Optimization Framework for Congestion Control Algorithms

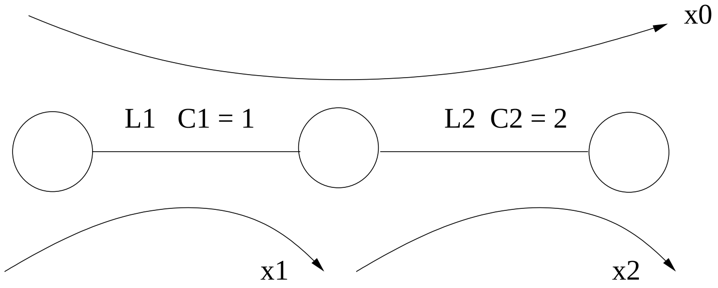
Laila Daniel and Krishnan Narayanan

18th March 2013

Outline of the talk

- A rate allocation example
- Fairness criteria and their formulation as utilities
- Convex optimization - some background
- Network Utility Maximization (NUM) principle
- Primal distributed congestion control algorithm
- Lyapunov functions and stability of primal algorithm
- Summary

A rate allocation example



(x0, x1, x2) = (0.5, 0.5, 1.5) is the Max−min fair allocation

- Links L_1 and L_2 in series with their respective capacities $C_1 = 1$ and $C_2 = 2$ shared by 3 flows x_0, x_1 and x_2 as follows. x_0 threads L_1 and L_2, x_1 is confined to L_1 and similarly x_2 is confined to L_2.
- A 'reasonable allocation' of rates for the flows is the following:

  $$x_0 = 0.5,\ x_1 = 0.5 \text{ and } x_2 = 1.5$$

Fairness is not Equal rates and Pareto efficiency

- Equal rates does not always imply fairness as capacity can be left unutilzed
- For our example, the allocation $x_0 = 0.5$, $x_1 = 0.5$ and $x_2 = 0.5$ leaves a spare capacity of 1 in link 2 unutilized
- Pareto Efficiency
  - An allocation is Pareto efficient if no allocation can be increased without making some one else's allocation smaller.
  - Pareto efficiency is the first principle that a general resource allocation scheme has to satisfy
  - Pareto efficient allocation need not be optimal
- Pareto efficient allocation for our sample network is $x_0 = 0.5$, $x_1 = 0.5$ and $x_2 = 1.5$

Throughput Efficiency vs Fairness

- In our example network the maximum throughput of the network (defined as the sum of the rates of the users) can be 3 for the allocation $x_0 = 0, x_1 = 1$ and $x_2 = 2$, which is greater than the Pareto efficient
- Notice the throughput efficiency comes at the expense of starving x_0, so not acceptable.

Max-min fair (MMF) allocation

- The idea behind the allocation is ' to divide the link capacity equally among the flows sharing the links and if there is any excess capacity share it equally between flows that require it'
- This allocation principle is 'max-min fair allocation'
- Definition: A vector of rates $\mathbf{x} = (x_s)_s \in S$ is max-min fair if it is feasible and no individual rate x_s can be increased without decreasing any other rate equal or smaller
- Max-min fair allocation
  - maximizes the minimum rate
  - can be viewed as giving the maximum protection to the minimum of the allocated rates (absolute property)
  - All unsatisfied sources get the same rate which means that there is no incentive for a source to benefit from inflating its required rate

Max-min Fair (MMF) allocation

- A link l is a bottleneck for a source s if the link is saturated and the source has the largest rate on that link of all the flows sharing that link.
- In our example link L_2 is a bottleneck for flow x_2 , and L_1 is a bottleneck for flows x_0 and x_1.

- Theorem
  A feasible allocation of rates is max-min fair iff every source has a bottleneck link.

- Theorem
  There is a unique max-min fair allocation which can be obtained by progressive filling (algorithm)
  - Max-min fair allocation can be adapted using weights
  - How to specify max-min fair allocation ?
  - Is it the only reasonable allocation ?
  - If not, what about other allocation schemes?

