

# What is TCP optimizing?

## CS653

In order to understand what TCP is optimizing, let us start with a model for the following resource allocation problem: Given a network and a set of users, (i.e., flows identified as source-destination pairs), what is the optimal allocation of rates to these users? The notion of “optimal” must consider two aspects: utilization, i.e., how much of the available capacity is used; and fairness, i.e., how equitable the allocation is across users.

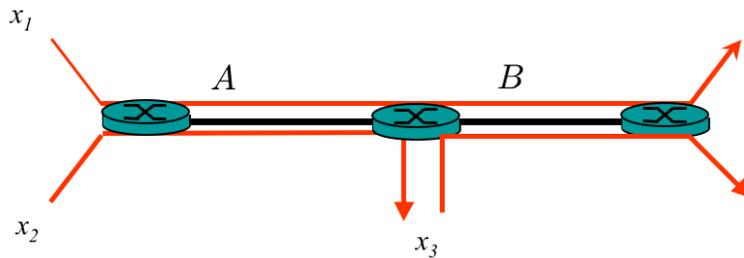


Figure 1: An example of network resource allocation over a two-link network with three flows. Link  $A$  has capacity 2 and link  $B$  has capacity 1.

Consider the network shown in Figure 1. Suppose the capacity of link  $A$  is 2 and the capacity of link  $B$  is 1. Suppose there are three users in the system  $x_1, x_2$ , and  $x_3$  corresponding respectively to flows from node 1 to node 3, node 1 to node 2, and node 2 to node 3. There are several feasible allocations given the capacity constraints on the links. For example, consider the following two allocations:

$$x_1 = 0.5, \quad x_2 = 1.5, \quad x_3 = 0.5 \quad (1)$$

$$x_1 = 0.1, \quad x_2 = 1.9, \quad x_3 = 0.9 \quad (2)$$

Notice that both allocations satisfy link capacity constraints and fully utilize the capacity on each link. If we viewed utilization as the sum of the rates of all users, then the second allocation is superior to the first. On the other hand, the first allocation seems fairer than the second as the latter nearly starves the flow  $x_1$ . Indeed, as we will see shortly, the first allocation is *max-min* fair. We can arrive at the first allocation using the following procedure. Suppose we attempt to divide the capacity on each link equally between all flows traversing the link. Then, on link  $A$ ,  $x_1$  and  $x_2$  would get a rate of 1 each and on link  $B$ ,  $x_1$  and  $x_3$  would get a rate of 0.5 each. However, since  $x_1$ 's route goes through both links  $A$  and  $B$ , it can only send at the minimum of the rates it is assigned on links  $A$  and  $B$ . Thus,  $x_1$  can only send at rate 0.5, which leaves one more unit of capacity on link  $A$ . This remaining capacity can now be allocated to the only source  $x_2$  that is able to use it, thus resulting in the allocation  $x_1 = 0.5, x_2 = 1.5, x_3 = 0.5$ , as in the first allocation.

Although max-min fair resource allocation is a particularly fair allocation, one could consider the second allocation as fair, for example, if  $x_1$  only needed 0.1 for satisfying its application requirements and viewed any additional rate as having zero benefit, while  $x_2$  and  $x_3$  derived higher benefit from higher rates. Thus, it is important to consider the *utility* of the rate allocated to each user.

An optimal allocation is one that maximizes the overall utility across users, thereby implicitly determining the fairness of the resulting allocation. As yet another example, if each user derived a utility simply equal to its rate, then the optimal allocation is  $x_1 = 0, x_2 = 2, x_3 = 1$  as it maximizes the overall utility (sum of the rates) of the system. What is the utility function corresponding to a max-min fair allocation? What is the utility function corresponding to a TCP-fair allocation? To understand this, we formalize the resource allocation problem as an optimization problem.

## 1 Resource allocation as an optimization problem

Let us suppose that each user  $r$  derives a utility (or benefit as perceived by the corresponding application) of  $U_r(x_r)$  when it is allocated a rate  $x_r$ . Then, we would like to allocate the network resources so as to solve the following optimization problem:

$$\max_{x_r \in S} \sum_r U_r(x_r) \tag{3}$$

subject to

$$\sum_{r:l \in r} x_r \leq c_l, \quad l \in L \tag{4}$$

$$x_r \geq 0, \quad r \in S \tag{5}$$

where  $L$  is the set of all links and  $S$  is the set of all sources in the network. The first constraint simply says the total rate across all users sharing a link is less than or equal to the capacity of the link, using the notation  $r : l \in r$  to denote all users sharing a link  $l$ . The second constraint says that all rates must be nonnegative. From classical optimization theory, we know that the above problem admits a unique solution if all  $U_r(\cdot)$  are strictly concave functions. From here on, we will assume that all  $U_r(\cdot)$  are continuously differentiable, nondecreasing, strictly concave functions.

