Advanced Topics on Cloud Computing **Spring 2021**

Lecture 2: Resource Allocation and Fairness

Lecturer: Dr. Georgios Paschos *Amazon*

This lecture focuses on *Fair Resource Allocation*: we must allocate resources to K users, *what is the best allocation*? We provide a fairness framework for studying this important question for cloud computing systems.

2.1 Introduction to Resource Allocation

In a system with K users we use x_k to denote the amount of resource allocated to user k; the ***allocation*** is then a vector $x = (x_k)_{k=1,\ldots,K}$. For example, $x = (5, 2, 3)$ may indicate that we allocate 5 VMs to user 1, 2 VMs to user 2, and 3 VMs to user 3. The set $\mathcal{X}$ contains all feasible allocations and is called the feasibility region. For example, the plot below provides set $\mathcal{X}$ when we have two users and 10 units of resources, and we must satisfy $x_1 + x_2 \leq 10$, while each user can receive at most 8 units.

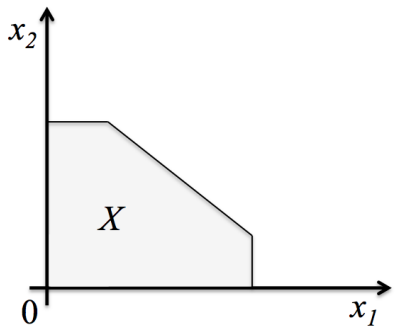

Figure 2.1: Feasible set of allocations.

The satisfaction of user k from receiving x_k resources is described by the utility function $U_k : \mathbb{R}_+ \to \mathbb{R}_+$. We seek to find the optimal resource allocation $(x_1^*, \ldots, x_K^*)$ as the solution to the welfare maximization problem:

Welfare maximization:

$$\max_{x \in \mathcal{X}} \sum_{k=1}^{K} U_k(x_k). \tag{2.1}$$

Our blanket assumptions are:

- Set $\mathcal{X}$ is bounded and restricted to the positive orthant: $x_k \geq 0,\ k = 1, \ldots, K$.
- Utilities are non-decreasing, $U_k(x_k^1) \geq U_k(x_k^2)$ if $x_k^1 > x_k^2,\ k = 1, \ldots, K$.
- Utilities are concave.
- Set $\mathcal{X}$ is convex.

First some explanations about the assumptions. The set is bounded since resources are in general finite, and allocations are non-negative as we cannot allocate negative resources. The utility functions are non-decreasing due to the logic that more resources can not harm satisfaction, while they are concave due to *diminishing returns*: the incremental value of resources becomes smaller as the allocation increases. For instance, an extra 1Mbps of Internet speed is very important for a user when the allocated connection speed is 1Mbps, but not so when it is 500Mbps. Finally, the set $\mathcal{X}$ is assumed convex for simplicity. Non-convex resource allocation problems are encountered in practice, but left out of scope. We mention that such problems are often solved by convexification.

The resource allocation problem is a constrained convex optimization problem and can be solved using any of the available convex algorithms, e.g., projected gradient ascent (lazy or greedy), interior point method, ADMM, etc.

> Most resource allocation problems can be stated as welfare maximization problems.

2.2 What is fairness?

Fairness has been a subject of philosophical studies since the era of Plato. Here we reduce the scope to an engineering explanation in the setting of resource allocation. Consider three persons sitting in a bar and deciding how to share a pitcher of 1lt of beer.[1] They inevitably face the question "what is a fair way to allocate the beer?" A simple answer is share 1lt of beer equally $(1000/3, 1000/3, 1000/3)$. However the answer may become less trivial if we add further detail:

- Each person has a different glass. Then an allocation $(500, 250, 250)$ that barely fills the corresponding glasses would be optimal.
- Third person does not want to drink more than 100ml. Then an allocation $(450, 450, 100)$ may seem fair to many.
- Second person is extremely thirsty. Then an allocation $(300, 400, 300)$ might be fair to some who think that a thirsty person deserves to drink more.

When sharing cloud computing resources, these questions become very relevant, and involve contractual agreements, urgent needs for scaling up resources, limits of consumption, and deciding how to allocate multiple resources. Below, we provide a mathematical framework to address these questions methodologically.

