u^T(v - \mathbb{A}x) + y^T\left(C - (\mathbb{I}x + z\mathbf{1})\right)$$

- The primal problem is

$$\max_{x \succeq 0, z \succeq 0} L(x, z)$$

And in the dual domain

$$\begin{aligned}\theta(u^T, y^T) &= \sup\{x \succeq 0, z \succeq 0 : \\ &\quad u^T v + y^T C - (y^T\mathbf{1} - 1)z - (y^T\mathbb{I} + u^T\mathbb{A})x\}\end{aligned}$$

The optimization problem in the dual domain is then

$$\min_{u^T, y^T \succeq 0} \theta(u^T, y^T)$$

Detailed Solution Procedure (Con't)

- Apply the KKT condition and Lagrange relaxation

$$\frac{\partial L}{\partial z^*} = 1 - (y^T)^* \mathbf{1} = 0 \quad \begin{array}{l} \Rightarrow (y^T)^* \mathbf{1} = 1 \\ \Rightarrow \sum_{l=1}^{L} y_l^* = 1 \quad (1) \end{array}$$

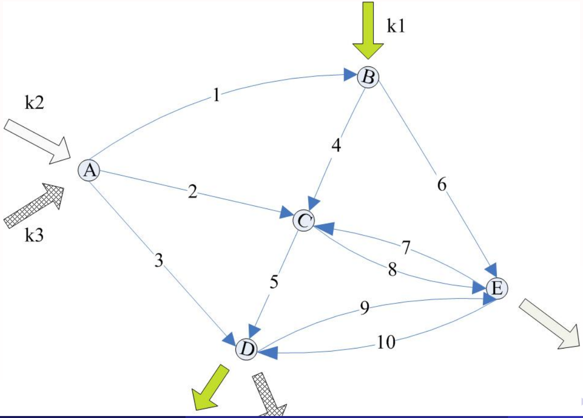
  and

$$\frac{\partial L}{\partial x^*} = (y^T)^* \mathbb{I} + (u^T)^* \mathbb{A} = 0 \Rightarrow \; -(u^T)^* \mathbb{A} = (y^T)^* \mathbb{I} \quad (2)$$

  Define y_l^* as the optimal cost of link l and define $(u^T)^*$ as the payment of unit demand.
- Eq.(1) represents that the sum cost over all the links is constant.
- Eq.(2) shows that the sum payment of each demand is equal to the sum of optimal cost along the path (shortest path).

An Illustrative Example in Optimal Routing

As shown in figure below, given the traffic demands $d(1) = 8$, $d(2) = 10$, $d(3) = 6$, and link capacity
$C^T = \begin{bmatrix} 10 & 3 & 10 & 10 & 4 & 7 & 10 & 8 & 10 & 8 \end{bmatrix}$, model the optimal routing problem and solve it using Lingo/Matlab. Explain the relationship between y and (x, C).



Solution

According to the topology described in above figure. Routing matrix can be written as

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 1 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & -1 & 1 \end{bmatrix},$$

and the traffic demands are

$$v^T = [\ 0 \quad 8 \quad 0 \quad -8 \quad 0 \quad 10 \quad 0 \quad 0 \quad 0 \quad -10 \quad 6 \quad 0 \quad 0 \quad -6 \quad 0\ ].$$

Solution (Con't)

Run with Lingo, $x_{k,l}$, the data rate of demand $k = 1, 2, 3$ over each link ($l = 1, .., 10$), are