**Example 1: Proportional fairness.** One example of a utility function is  $U_r(x_r) = w_r \log x_r$ . Let the set of rates  $\{\hat{x}_r\}$  denote the optimal solution to the resource allocation problem in (3) for the utility function  $U_r(x_r) = w_r \log x_r$ . Then, we know that the following property is true for any other feasible allocation  $\{x_r\}$

$$\sum_{r \in S} w_r \frac{x_r - \hat{x}_r}{\hat{x}_r} \tag{6}$$

i.e., if we deviate from the optimal to any other solution  $\{x_r\}$ , then the weighted sum of the proportional changes in each user's rate is less than or equal to zero. So, the resource allocation corresponding to the utility function  $U_r(x_r) = w_r \log x_r$  is called *weighted proportionally fair*. If all weights are one, it is simply called *proportionally fair*.

Let us now compute the proportionally fair resource allocation for the two-link network in Figure 1. The resource allocation problem is given by

$$\log x_1 + \log x_2 + \log x_3$$

subject to

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 + x_3 &\leq 1 \end{aligned}$$

and  $x_1, x_2, x_3$  are all non-negative. We make two observations. First, in this example, both constraints above would be satisfied with equality. If not, it means that there is some unused capacity at one or both links and by increasing the rates given to user 2 or 3, this unused capacity can be allocated so as to increase the overall utility. Second, as  $\log x \rightarrow -\infty$  as  $x \rightarrow 0$ , it is clear that the optimal solution will allocate nonzero rates to all users. Thus, we can ignore the non-negativity constraints and use the Lagrange multiplier method to solve the problem as below.

Let  $\lambda_A$  and  $\lambda_B$  be the Lagrange multipliers corresponding to the capacity constraints on links  $A$  and  $B$  respectively. Then, the Lagrangian for this problem is given by

$$L(\mathbf{x}, \lambda) = \log x_1 + \log x_2 + \log x_3 - \lambda_A(x_1 + x_2) - \lambda_B(x_1 + x_3)$$

where  $\mathbf{x}$  is short-hand for the vector consisting of the three rates and  $\lambda$  is the vector of Lagrange multipliers. Setting  $\frac{\partial L}{\partial x_r} = 0$  for each  $r$  gives

$$x_1 = \frac{1}{\lambda_A + \lambda_B}, \quad x_2 = \frac{1}{\lambda_A}, \quad x_3 = \frac{1}{\lambda_B}$$

Using  $x_1 + x_2 = 2$  and  $x_1 + x_3$  yields

$$\lambda_A = \frac{\sqrt{3}}{\sqrt{3} + 1}, \quad \lambda_B = \sqrt{3}$$

Thus, the optimal rates are

$$\hat{x}_1 = \frac{\sqrt{3} + 1}{3 + 2\sqrt{3}}, \quad \hat{x}_2 = \frac{\sqrt{3} + 1}{\sqrt{3}}, \quad \hat{x}_3 = \frac{1}{\sqrt{3}}$$

**Example 2: Minimum potential delay fairness.** Now, suppose  $U_r(x_r) = -w_r/x_r$ . If the source  $r$  is transmitting a file of size  $w_r$ , then  $w_r/x_r$  is the amount of time it would take to transfer the file at a rate  $x_r$ . Thus, the optimization problem in (3) can be thought of as trying to minimize the sum of file transfer delays of all the sources. Hence, the resulting allocation is called *minimum potential delay fair*.

Assuming that all  $w_r$ 's are equal to 1, verify that the solution to the resource allocation problem in (3) satisfies

$$\frac{1}{x_1^2} = \lambda_A + \lambda_B,$$

$$\begin{aligned}\frac{1}{x_2^2} &= \lambda_A, \\ \frac{1}{x_3^2} &= \lambda_B, \\ x_1 + x_2 &= 2, \\ x_1 + x_3 &= 1.\end{aligned}$$

and solve the equations above.

We will see that the potential delay utility function above closely characterizes what TCP is trying to optimize. In fact, the Lagrange multipliers  $\lambda_A$  and  $\lambda_B$  also referred to as *shadow prices* can be interpreted as loss rates on the respective links  $A$  and  $B$ . As can be seen, the optimal sending rate is inversely proportional to the square root of the shadow price or the loss rate—a relationship we have seen before assuming a model [1] that the loss rate is fixed and unaffected by a single TCP flow. To appreciate TCP-fairness and other kinds of fairness including max-min fairness, we introduce a general class of utility functions next.

