Multi-Resource Allocation: Fairness-Efficiency Tradeoffs in a Unifying Framework

Carlee Joe-Wong*, Soumya Sen*, Tian Lan†, Mung Chiang*
*Department of Electrical Engineering, Princeton University, Princeton, NJ 08544
†Department of Electrical and Computer Engineering, George Washington University, Washington, DC 20052
Email: {cjoe, soumyas, chiangm}@princeton.edu, tlan@gwu.edu

***Abstract*—Quantifying the notion of fairness is under-explored when users request different ratios of multiple distinct resource types. A typical example is datacenters processing jobs with heterogeneous resource requirements on CPU, memory, etc. A generalization of max-min fairness to multiple resources was recently proposed in [1], but may suffer from significant loss of efficiency. This paper develops a unifying framework addressing this fairness-efficiency tradeoff with multiple resource types. We develop two families of fairness functions which provide different tradeoffs, characterize the effect of user requests' heterogeneity, and prove conditions under which these fairness measures satisfy the Pareto efficiency, sharing incentive, and envy-free properties. Intuitions behind the analysis are explained in two visualizations of multi-resource allocation.**

I. Introduction

A. Motivation

Many works have studied the fairness of allocating a *single* type of resource. Fairness can be quantified with a variety of metrics, e.g. Jain's index [2]. Other notions of fairness, including proportional and max-min fairness, can be represented as maximization of α-fair utility functions [3]. These approaches, as well as others from economics and sociology, have recently been unified as the unique family of functions satisfying four axioms for fairness metrics [4]. Works such as [5]–[7] have also studied the tradeoff between fairness and efficiency.

Despite these works, allocating *multiple* types of resources has been much less studied, with [1] a recent notable exception. Indeed, it is unclear what "fair" means for a multi-resource allocation. Each user in a network requires a certain *combination* of different resources to process one job, which may differ from user to user. For example, datacenters allocate resources (memory, CPUs, etc.) to competing users with different requirements. One user might run computational jobs requiring more CPU cycles than memory, while another might require the opposite. Figure 1's simple example illustrates the need for multi-resource fairness functions: here two users require CPUs and memory to perform jobs. User 1 requires 2 GB of memory and 3 CPUs per job, while user 2 needs 2 GB and 1 CPU per job. There are a total of 6 GB and 4 CPUs.

Many allocations might be considered "fair" in this example: should users be allocated resources in proportion to their resource needs? Or should they be allocated resources so as to process equal numbers of jobs? The fairness measure proposed in [1], called **Dominant Resource Fairness (DRF)**, allocates resources using max-min fairness on dominant resource shares.

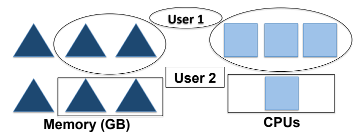


Fig. 1. An example of multi-resource requirements in datacenters.

In this example, DRF would allocate 0.76 jobs to user 1 and 1.71 jobs to user 2, for a total of 2.47 jobs processed. But this allocation is somewhat inefficient; e.g., allocating 0.17 jobs to user 1 and 2.83 jobs to user 2 yields a total of 3 jobs. An in-between allocation is realized by adapting the well-known α-fairness for multiple resources as in Section III-B. For $\alpha = 0.5$, user 1 has 0.57 jobs and user 2 has 2.29 jobs, for 2.86 total jobs. Each of these allocations represents one point of the fairness-efficiency tradeoff. This paper develops a unifying framework for studying this tradeoff in light of multiple types of resources and heterogeneity in users' resource requirements.

Multi-resource allocation problems arise in increasingly many applications. Datacenters selling *bundles* of CPUs, memory, and network bandwidth are just one example. In fact, even the classical problem of bandwidth allocation in a congested network can be viewed as a multi-resource allocation. Given a network and its topology, we can view each link as a separate resource with distinct capacity. Users are represented by network flows using a pre-defined subset of links, and resource requests on all the links are for each user.

In general, multi-resource allocation *cannot* be turned into single-resource allocation by assuming different resources are interchangeable. For example, CPU allocation cannot meet a cloud client's networking bandwidth requirements.