2.2.1 Pareto efficiency

In resource allocation we experience competition for resources; one user can get more satisfaction if another gets less. There exist, however, suboptimal allocations in set $\mathcal{X}$ from where several users can improve without any user loosing utility. To classify allocations accordingly we introduce the concept of Pareto efficiency.

Given an initial allocation $x \in \mathcal{X}$, a ***Pareto improvement for*** x is another allocation $y \in \mathcal{X}$ where some users improve their utilities, but no users decrease their utility. It should hold:

$$U_k(y_k) \geq U_k(x_k), \quad k = 1, \dots, K,$$

and the inequality is strict for at least one user. In our beer example, if the current allocation is $(250, 125, 125)$ filling the glasses half-way, we can obtain a Pareto improvement by pouring more beer to any person.

We say that an allocation $x \in \mathcal{X}$ is ***Pareto efficient*** if it admits no Pareto improvement. Furthermore, the set of all Pareto efficient solutions is called Pareto frontier, see fig. 2.2. Practically speaking, solutions that are not Pareto efficient should be avoided, in the sense that they can be improved without harming anyone. Indeed, we have the following result.

Claim 2.1 *Let x^* be a solution to the welfare maximization problem* (2.1)*. Then x^* is Pareto efficient.*

One can prove the claim by showing that the total welfare can be improved by a Pareto improvement. Hence, a welfare maximal allocation is always Pareto efficient. As explained below, by selecting a fairness objective, we may choose among many Pareto efficient allocations which is the best.

> Pareto efficient allocations are candidate solutions to fairness objectives.

[1] Peculiar thing about the author: he is allergic to beer.

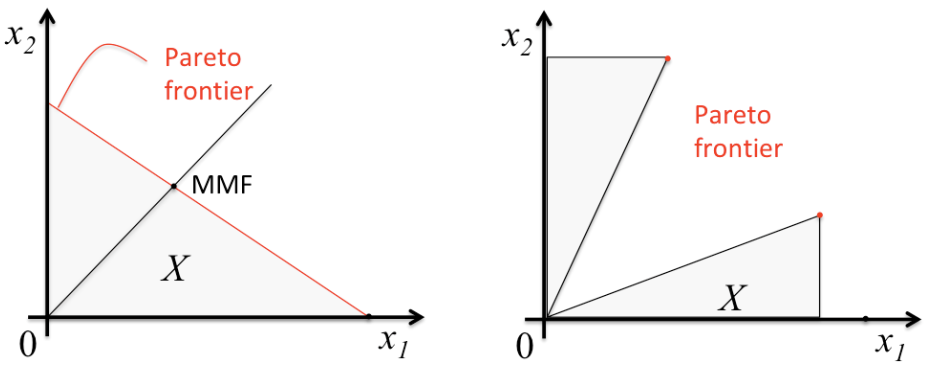

Figure 2.2: (left) A convex set with the Pareto frontier and its unique Max-Min Fair (MMF) point. (right) A non-convex set with no MMF point.

2.2.2 Max-min fairness

We say that $x \in \mathcal{X}$ is ***max-min fair*** if for any $y \in \mathcal{X}$ it holds:

$$y_m > x_m \quad \Rightarrow \quad \exists\, n \neq m : y_n < x_n \leq x_m.$$

Therefore, if x is max-min fair, any allocation y that improves the utility of a user m, must deteriorate the utility of another user n that was originally no richer than m. Max-min fairness captures the concept of egalitarian fairness, where each user receives as equal resources as possible.

Claim 2.2 *If $\mathcal{X}$ is convex, a max-min fair vector exists and is unique, see [1]. We will call it x^{MMF}.*

Claim 2.3 *Let $\mathbf{1}$ be a K-dimensional vector of ones, if $\beta\mathbf{1}$ is Pareto efficient, then it is also max-min fair.*

> Max-Min fairness is the objective of sharing resources in a way as equal as possible, while still Pareto efficient.

2.2.2.1 Progressive filling algorithm

We describe an iterative algorithm that progressively modifies allocations such that they converge to the max-min fair allocation. The algorithm allocates resources to all users to marginally increase their utility equally, progressively eliminating users from options when they reach their maximum allowable allocation. In [3], the algorithm is presented in a network setting.