### 1.1 A general class of utility functions

Let the utility function of user  $r$  be given by

$$U_r(x_r) = w_r \frac{x_r^{1-\alpha_r}}{1-\alpha_r} \tag{7}$$

**Case I: Minimum potential delay fairness.**  $\alpha_r = 2, \forall r$ .

In this case, the utility function of user  $r$  is given by

$$U_r(x_r) = -\frac{w_r}{x_r}$$

and the network's resource allocation is weighted minimum potential delay fair.

**Case II: Proportional fairness.**  $\alpha_r = 1, \forall r$ .

In this case, the utility function in (7) is not well-defined. However, let us consider the derivative of the utility function in the limit as  $\alpha_r \rightarrow 1$ :

$$\begin{aligned}\lim_{\alpha_r \rightarrow 1} U_r'(x_r) &= \lim_{\alpha_r \rightarrow 1} w_r x_r^{-\alpha_r} \\ &= \frac{w_r}{x_r}\end{aligned}$$

Thus, in the limit as  $\alpha_r \rightarrow 1$ , the utility function behaves as though  $U_r(x_r) = w_r \log_{x_r}$ , which leads to weighted proportional fairness. So, we redefine the general class of utility functions as follows:

$$U_r(x_r) = w_r \frac{x_r^{1-\alpha_r}}{1-\alpha_r}, \quad \alpha_r > 0, \alpha_r \neq 1 \tag{8}$$

$$= w_r \log_{x_r}, \quad \alpha_r = 1 \tag{9}$$

**Case III: Max-min fairness.**  $w_r = 1, \alpha_r = \alpha, \forall r, \alpha \rightarrow \infty$ .

We formally define max-min fairness independent of utility functions below first.

**Definition.** A vector of rates  $\{x_r\}$  is said to be max-min fair if, for any other set of rates  $\{y_r\}$  that satisfy the capacity constraints, the following is true: if  $y_s > x_s$  for some  $s \in S$ , then there exists  $p \in S$  such that  $x_p \leq x_s$  and  $y_p < x_p$ .

i.e., an allocation is max-min fair if you can not increase the rate of one flow without decreasing the rate of another flow with a lower rate.

We state without proof the result (proved in [2]) that the utility function above with  $w_r = 1, \alpha \rightarrow \infty$  achieves a max-min fair resource allocation.

## 2 Congestion control: A decentralized solution to the network resource allocation problem

In the previous section, we posed network resource allocation as an optimization problem and solved it in a centralized manner assuming we knew all the users and their utility functions and link capacities. In practice, we would like a decentralized solution. In fact, TCP is precisely that—a decentralized solution to the network resource allocation problem optimizing the utility function  $U_r(x_r) = w_r \frac{x_r^{1-\alpha}}{1-\alpha}$  with  $\alpha = 2$  and  $w_r = \frac{1}{T^2}$ , where  $T$  is the round-trip delay along the path corresponding to TCP flow  $r$ .

You may now wonder how TCP is achieving the optimal solution to the above network resource allocation problem. TCP, as we understand it, responds to loss signals using the AIMD congestion control algorithm. What is the connection between this congestion control algorithm and the network resource allocation optimization problem? This connection lies in interpreting the Lagrange multiplier or the shadow price corresponding to each link ( $\lambda_A$  and  $\lambda_B$  in the example in the previous section) as the loss rate of that link at steady state.

### 2.1 An economic analogy

Consider a scenario<sup>1</sup> where a user  $r$  set his rate  $x_r$  in response to the price  $q_r$  per unit rate being delivered along the path. We define the price  $q_r$  along a path as the sum of the prices  $p_l$  for each link  $l \in r$  along the path. Recall that we use the notation  $l \in r$  to denote the sequence of links along user  $r$ 's path. Suppose you view the user's net value as  $U_r(x_r) - q_r x_r$ , i.e., the utility minus the cost. For a given price  $q_r$ , the user can maximize his value by choosing a rate  $x_r$  such that  $U_r'(x_r) = q_r$ , i.e., when the marginal utility equals the price.

For any given set of prices  $\{p_l\}, l \in L$ , each user  $r$  will choose his rate  $x_r$  so as to maximize his value based on the price  $q_r$  received along the path. Assume that the network makes revenue according to a congestion pricing model that works as follows. A link  $l$  must set the price  $p_l$  to 0 if it is underutilized, i.e., the total sending rate  $y_l$  of all users traversing the link is less than the capacity  $c_l$  of the link. If  $y_l$  exceeds  $c_l$ , then the link can set the price to a positive value. How should the saturated link  $l$  set its price  $p_l$ ? If  $p_l$  is too high, the users traversing  $l$  will reduce their rates in response. If the reduction is so high that the link becomes underutilized, i.e.,  $y_l < c_l$ , then  $l$  makes no revenue. Thus,  $l$  should set  $p_l$  such that the  $y_l$  exactly equals  $c_l$ .