B. Unique Challenges of Multi-Resource Fairness

The following new challenges on fairness arise due to the presence of multiple types of resources:

- In a single-resource scenario, users' resource requirements can be represented with a scalar. With multiple resources, users have distinct vectors of resource requirements, which must be scalarized before fairness can be evaluated. We present two ways to visualize user heterogeneity in Section III-A and two scalarization methods in Section III-B, yielding parametrized families of multi-resource fairness measures that satisfy the axioms of [4].
- In a single-resource scenario, the most efficient allocation uses the entire resource. In a multi-resource scenario,

however, users' heterogeneous resource requirements may not allow each resource to be completely used. Even how to measure efficiency is unclear: should we use the total number of jobs allocated[1]? Or the amount of leftover resource capacity? Section V numerically examines both of these efficiency metrics, while Props. 1 and 2 and their corollaries examine the impact of user heterogeneity on the number of jobs processed.
- The extension of max-min fairness to multiple resources is shown in [1] to satisfy such properties as Pareto-efficiency for certain parameter values. We characterize the parametrizations under which our multi-resource fairness functions satisfy Pareto-efficiency, sharing incentive, and envy-freeness (Props. 3-5 and their corollaries).
- The existence of a fairness-efficiency tradeoff depends on both users' resource requirements and the subsequent fairness evaluation. We show that a greater emphasis on equity need not always decrease efficiency (Prop. 6) and give analytical conditions on when the fairness-efficiency tradeoff exists (Props. 7 and 8 and their corollaries).

After Section II's further discussion of related work, Section III develops two new families of fairness functions, which we call **Fairness on Dominant Shares (FDS)** and **Generalized Fairness on Jobs (GFJ)**. FDS includes the fairness measure DRF proposed in [1] as a special case. We investigate key properties of these functions in Section IV, and derive conditions under which they are satisfied by FDS and GFJ. Section V then applies our fairness functions to numerical examples of datacenters. We examine the relationship between the fairness-efficiency tradeoff and FDS and GFJ parametrizations.

The technical report contains proofs of all propositions [8].

II. Related Work

Much of the existing theory on the fairness of resource allocations is devoted to allocations of a single resource [4], [9], [10] (e.g. allocating available link bandwidth to network flows [11]–[13]). The recent work [4] develops the following family of fairness functions for a single resource, unifying previously developed fairness measures. It was proven that this family, parametrized by two numbers, is the *only* family of functions satisfying four simple axioms of fairness metrics:

$$f_{\beta,\lambda}(\mathbf{x}) = \mathrm{sgn}(1-\beta)\left(\sum_{i=1}^{n}\left(\frac{x_i}{\sum_{j=1}^{n}x_j}\right)^{1-\beta}\right)^{\frac{1}{\beta}}\left(\sum_{i=1}^{n}x_i\right)^{\lambda}, \tag{1}$$

where $\beta \in \mathbb{R}$ and $\lambda \in \mathbb{R}$ are parameters. The parameter β gives the "type" of fairness measured by (1), and the parameter λ gives the emphasis on efficiency. A larger $|\lambda|$ indicates greater emphasis on efficiency over fairness. If we take $\lambda = \frac{1-\beta}{\beta}$, then taking the limit as $\beta \to 1$ yields proportional fairness.

Even multi-resource allocation problems, such as scheduling jobs in a datacenter, are often treated as a single resource problem (e.g. the Hadoop and Dryad schedulers [14]). Given the limitations of this approach, [1] generalizes max-min fairness to multiple resource settings. Our work develops a unified analytical framework for fairness of multi-resource allocations. In particular, in contrast to [1], we study the tradeoff between fairness and efficiency in multi-resource settings.

[1]The phrases "jobs allocated" and "jobs processed" are used interchangeably throughout the paper.

III. Fairness-Efficiency of Multi-Resource Allocations

We first present "dual" visualizations of heterogeneity among users' requirements for multiple resources in Section III-A. Section III-B then develops two new families of fairness functions, which scalarize these resource requirement vectors in order to evaluate the fairness of multi-resource allocations. These two families are Fairness on Dominant Shares (FDS) and Generalized Fairness on Jobs (GFJ). FDS measures the fairness of users' resource allocations by accounting for both the number of jobs allocated to each user (a function of the resources available) and the heterogeneity in different resource requirements across users. GFJ, on the other hand, assumes that users' utility depends solely on the number of jobs they are allocated, irrespective of their differing resource needs.

A. Visualizing User Heterogeneity

A major challenge of multi-resource fairness is incorporating the heterogeneity of different users' requirements for different resources into the assessment of its fairness. Visualizing this heterogeneity can yield useful insights. Moreover, Section V examines in detail how heterogeneity affects the optimal allocation and achieved efficiency.

Figure 2 provides two ways to visualize user heterogeneity. Each user j requires R_{ij} of resource type i for each job.

In the first (top) visualization, each dimension and associated axis corresponds to a different type of resource (two types of resources here for visual simplicity); the box represents the resource constraints. The slope σ_i of the line corresponding to each user i is the ratio of that user's requirements for the two resources. The heterogeneity of users' resource requirements can be captured with the variance of the $\{\sigma_i\}$, which are assumed to be realizations of a random variable σ. Homogeneity occurs at 0 variance, i.e., the dashed line becomes straight. Heterogeneity increases with the variance of σ.