The need for utility function

- Scenario: Suppose the flow x_1 needs minimum rate 0.75 and flow x_0 has little 'worth' for any rate greater than 0.25
- (flow x_1 corresponds to real-time traffic and flow x_0 is a delay-insensitive (elastic) traffic)
- The notion of a utility function quantifies the worth of a given rate to a flow.
- Allocation of network resources based on the utility that sources specify for their rates
- Examples of utility functions are log x and -$\frac{1}{x}$
- Utility function in general is a smooth concave function.
- Utility function can model fairness requirements.

An Example of rate allocation involving Utility

- For the same network scenario as above, but with logarithmic utility for the sources
- Now the resource allocation problem becomes

$$max(log\, x_0 + log\, x_1 + log\, x_2)\ such\, that$$

$$x_0 + x_1 \leq 1$$

$$x_0 + x_2 \leq 2$$

$$x_0 \geq 0, x_1 \geq 0, x_2 \geq 0$$

- Note that the constraints are linear inequalities and the objective function is a concave function.

Convex optimization problem

- If a function f is convex if and only if its negation -f is concave. E.g. The exponential function is convex and the log function is concave
- Geometrically, the set of linear constraints (which in general can include both weak inequalities as well as equations defines a convex set
- The optimization problem of minimizing a convex function (objective) over a convex set called a convex optimization problem
- Maximizing a concave function is equivalent to minimizing a convex function obtained by its negation
- So our rate allocation example is a convex optimization problem

Using Lagrangian to solve the rate allocation example

- Let $\lambda_1 and \lambda_2$be the Lagrangian multipliers correspondingly to the capacity constraints on the links $L_1 \, and \, L_2$ respectively
- The Lagrangian for our problem is given by

  $L(\mathbf{x}, \boldsymbol{\lambda}) = log\, x_0 + log\, x_1 + log\, x_2 - \lambda_1(x_0 + x_1 - 1) - \lambda_2(x_0 + x_2 - 2)$
- Here $\mathbf{x}$ is the data rate allocation vector and $\boldsymbol{\lambda}$ is a vector of Lagrangian multipliers (non-negative real numbers)
- To solve this, set $\frac{\partial L}{\partial x_r} = 0$ for each $r \in 0, 1, 2$
- This gives us

$$x_0 = \frac{1}{\lambda_1 + \lambda_2}, \quad x_1 = \frac{1}{\lambda_1}, \quad x_2 = \frac{1}{\lambda_2}$$

- Using $x_0 + x_1 = 1$ and $x_0 + x_2 = 2$ and solving we get

$$\lambda_1 = \sqrt{3}, \qquad \lambda_2 = \frac{\sqrt{3}}{\sqrt{3}+1}$$

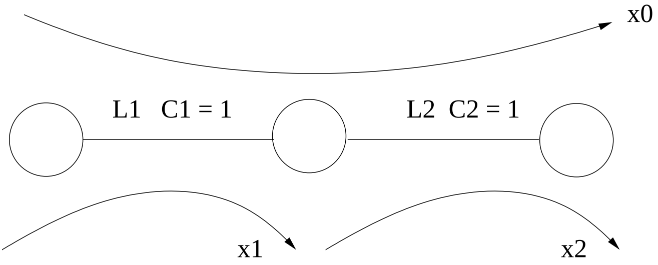
- Substituting the λ values we get the optimal allocation rates:

$$x_0 = 0.42 \;\; x_1 = 0.58 \, and \;\; x_2 = 1.58$$

Observations about Log utility

- Notice in comparison with MMF-allocation x_0 is smaller though in both cases both the links are saturated.
- Note that the non-zero Lagrangian multipliers $\lambda_1 and \lambda_2$ correspond to the case where the capacity constraints are active (i.e. equality) and the Lagrangian multipliers corresponding to $x_0, x_1\, and\, x_2$ are all 0 as these constraints are slack, meaning $x_0 > 0$, $x_1 > 0\, and x_2 > 0$ (CS principle)
- The log utility seems 'natural' in that it pulls up the smaller rates thereby giving them protection as in MMF allocation but not as strongly as MMF in an attempt to maximize the global system utility