At iteration i of the algorithm, the allocation is denoted with $x^{(i)}$. The next allocation is:

$$x_k^{(i+1)} = \begin{cases} x_k^{(i)} + \eta^{(i)} \frac{\partial U_k(x^{(i)})}{\partial x^{(i)}} & \text{if } x_k^{(i)} < x_k^{\max} \\ x_k^{(i)} & \text{if } x_k^{(i)} \geq x_k^{\max} \end{cases}$$

where $\eta^{(i)}$ is a stepsize chosen to be sufficiently small to detect when allocations hit maximum.

Claim 2.4 *If $\mathcal{X}$ is convex, the progressive filling algorithm converges to x^{MMF}.*

2.2.2.2 Alpha fairness

Finding x^{MMF} can be cast as a lexicographic maximization problem, where we maximize the utility of the worse-off user, and then subject to that maximizing the utility of second worse-off user, and so on, [2]. For convex sets, there is a simpler way to find max-min fair vectors. For parameter $0 \leq \alpha < \infty$, define the family of concave ***alpha-fair*** functions:

$$g_\alpha(x) = \begin{cases} \frac{x^{1-\alpha}}{1-\alpha} & \text{if } \alpha \in [0,1) \cup (1,\infty) \\ \log x & \text{if } \alpha = 1 \end{cases}$$

Lemma 2.5 *Let $x^*(\alpha)$ be the solution to the welfare maximization problem (2.1) when $U_k(x_k) = g_\alpha(x_k)$, then:*

$$\lim_{\alpha\to\infty} x^*(\alpha) = x^{MMF}.$$

The proof is provided in [4].

Practically speaking, maximizing $\sum_{k=1}^{K} \frac{x_k^{1-\alpha}}{1-\alpha}$ for large values of α allows us to approximate the max-min fair resource allocation by the solution to the welfare maximization problem.

We note that for $\alpha = 0$, the objective of Alpha fairness becomes to maximize $\sum_k x_k$, corresponding to the maximum total system efficiency. For interim values of α we obtain a spectrum of other fairness objectives.

Technically speaking, for $\alpha > 0$, the alpha fair functions are strictly convex, leading to unique solutions on convex set $\mathcal{X}$.

2.2.3 Proportional fairness

In max-min fairness we aim to allocate to users as equal resources as possible. Sometimes this has adverse effects in the efficiency of a system, because the resources for some users are much more expensive than others. A typical example is that of wireless communication systems, where a poorly located user (e.g. situated at the cell edge, or in a garage) can require a very large time allocation from the scheduler in order to receive a transmission rate equal to other users. In such situations, a practical desirable condition is to skew the allocation to provide more resources to favorable users, but avoid starving the unfavourable ones.

We say an allocation $x \in \mathcal{X}$ is ***proportional fair*** (PF) if for any other allocation $y \in \mathcal{X}$ the sum of proportional changes is non-positive:

$$\sum_k \frac{y_k - x_k}{x_k} \leq 0. \tag{2.2}$$

for all users m, n for which the allocation is not maximal. The proportionally fair vector is also the solution to the problem:

$$\max_{x\in\mathcal{X}} \sum_{k=1}^{K} \log(x_k). \tag{2.3}$$

Notice, that this corresponds to Alpha fairness for $\alpha = 1$. As a result, we have that PF allocation exists and it is unique.

> Proportional fairness ($\alpha = 1$) is a compromise between maximum system efficiency ($\alpha = 0$) and egalitarian fairness ($\alpha \to \infty$), often pursued in systems where equality may come at a very high price (e.g. in wireless downlink systems).

Example

Find the PF allocation when a user consumes double the resources from the other for the same allocation, and there is in total 1 unit of resources.