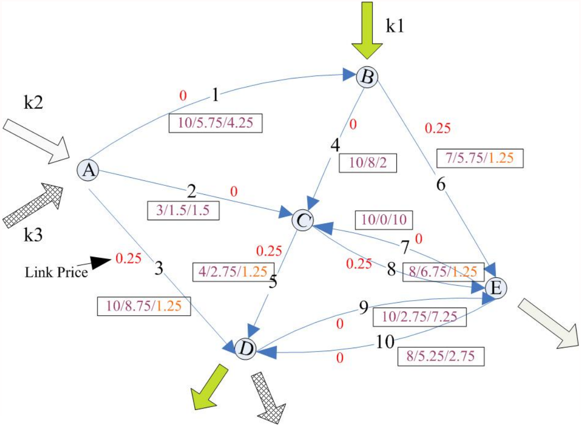
$$\begin{array}{rcccccccccc} l = & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ \mathbf{x}_{1,\cdot} = & [0.00 & 0.00 & 0.00 & 8.00 & 2.75 & 0.00 & 0.00 & 5.25 & 0.00 & 5.25] \\ \mathbf{x}_{2,\cdot} = & [5.75 & 1.50 & 2.75 & 0.00 & 0.00 & 5.75 & 0.00 & 1.50 & 2.75 & 0.00] \\ \mathbf{x}_{3,\cdot} = & [0.00 & 0.00 & 6.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00] \\ \mathbf{z}_{l} = & [4.25 & 1.50 & 1.25 & 2.00 & 1.25 & 1.25 & 10.00 & 1.25 & 7.25 & 2.75] \end{array}$$

$z = 1.25$ and the corresponding dual price of each link y is

$$y = \left[\begin{array}{cccccccccc} 0 & 0 & 0.25 & 0 & 0.25 & 0.25 & 0 & 0.25 & 0 & 0 \end{array}\right].$$

Note: It can be noted that bottlenecks occur when the link with non-zero dual price ($y_l = 0.25$), while the links with 0 dual price are not bottlenecks (here we mean those links with *smallest spare capacity*). Also with KKT condition, we can verify

$$y_l \left[c_l - \sum_{k=1}^{3} x_{k,l} - z^* \right] = 0, \qquad \forall l$$



We can solve the optimal routing problem using *Dijkstra* algorithm with link cost y^* achieved above. The shortest path for each flow is $K_1 \to B \to C \to D$, $K_1 \to B \to C \to E \to D$; $K_2 \to A \to B \to E$, $K_2 \to A \to C \to E$, $K_2 \to A \to D \to E$; $K_3 \to A \to D$.

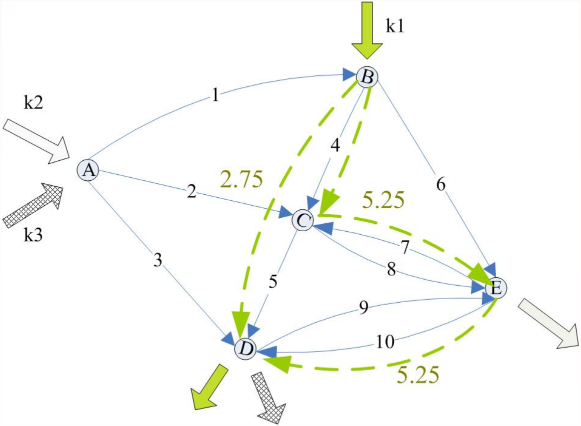


Figure: Traffic Flow with $d(1) = 8$

Solution (Con't)

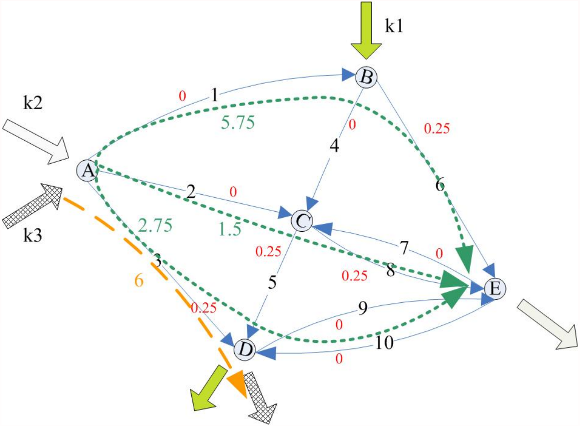


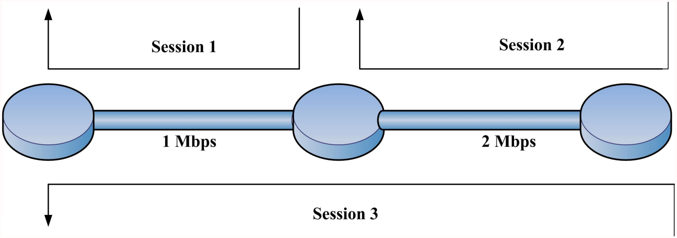
Figure: Traffic Flow with $d(2) = 10$ and $d(3) = 6$

Bandwidth Sharing – Motivation

Q: how to share available bandwidth among competing flows of elastic traffic?