Fairness: Max-min vs Proportional



(x0, x1, x2) = (0.5, 0.5, 0.5) is the Max−min fair allocation

(x0, x1, x2) = (1/3, 2/3, 2/3) is the Proportional fair allocation

- In the case of a SINGLE bottleneck link, max-min allocation and proportional fairness allocation coincide
- The sum of the rates received from all links is the same for all the users under proportional fairness criteria
- Engenders the view that a resource does useful 'work' for a flow, so under proportional fairness allocation ALL FLOWS receive the SAME amount of work from the network

Log utility and proportional fairness

- A rate vector $\mathbf{x}^* = (x_s^*)_{s \in S}$ is proportionally fair if for any other rate vector $\mathbf{x} = (x_s)_{s \in S}$ the aggregate of the proportional changes is non-positive i.e,

$$\sum_{s \in S} \frac{x_s - x_s^*}{x_s^*} \leq 0$$

- The log utility function $U_s(x_s) = w_s log\, x_s$ has this property (w_s is the weight)
- The resource allocation scheme corresponding to $U_s(x_s) = w_s log\, x_s$ is called weighted proportionately fair (WPF)
- If $w_s = 1 \; \forall s \in S$, then it is called proportionally fair
- Kelly has shown that AIMD principle is roughly corresponds to proportional fairness
- TCP fairness has been shown to be close to proportional fairness

Minimum potential delay fairness (MPD)

- Here the utility function $U(x) = -\frac{w}{x}$ where w is the weight (usually a constant ≥ 0)associated with the rate x
- Here the motivation for the choice of the utility function is is to minimize the time taken to complete a transfer, i.e, the higher the allocated rate, the smaller the transfer time.
- So the optimization problem can be regarded as minimizing the total time to complete all the transfers.
- Corresponding allocation is MPD allocation
- For MPD allocation, for our rate allocation example we get the following equations in the same manner as before

$$\frac{1}{x_0^2} = \lambda_1 + \lambda_2, \quad \frac{1}{x_1^2} = \lambda_1 \quad \mathit{and} \quad \frac{1}{x_2^2} = \lambda_2$$

- Solution: $x_0 = 0.49 \;\; x_1 = 0.51 \, \mathit{and} \;\; x_2 = 1.51$
- This solution was computed using MAPLE, a symbolic computing package
- MPD fairness is a compromise between MMF and WPF

Some Observations

- The goal of the congestion control algorithm is to provide the rate allocation of the SYSTEM OPTIMUM RATES (obtained by solving the network optimization problem) as the EQUILIBRIUM RATES when all the flows have enough data to send
- The mechanism should be STABLE for this equilibrium rates and should converge to this equilibrium no matter what the initial state is
- The network equilibrium is characterized by some fairness criterion
- How quickly the algorithm can converge to this equilibrium without large oscillations of the network allocated rates is also an important question
- These are some considerations in the design of a congestion control algorithm

Optimal bandwidth sharing

- Optimal bandwidth sharing is got by solving the following utility maximization problem:

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

$$\sum_{s \in S_l} x_s \leq C_l, \forall l \in L$$

$$x_s \geq 0, \forall s \in S$$

- A unique maximizer called primal optimal solution exists as the objective function is strictly concave and the feasible region is a compact convex set
- A rate vector **x** will be optimal iff KKT holds at this rate vector.
- Now use KKT to get the following relationships between the optimal rates x_s and the dual variables $p_l, l \in L$

System Model: Notation

- Based on the paper: Kelly etal. Rate control for communication networks (P13 in the course page)
- A network with J resources, C_j finite capacity of the resource j, for $j \in J$
- *route* r a non-empty subset of J. R set of possible routes. Associate a route r with a user
- $A_{i,j} = 1$ if $j \in r$, j lies on route r and $A_{i,j} = 1$ otherwise
- A rate x_r is allocated to user r and x_r has a utility $U_r(x_r)$
- Assume $U_r(x_r)$ is increasing, strictly concave and continuously differentiable and utilities are additive,
- Aggregate utility rates $x = (x_r, r \in R)$ is $\sum_{r \in R} U_r(x_r)$
- $U = (U_r(.), r \in R)$ $C = (c_j, j \in J)$