Assuming user 1 is the user that wastes resources, we may express the set of feasible allocations as:

$$\mathcal{X} = \left\{(x_1, x_2) \in \mathbb{R}_+^2 \mid \ 2x_1 + x_2 \leq 1\right\}.$$

Consider the Lagrangian function:

$$L(x_1, x_2, \lambda) = \log x_1 + \log x_2 + \lambda(2x_1 + x_2 - 1)$$

From Karuhn-Kush-Tucker conditions, at optimality the first-order optimality conditions for the Lagrangian must be satisfied. This yields:

$$\frac{\partial L}{\partial x_1} = 0 \Leftrightarrow 1/x_1 + 2\lambda = 0 \Leftrightarrow x_1 = -1/(2\lambda)$$
$$\frac{\partial L}{\partial x_2} = 0 \Leftrightarrow 1/x_2 + \lambda = 0 \Leftrightarrow x_2 = -1/\lambda$$
$$\frac{\partial L}{\partial \lambda} = 0 \Leftrightarrow 2x_1 + x_2 = 1 \Rightarrow -2/\lambda = 1 \Rightarrow \lambda = -2$$

Since the welfare optimization for proportional fairness ($\max_{x\in\mathcal{X}} \log x_1 + \log x_2$) is a strictly convex maximization, it has a unique solution, which is $(x_1^*, x_2^*) = (1/4, 1/2)$. Instead, it is not hard to see that the MMF point is $(1/3, 1/3)$, while the sum of resources can be maximized at $(0, 1)$.

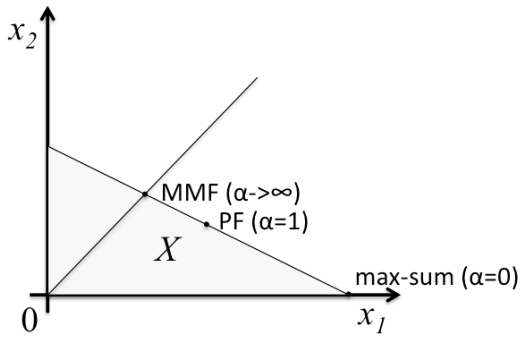

Figure 2.3: Three different allocations achieved via solving the welfare maximization for alpha-fair utilities, Max-Min Fair ($\alpha \to \infty$), Proportionally fair ($\alpha = 1$) and max sum ($\alpha = 0$).

Finally, for PF $(1/4, 1/2)$, MMF $(1/3, 1/3)$, and max total $z = (0, 1)$ we verify (2.2):

$$\frac{1/3 - 1/4}{1/4} + \frac{1/3 - 1/2}{1/2} = 1/3 - 1/3 = 0$$

$$\frac{0 - 1/4}{1/4} + \frac{1 - 1/2}{1/2} = -1 + 1 = 0$$

2.3 Multi-resource fairness

In cloud computing, users may require more than one resource, e.g., memory and CPU. User $k = 1, \ldots, K$ has a requirement for w_{kj} amount of resource j. Let x_k denote its allocation. We have available c_j resources for type j. A fundamental question in cloud computing is to determine the best allocation x that additionally ensures the capacities are not exceeded, i.e.,

$$\forall j \quad \sum_k w_{kj} x_k \leq c_j.$$

Let us take a look at a simple toy example from [5]. There are two users with requirements $(w_{11}, w_{12}) = (1, 4)$, and $(w_{21}, w_{22}) = (3, 1)$, indicating that first user needs more memory (4GB for each CPU) and the second needs more CPU (3CPUs for each 1GB). As the figure shows, we have a server system with 9CPUs and 18GB of memory, and we must decide how to split these resources to the two users.

We introduce the concept of ***Dominant Resource Fairness*** [5] which is the prevalent approach used in the industry. For user k, we say j_k is the dominant resource if $j_k \in \arg\max_j w_{kj}/c_j$. Then, let $y_k = w_{kj_k} x_k / c_{j_k}$ be the dominant resource allocation for user k. We say x is ***Dominant Resource Fair***, if y is max-min fair.

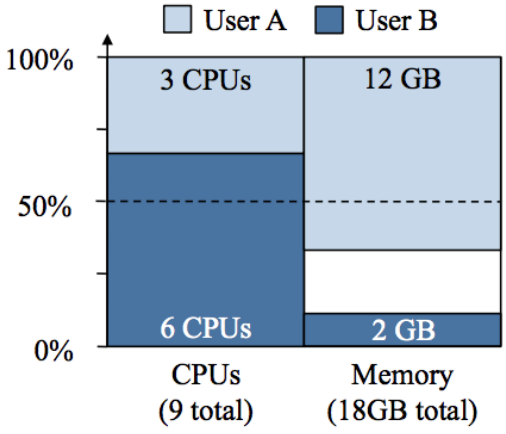

Figure 2.4: Example of resource allocation in cloud computing from [5].