- A typical resource allocation problem
- In a network, requiring equal rates leads to unutilized bandwidth in some links.
- Many notions of fairness can be defined as being desirable, or achievable by specific congestion control and bandwidth-sharing mechanisms.
- Address this complex issue and motivate the use of distributed algorithm for achieving fair bandwidth sharing.

Bandwidth Sharing Model and Definition

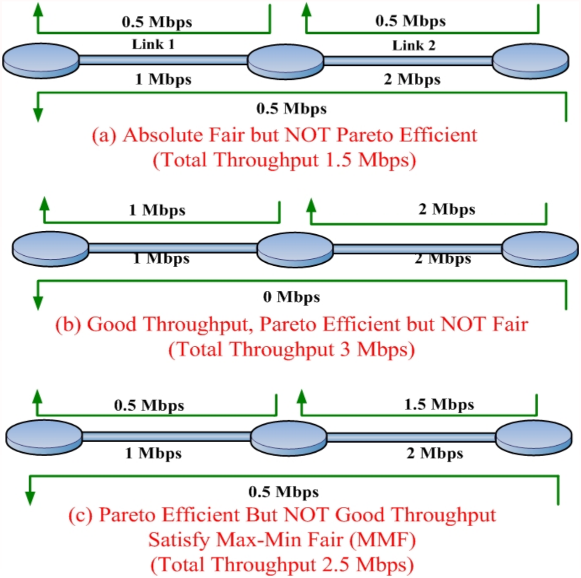


Definition

An allocation of resources in a system is *Pareto efficient* if there does not exist another allocation in which some individual is better off and no individual is worse off.

- In the following, we will see a trade-off between fairness and the total amount of user traffic that the network carries.

Bandwidth Sharing Scenarios



Notation

- L: The set of links; each link is assumed to be unidirectional – a full-duplex link between two nodes is viewed as two links.
- C_l: The capacity of link $l \in L$ (assumed here to be the same in both directions)
- $\mathbf{C}$: The vector of link capacities $(C_1, C_2, ..., C_{|L|})$.
- S: The set of sessions, and consequently,
  - L_s: the set of links used by session $s \in S$.
  - S_l: the set of sessions through link $l \in L$.
  - n_l: the number of sessions through link $l \in L$.
  - r_i: the rate of the ith session, $1 \leq i \leq |S|$; $\mathbf{r} = (r_1, r_1, ..., r_{|S|})^T$ denotes the *rate vector*.

Max-Min Fair Bandwidth Sharing

Definition

A feasible rate $\mathbf{r}$ is *max-min fair* (MMF) for (L, C, S) if it is not possible to increase the rate of a session s, while *maintaining feasibility*, without reducing the rate of some session p with $r_p \leq r_s$.

- Consider a feasible rate vector and look at the smallest rate in this vector. The MMF rate vector has the largest value of this minimum rate.
- Among all feasible rate vectors with this value of the minimum rate, consider the next larger rate. The MMF rate has the largest value of the next larger rate as well, and so on.

Important Property of the MMF Allocation

- The MMF rate vector is characterized in terms of the notion of bottleneck links, which is defined:

Definition

Given a rate vector $\mathbf{r}$, a link l is said to be a bottleneck link for session j if
(i) Link l is saturated (i.e. $f_l(\mathbf{r}) = C_l$) and
(ii) For all the sessions $s \in S$, $r_s \leq r_j$; that is, every session in link l has a rate no more than that of session j.

Example

In the MMF example, both link 1 and link 2 are totally saturated. For the two-link session (i.e., session 3), link 1 is the bottleneck link of session 3 but link 2 is not.

Important Property of MMF

Theorem

A feasible rate vector $\mathbf{r}$ *is MMF if and only if every session* $s \in S$ *has a bottleneck link.*

- Note that with max-min fair flow rates, even though every session has a bottleneck link, *not every link is a bottleneck for some session.*
- If a central entity knew the network topology, the session topology, and the capacities of all the links, then it could calculate the MMF rate using the centralized algorithm – iterative calculation and updating.
- However, another important and more general approach is applying *Network Utility Optimization* and solve the problem in a distributive way.