In the bottom visualization, each dimension and associated axis corresponds to the jobs allocated to each user (two users here for visual simplicity), with feasible allocations in the shaded region bounded by linear resource constraints. The slope τ_i of constraint line i reflects the ratio of user 1's and user 2's requirements for resource i. Again, the variance of the τ_i, which are treated as realizations of a random variable τ, captures the heterogeneity of users' resource requirements. Homogeneity occurs at zero variance; in that case the resource constraints reduce to one constraint. Heterogeneity increases with the variance of τ.

B. Defining Multi-Resource Fairness

1) Fairness on Dominant Shares (FDS): As defined in [1], a user's **dominant share** is the maximum share of any resource allocated to that user.

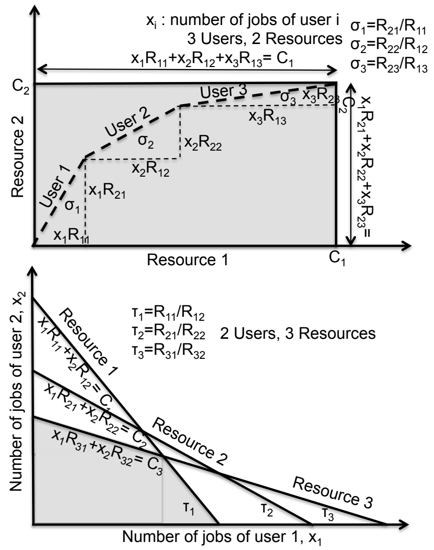


Fig. 2. Two visualizations of user heterogeneity. The lines in the top graph show the ratio of users' requirements for two different resources, while the lines in the bottom graph show the feasible allocation region. The slopes of those lines reflect the ratio of two users' requirements for each resource.

Let x_j denote the number of jobs allocated to each user j and C_i the capacity of each resource i. Then we have the resource constraints $\sum_{j=1}^{n} R_{ij}x_j \leq C_i$ for all resources i, where R_{ij} is the amount of resource i which user j requires for one job, and there are n users. For ease of notation, we define $\gamma_{ij} = R_{ij}/C_i$ as the share of reosurce i required by user j to process one job. We let

$$\mu_j = \max_i \left\{ \frac{R_{ij}}{C_i} \right\} \tag{2}$$

denote the maximum share of a resource required by user j to process one job; then $\mu_j x_j$ is user j's dominant share.

We introduce the fairness measures $f_{\beta,\lambda}^{\text{FDS}}$:

$$\text{sgn}(1-\beta)\left(\sum_{j=1}^{n}\left(\frac{\mu_j x_j}{\sum_{k=1}^{n}\mu_k x_k}\right)^{1-\beta}\right)^{\frac{1}{\beta}}\left(\sum_{j=1}^{n}\mu_j x_j\right)^{\lambda}. \tag{3}$$

These fairness measures extend those developed in [4] for a single resource; details on their derivation are given in that work and the technical report [8]. Here $\beta \neq 1$ and λ are pre-specified parameters. Note that $\beta = 1$ is a trivial case, since (3) then reduces to $n\left(\sum_{j=1}^{n}\mu_j x_j\right)^{\lambda}$, so that each allocation gives equal fairness. We make a standard assumption that all resources and all jobs are infinitely divisible, which is typical of many multi-resource settings [15], [16]. An illustrative example of FDS is given in Section III-B3.

The fairness function (3) may be divided into two components, one representing fairness and one efficiency. The sum of the dominant shares raised to the power λ represents efficiency; thus, λ parametrizes efficiency's relative importance.

The remainder of (3) is parametrized exclusively by β and represents the allocation's fairness. It is easily seen that for any value of $\beta \neq 1$, this component of (3) is maximized at an equal allocation. However, the orderings of unequal allocations will depend on the value of β. If allocation A is more fair than allocation B when $\beta = \beta_1$ is used to parametrize fairness, allocation B may be more fair when $\beta = \beta_2 \neq \beta_1$ is used.

Though different values of β give different types of fairness, we can generally say that "larger β is more fair." As $\beta \to \infty$, we obtain max-min fairness on the ratio of each user's dominant share to the sum of all the dominant shares.