---

<sup>1</sup>The scenario described here is different from the Walrasian auction involving price taking customers described in [3] to explain proportional fairness, or  $\alpha = 1$ .

## 2.2 Decentralized user and link algorithms

Based on the scenario described above, we write the user algorithm for adjusting the rate as follows

$$\dot{x}_r = k_r(x_r)(U_r'(x_r) - q_r), \quad (10)$$

where  $k_r(x_r)$  is any nondecreasing continuous function and the link algorithm for adjusting the price  $p_l$  as follows:

$$\dot{p}_l = h_l(p_l)(y_l - c_l)_{p_l}^+ \quad (11)$$

where  $h_l(p_l) > 0$  is a nondecreasing continuous function. The second term in the product above is simply  $(y_l - c_l)$  if  $p_l > 0$  and is  $\max(y_l - c_l, 0)$  if  $p_l = 0$ . Thus, the link keeps increasing the price  $p_l$  when  $y_l > c_l$  until  $y_l \leq c_l$ , and keeps decreasing  $p_l$  if  $y_l < c_l$  until  $p_l = 0$ .

The above algorithm is known to be globally asymptotically stable [5], i.e., it will converge to the equilibrium from any set of initial rates. Note that at the equilibrium point,  $U_r'(\hat{x}_r) = \hat{q}_r$  and all user paths will have at least one link saturated (as otherwise, the path price would be zero causing the user to increase the rate until some link saturates).

## 2.3 Examples

In the examples below, we assume  $k_r(x_r)$  in equation (10) is simply  $\kappa x_r$  for some constant  $\kappa$ .

**Proportional fairness:**  $U_r(x_r) = w_r \log x_r$

The user algorithm is

$$\dot{x}_r = \kappa(w_r - x_r q_r) \quad (12)$$

At equilibrium, the path price  $q_r$  for a user  $r$  will be  $w_r/\hat{x}_r$ .

**Potential delay fairness:**  $U_r(x_r) = -w_r/x_r$

The user algorithm is

$$\dot{x}_r = \kappa(w_r/x_r - x_r q_r) \quad (13)$$

At equilibrium, the path price  $q_r$  for a user  $r$  will be  $w_r/\hat{x}_r^2$ . Thus, if we think of  $w_r$  as the inverse of the round-trip time squared  $T^2$  to get the familiar  $x_r = \frac{1}{T\sqrt{q_r}}$  relationship at equilibrium that characterizes TCP.

**$\alpha$ -fairness:**  $U_r(x_r) = w_r \frac{x_r^{1-\alpha}}{1-\alpha}$

The user algorithm is

$$\dot{x}_r = \kappa x_r (w_r/x_r^\alpha - q_r) \quad (14)$$

At equilibrium, the path price  $q_r$  for a user  $r$  will be  $w_r/\hat{x}_r^\alpha$ .

## 2.4 Adaptive virtual queue

The adaptive virtual queue algorithm adjusts the price  $p_l$  on a link  $l$  as follows

$$p_l = \left(\frac{y_l}{\bar{c}_l}\right)^\beta \quad (15)$$

where  $\bar{c}_l$  is a virtual queue capacity that is adjusted as

$$\dot{\bar{c}}_l = \alpha_l(c_l - y_l) \frac{+}{\bar{c}_l} \quad (16)$$

where  $\alpha_l$  is a step-size parameter. When the link is underutilized, i.e.,  $c_l > y_l$ , the virtual queue capacity increases thereby decreasing the price. When the aggregate sending rate  $y_l \geq c_l$ , then the virtual queue capacity decreases thereby increasing the price.

## 3 MIMD(n,m)

TBD. Refer stability condition in slides.

## 4 Multipath TCP

TBD. Refer controller and stability conditions in slides.

## References

- [1] J. Padhye, V. Firoiu, D. Towsley, J. Kurose, Modeling TCP Reno performance: a simple model and its empirical validation, *IEEE/ACM Transactions on Networking (ToN)* 2000.
- [2] J. Mo, J. Walrand, Fair End-to-End Window-based Congestion Control, *IEEE/ACM Transactions on Networking (ToN)* 2000.
- [3] Frank Kelly, Aman Maulloo and David Tan, Rate control in communication networks: shadow prices, proportional fairness and stability, *Journal of the Operational Research Society* 49 (1998) 237-252.
- [4] Frank Kelly and Thomas Voice, Stability of end-to-end algorithms for joint routing and rate control, *Computer Communication Review* 35:2 (2005) 5-12.
- [5] R. Srikant, *The Mathematics of Internet Congestion Control*, Birkhauser, 2003.