Basic Network Utility Maximization (NUM)

- What is user utility?
  Utility as function of QoS parameter: throughput, latency, jittering, distortion, energy efficiency,...
- Each source of data, s, has a *utility function*, $U_s(\cdot)$, such that when the source receives the rate r_s it obtains a utility $U_s(r_s)$.
- If the assigned rate vector is r, then the total *utility* of all the users in the network is $\sum_{s \in S} U_s(r_s)$.
- For simplicity, let us assume that all the sources have the same utility function $U(\cdot)$. With w as a constant *weighting* factor for a session, the following utility functions have been proposed:

$$U(r) := w \log(r)$$

- $U(r)$ must be an *increasing*, *strictly concave*, *twice differentiable* function of r.

Utility Maximization for Fairness

- Family of utility function parameterized by $\alpha \geq 0$:

$$U^{\alpha}(r) := \begin{cases} w\dfrac{r^{1-\alpha}}{(1-\alpha)}, & \text{if } \alpha \neq 1 \\ w\log(r), & \text{otherwise} \end{cases}$$

- $\alpha = 0$: throughput maximization (may not be fair)
- $\alpha = 1$: proportional fairness
- $\alpha = 2$: harmonic-mean fairness
- $\alpha = \infty$: MMF

Lemma

All these utility functions are nondecreasing and concave functions of r.

- For all $\alpha \geq 0$, it has

$$\frac{dU^{\alpha}(r)}{dr} = \frac{1}{r^{\alpha}}$$

Basic Framework

- The optimal bandwidth sharing is provided by the solution of following utility maximization problem :

$$\max \sum_{s \in S} U(r_s)$$

subject to

$$\begin{array}{ll} \sum_{s \in S_l} r_s \leq C_l, & \text{for every link } l \in L \\ r_s \geq 0, & \text{for every source } s \in S \end{array}$$

- This is a nonlinear maximization problem, with a concave objective function and linear constraints.

Solution of NUM

- A rate vector r will be optimal for the problem if and only if the KKT conditions hold at this rate vector. Let us denote the dual variable for each link capacity constraint by $p_l \geq 0$, $l \in L$.
- Then the KKT conditions yield the following relationships between the optimal rates r_s, $s \in S$, and the dual variables:

$$\begin{array}{ll} \text{For every } s \in S & \dot{U}(r_s) = \sum_{l \in L_s} p_l \\ \text{If } \sum_{s \in S_l} r_s < C_l & p_l = 0 \\ \text{For every } l \in L & \sum_{s \in S_l} r_s \leq C_l \end{array}$$

- For notational convenience, we denote $dU(r_s)/dr_s$ as $\dot{U}(r_s)$.

Solution of NUM (Con't)

- To understand above KKT conditions, consider the Lagrangian decomposition:

$$L(r,p) = \sum_{s} U(r_s) + \sum_{l \in L} p_l (c_l - \sum_{s \in S_l} r_s).$$

It follows

$$\frac{dL(r,p)}{dr_s} = \frac{dU(r_s)}{dr_s} - \sum_{l \in L_s} p_l = 0,$$

so that

$$\frac{dU(r_s)}{dr_s} = \sum_{l \in L_s} p_l.$$

Interpretation

- One can interpret the dual variable associate with link l as a *price* charged by the link per unit of flow that it carries for a user.
- The KKT conditions tell us that:
    1. *At the fair rate, the derivative of a source's utility is equal to the total price along its path;*
    2. *The price of a link is 0 if the link has spare bandwidth at the fair rate;*
    3. *The rate must be feasible.*

Dual Decomposition

- A distributive algorithm for obtaining the fair rate can be obtained from the dual of the preceding optimization problem. Consider the Lagrangian decomposition:

$$L(\mathbf{r}, \mathbf{p}) := \sum_s U(r_s) + \sum_{l \in L} p_l (C_l - \sum_{s \in S_l} r_s)$$

where p_l, $l \in L$, are nonnegative dual variables or Lagrange multipliers.