System Model

- $SYSTEM(U, A, C)$:

$$max \sum_{r \in R} U_r(x_r)$$

  subject to

$$Ax \leq C$$

  over

$$x \geq 0$$

- These equations say that the system objective is to maximize the total utility of all the users in the network subject to the link capacity constraints
- This system optimization problem is also called the NUM (Network Utility Maximization) problem
- Here we seek to maximize a concave utility function over a convex feasible region defined by the linear constraints

Decomposing the System Model for distributed solution

- We seek a distributed solution to the NUM problem as a centralized solutions are impractical for large scale networks
- We show that the NUM problem can be decomposed into subproblems for users and a problem for the Network
- w_r be the amount user r chooses to pay per unit time
- User r gets a flow rate x_r in proportion to w_r
- $x_r = \frac{w_r}{\lambda_r}$, λ_r charge per unit flow for user r
- $USER_r(U_r; \lambda_r)$:

$$max \left(U_r \left(\frac{w_r}{\lambda_r} \right) - w_r \right)$$

$$\text{over} \quad w_r \geq 0$$

- The above model says that user r tries to maximize its utility 'minus' cost which is to maximize its selfish goal

Network Subproblem

- $NETWORK(A, C; w)$:

$$max \sum_{r \in R} w_r \, log \, x_r$$

  subject to

$$Ax \leq C$$

$$\text{over} \quad w_r \geq 0$$

- Network knows the vector the vector $w = (w_r, \; r \in R)$, w_r what user chooses to pay per unit time
- Network does not know the utility of each user and uses $log \, x_r$ to represent users utility with weight w_r
- Network tries to maximize the weighted log utilities of the users in the network subject to link capacity constraints

Solution Concept

- $SYSTEM(U, A, C) = NETWORK(A, C; w) + USER_r(U_r; \lambda_r)$
- There exists vectors $\lambda = (\lambda_r,\ r \in R)$, $w = (w_r,\ r \in R$ and $x = (x_r,\ r \in R)$ such that $w_r = \lambda_r x_r$
  - w_r solves the $USER_r(U_r; \lambda_r)$ for $r \in R$
  - x solves the $NETWORK(A, C; w)$
  - furthermore, x is the unique solution to the $SYSTEM(U, A, C)$
- This means that the congestion pricing of the network balances the selfish goals of the users with the social goal of the network.
- This **equilibrium** is attained under proportional fairness criterion for resource allocation which **aligns** **selfish optimum objective** of the individual users with the **social optimum objective** of the network

Primal Algorithm

$$\frac{d}{dt}x_r(t) = \kappa_r \left(w_r(t) - x_r(t) \sum_{j \in r} \mu_j(t) \right)$$

where

$$\mu_j(t) = p_j \left(\sum_{s:j \in S} x_s(t) \right)$$

- $x_r(t)$: flow on route r
- $w_r(t)$: what the user r willing to pay per unit time for its flow
- $p_j(y)$: price charged per unit flow by the resource j through the resource j when the total flow through the resource j is y
- $\mu_j(t)$: price charged per unit flow by the resource j at time t
- The differential equation tries to equalize the total cost of the flow against the price the user w_r the user is prepared to pay

Primal Algorithm (contd)

$$\frac{d}{dt} x_r(t) = \kappa_r \left(w_r(t) - x_r(t) \sum_{j \in r} \mu_j(t) \right)$$

where

$$\mu_j(t) = p_j \left(\sum_{s: j \in S} x_s(t) \right)$$

- This equation corresponds to a rate control algorithm for user r which has two components
- Steady increase at a rate proportional to $w_r(t)$
- Steady decrease proportional to the stream of congestion indication signals received
- Note that this equation corresponds to AIMD model of adjusting the source rates
- Key assumptions: time lags and stochastic perturbations ignored in the model

Lyapunov function and the Global stability of the Primal algorithm

- Lyapunov function is obtained from the system objective of the NUM problem
- More precisely it is a relaxation of the optimization problem $SYSTEM(U, A, C)$
- Lyapunov function, a global view of the local description given by the primal algorithm
- Lyapunov function shows that the primal algorithm converges to a unique equilibrium point given by the system optimum.
- Intuitively, Lyapunov function can be regards as a 'hill' and the primal algorithm advances every step to reach the 'top'