> Dominant Resource fairness is the objective of sharing multiple resources such that the dominant resource for each user is scaled in an as equal as possible.

In the simple example presented before, we work as follows. First, we determine the dominant resource for each user:

$$\text{for user 1, } \frac{w_{11}}{c_1} = \frac{1}{9}, \text{ and } \frac{w_{12}}{c_2} = \frac{2}{9}$$
$$\text{for user 2, } \frac{w_{21}}{c_1} = \frac{3}{9}, \text{ and } \frac{w_{22}}{c_2} = \frac{1}{18}$$

hence CPU for user 1 and memory for user 2. It follows that $y_1 = 2x_1/9$ and $y_2 = 3x_2/9$. Also, we must satisfy the capacity constraints for the two resources:

$$\text{for CPU, } \quad x_1 + 3x_2 \leq 9$$
$$\text{for Memory, } \quad 4x_1 + x_2 \leq 18$$

which are equivalent to

$$\text{for CPU, } \quad y_1 + 2y_2 \leq 2$$
$$\text{for Memory, } \quad 6y_1 + y_2 \leq 6$$

Finally, the DRF vector can be recovered by solving the following welfare maximization:

$$\max_{y \geq 0} \frac{y_1^{1-\alpha}}{1-\alpha} + \frac{y_2^{1-\alpha}}{1-\alpha}$$
$$\text{s.t. } y_1 + 2y_2 \leq 2$$
$$6y_1 + y_2 \leq 6$$

for a limiting value of α. In this case, the MMF point is on the line $y_1 = y_2 = y$ (we argue that this point is Pareto). Observe that the two constraints become $y \leq 2/3$ and $y \leq 6/7$, hence the MMF point is $(y_1, y_2) = (2/3, 2/3)$. Finally, the DRF point is obtained by back substituting, and it is equal to $(x_1, x_2) = (3, 3)$. The allocation is then 3 CPUs and 12 GBs for user 1 and 6 CPUs and 2 GBs for user 2. This is the allocation shown in the figure.

2.4 Quiz

In a caching application, we are given enough storage to cache 100 videos. The cache hit probability for video m is given by $h_m = 1 - e^{\lambda_m t_m}$, where λ_m is rate of requests (popularity), and t_m is a tunable parameter. For the hit

probabilities it must be that $h_m \in [0,1]$ and $\sum_{m=1}^{M} h_m = B$, where $B = 100$ and $M = 1000$. The latter constraint implies that the tunable parameters t_m are such that the cached videos always fit in the available storage.

The different videos are not equally important. We assume that the video's important is related to its rank in the following manner: $w_m = 1/m$. We desire to select the tunable parameters in order to achieve weighted max-min fairness. Use cvxopt to plot the hit probability distributions that correspond to this solution.

References

[1] B. Radunovic and J.-Y. Le Boudec, *A Unified Framework for Max-Min and Min-Max Fairness with Applications*, Allerton, 2002.

[2] D. Nace and M. Pioro *A Tutorial on Max-Min Fairness and its Applications to Routing, Load-Balancing and Network Design*, 2006.

[3] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1992.

[4] J. Mo and J. Walrand, *Fair End-to-End Window-based Congestion Control*, IEEE/ACM Transactions on Networking, 8(5), pp. 556–567, 2000.

[5] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, and I. Stoica, *Dominant Resource Fairness: Fair Allocation of Multiple Resource Types*, NSDI, 2011.

[6] C. Joe-Wong, S. Sen, T. Lan, and M. Chiang, *Multi-Resource Allocation: Fairness-Efficiency Tradeoffs in a Unifying Framework*, Transactions on Networking, 2012.

[7] T. Bonald and J. Roberts, *Multi-Resource Fairness: Objectives, Algorithms and Performance*, ACM Sigmetrics, 2015.