- Dual Problem

$$\min \Theta(\mathbf{p})$$

subject to

$$p_l \geq 0 \qquad \text{for every } l \in L$$

where the dual objective function $\Theta(\mathbf{p})$ is defined by

$$\Theta(\mathbf{p}) = \sup\{\mathbf{r} \geq 0: \;\; L(\mathbf{r}, \mathbf{p})\}$$

Dual Decomposition

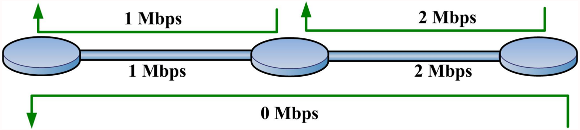
- For given $\mathbf{r} \geq 0$, define, for $s \in S$, $p^{(s)} = \sum_{l \in L_s} p_l$; that is, $p^{(s)}$ is the path cost, which is the sum of the dual variables, or the *total price* per unit of flow, along the path of session s. Now we can see that

$$\begin{aligned}
\Theta(\mathbf{p}) &= \sup\left\{\mathbf{r} \geq 0\colon \ \sum_s U(r_s) - \sum_{l \in L} p_l \sum_{s \in S_l} r_s\right\} + \mathbf{pC} \\
&= \sup\left\{\mathbf{r} \geq 0\colon \ \sum_s \left(U(r_s) - r_s p^{(s)}\right)\right\} + \mathbf{pC} \\
&= \sum_s \sup\left\{\mathbf{r} \geq 0\colon \ \left(U(r_s) - r_s p^{(s)}\right)\right\} + \mathbf{pC}
\end{aligned}$$

- Given $p^{(s)}$, each of the terms in the $\sum_{s \in S}$ can be individually optimized – dual objective function now involves an optimization for each source.

Dual Decomposition (Con't)

- The net *profit* for each source is the utility minus the price it pays; naturally, given the prices, each source would want to adjust its flow to maximize its net profit.
- To illustrate, consider the above example



Applying

$$\sum_{l \in L} p_l \sum_{s \in S_l} r_s = \sum_{s \in S} r_s p^{(s)},$$

we have

$$p_1(r_1 + r_3) + p_2(r_2 + r_3) = r_1(p_1) + r_2(p_2) + r_3(p_1 + p_2).$$

Canonical Distributed Algorithm - Source

- Source Algorithm: we want to maximize the net profit, i.e.

$$\text{For any } s: \;\; r_s = \arg\max \left[U(r_s) - r_s \cdot p^{(s)} \right],$$

which means

$$\frac{dU(r_s)}{dr_s} - p^{(s)} = 0 \Rightarrow p^{(s)} = \dot{U}(r_s) = \frac{1}{r_s^{\alpha}} \quad (3)$$

- Thus, for every source $s \in S$,

$$r_s(k+1) = r_s(k) + \beta(k) \left\{ \frac{1}{[r_s(k)]^{\alpha}} - p^{(s)} \right\}$$

where $\beta(k)$ is the stepsize and k is the iteration indicator.

- α controls the trade-off between carried traffic and fairness: a small α puts emphasis on carried traffic, and a large α emphasize fairness.

Canonical Distributed Algorithm - Link

- Link algorithm (gradient or sub-gradient descent algorithm):

$$p_l(k+1) = p_l(k) - \theta(k) \left[C_l - \sum_{s \in S_l} r_s(\mathbf{p}(k)) \right].$$

where $\mathbf{p}(k)$ is the vector of prices at the kth iteration.

- $\theta(k)$: gain factor – determines the speed of convergence and steady state error
- Certain choices of $\theta(k)$ of distributive algorithm guarantee convergence to globally optimal (r^*, p^*)

Summary

- We have seen two applications in networking design and optimization
- Optimal Routing
- Bandwidth Sharing

Other Research Applications

1. Rate allocation and TCP congestion control
2. Power control in wireless and DSL
3. Wireless networks and cognitive radio networks
4. Cross layer design and optimization
5. P2P networks and Internet pricing issues

Summary (Con't)

- Distributed algorithms are preferred because:
- It is scalable
- It is robust
- Centralized server is not feasible or is too costly

Key Issue

1. Local computation vs. global *communication*;
2. Scope, scale, and physical meaning of communication *overhead*;
3. Theoretical issues: Robustness? Synchronization? Complexity? Stability?
4. ...