Lyapunov function of the primal algorithm

$$\mathcal{U}(x) = \sum_{r \in R} w_r \, log \, x_r \; - \sum_{j \in J} \int_0^{\sum\limits_{s:j \in s} x_s} p_j(y) dy$$

- We can tidy up the appearance of the above equation by introducing the quantity $C_j(y)$
- Let $C_j(y)$ is the rate at which cost is incurred at resource j when the load through it is y
- $\frac{d}{dy} C_j(y) = p_j(y)$
- $p_j(y)$ strictly increasing function of y bounded above by unity

$$\mathcal{U}(x) = \sum_{r \in R} w_r log x_r \; - \sum_{j \in J} C_j \left(\sum_{s:j \in s} x_s \right)$$

Network interpretation of the Lyapunov function $\mathcal{U}(x)$

$$\mathcal{U}(x) = \sum_{r \in R} w_r log x_r \; - \sum_{j \in J} C_j \left(\sum_{s : j \in s} x_s \right)$$

- $\mathcal{U}(x)$ can be interpreted as the weighted sum of the log utilities of the users minus the total congestion price incurred by the network resources (links) in routing the flows through them.
- So, maximizing $\mathcal{U}(x)$ can be regarded as the network maximizing its (benefits minus cost)

Equilibrium point corresponds to maximizing the Lyapunov function $\mathcal{U}(x)$

$$\mathcal{U}(x) = \sum_{r \in R} w_r \, log \, x_r \, - \sum_{j \in J} \int_0^{\sum\limits_{s:j \in s} x_s} p_j(y) dy$$

- $\mathcal{U}(x)$ is strictly concave, its local maximum coincides with the global maximum (implication of maximizing a concave function over a convex set)
- At the local maximum, $\frac{\partial}{\partial x_r}\mathcal{U}(x) = 0$, so we get

$$\frac{\partial}{\partial x_r}\mathcal{U}(x) = \frac{w_r}{x_r} - \sum_{j \in r} p_j \left(\sum_{s:j \in s} x_s \right) = 0$$

- We next solve the equation to get x_r corresponding to $\max(\mathcal{U}(x))$

Equilibrium rates obtained from maximizing $\mathcal{U}(x)$

$$\frac{\partial}{\partial x_r}\mathcal{U}(x) = \frac{w_r}{x_r} - \sum_{j \in R} p_j \left(\sum_{s: j \in s} x_s \right) = 0$$

- So the equilibrium rate x_r is given by

$$x_r = \frac{w_r}{\sum_{j \in R} p_j \left(\sum_{s: j \in s} x_s \right)}$$

- Note that the equilibrium value of x_r is precisely the quantity obtained by setting the primal rate control equation x_r to zero

$$\frac{d}{dt} x_r(t) = \kappa \left(w_r(t) - x_r(t) \sum_{j \in r} \mu_j(t) \right)$$

The rate control algorithm climbs the Lyapunov hill to reach the top

$$\frac{d}{dt}\mathcal{U}(x(t)) = \sum_{r \in R} \frac{\partial \mathcal{U}}{\partial x_r} \times \frac{d}{dt} x_r(t)$$

$$= \kappa \sum_{r \in R} \frac{1}{x_r(t)} \left(w_r - x_r(t) \sum_{j \in r} p_j \left(\sum_{s:j \in s} x_s(t) \right) \right)^2$$

- This shows that $\mathcal{U}(x(t)$ is strictly increasing with t unless $x(t)$ corresponds to x which is the unique x maximizing $\mathcal{U}(x(t)$
- So we conclude that the primal algorithm steadily climbs the hill till it reaches the top.

Summary of the talk

- Utility functions can be used to specify different fairness properties
- Log utility function is associated with proportional fairness
- The optimization problem of congestion control can be decomposed into subproblems solved by each SOURCE and each LINK in the network
- The operating point of the network is given by the equilibrium between the users willingness to pay (in price per unit time) and the system allotted rates computed (packets/unit price) are in equilibrium
- The equilibrium point of the AIMD primal algorithm given by proportional fairness criterion
- The equilibrium point aligns the social optimum with the selfish optimum