As $\beta \to \infty$ and $\lambda \to -1$, the fairness function $f_{\beta,\lambda}$ approaches max-min fairness on the dominant shares. Dominant resource fairness (DRF), proposed in [1], is thus a special case of FDS. In our notation, DRF can be expressed as

$$\min\left\{\mu_1 x_1,\ \mu_2 x_2, \ldots, \mu_n x_n\right\}. \tag{4}$$

Maximizing this equation subject to the constraints $\sum_{j=1}^{n} R_{ij}x_j \leq C_i,\ \forall i$, yields the DRF-optimal allocation. FDS is therefore a generalization of DRF, in which choosing the parameters β and λ allows one to achieve different tradeoffs between fairness and efficiency.

FDS also includes the well-known α-fairness family of functions as a special case. This fact easily follows from the relationship of the single-resource functions in [4] to α-fairness, which is generally used to measure fairness in bandwidth allocation (see references in Section II). Taking $\alpha = \beta \geq 0$ and $\lambda = \frac{1-\beta}{\beta}$, the FDS function (3) becomes

$$\text{sgn}(1-\beta)\left(\sum_{i=1}^{n}(\mu_i x_i)^{1-\beta}\right)^{\frac{1}{\beta}}; \tag{5}$$

optimizing this function is equivalent to optimizing the α-fairness function on dominant shares

$$\sum_{j=1}^{n}\frac{(\mu_j x_j)^{1-\alpha}}{1-\alpha}. \tag{6}$$

2) Generalized Fairness on Jobs (GFJ): Since some users require more resources per job than others, it might be more fair for those who require more resources to be allocated fewer jobs, thus increasing efficiency across all users. FDS captures this perspective. However, an individual user often cares only about the number of jobs processed (without accounting for heterogeneous resource requirements), and hence each user's notion of fairness may be based only on the number of jobs she is allocated. This motivates us to introduce another fairness measure called Generalized Fairness on Jobs (GFJ), which uses only the number of jobs allocated in the fairness function.

GFJ can be further motivated with bandwidth allocation examples. The utility function used in these scenarios is generally α-fairness applied to the bandwidth allocated to each flow. These functions are therefore a special case of GFJ, a family of functions given by

$$f_{\beta,\lambda}^{\text{GFJ}} = \text{sgn}(1-\beta)\left(\sum_{j=1}^{n}\left(\frac{x_j}{\sum_{k=1}^{n}x_k}\right)^{1-\beta}\right)^{\frac{1}{\beta}}\left(\sum_{k=1}^{n}x_k\right)^{\lambda}. \tag{7}$$

Here β and λ are two parameters (just as in FDS) and x_j is the number of jobs processed for user j. As for FDS, we have the resource constraints $\sum_{j=1}^{n} R_{ij}x_j \leq C_i$ for each resource i. An illustrative example is given in the next section.

For $\beta > 0$ and $\lambda = \frac{1-\beta}{\beta}$, GFJ reduces to α-fairness on the number of jobs allocated to each user.

3) Differences between FDS and GFJ: We can summarize FDS' and GFJ's approaches as follows:

- FDS measures fairness in terms of the relative size of the dominant shares, explicitly accounting for heterogeneous resource requirements in both the objective function and the constraints. As a limiting case of FDS, DRF also follows this approach.
- On the other hand, GFJ measures fairness only in terms of the number of jobs allocated to each user; the heterogeneity in resource requirements only appears in the resource constraints. Users requiring more resources are thus treated equally, a result observed in Section V.

When $\mu_j = \mu$ for all j, FDS and GFJ are equivalent.

Revisiting the example in the Introduction, we have the resource constraints $2x_1 + 2x_2 \leq 6$ and $3x_1 + x_2 \leq 4$. Thus, the dominant share of user 1 is $\frac{3}{4}x_1$, since user 1 requires $\frac{3}{4}$ of the available CPUs and $\frac{1}{3}$ of the available memory for each job. Similarly, the dominant share of user 2 is $\frac{1}{3}x_2$, since user 2 requires $\frac{1}{3}$ of the available memory and $\frac{1}{4}$ of the available CPUs for each job. FDS and GFJ can then be expressed as

$$\max_{x_1,x_2} f(x_1,x_2) \quad \text{s.t. } 2x_1 + 2x_2 \leq 6,\ 3x_1 + x_2 \leq 4, \tag{8}$$

where the fairness functions for FDS and GFJ are respectively

$$f = \text{sgn}(1-\beta)\left(\frac{\left(\frac{3x_1}{4}\right)^{1-\beta} + \left(\frac{x_2}{3}\right)^{1-\beta}}{\left(\frac{3x_1}{4} + \frac{x_2}{3}\right)^{1-\beta}}\right)^{\frac{1}{\beta}} \left(\frac{3x_1}{4} + \frac{x_2}{3}\right)^{\lambda}$$

$$f = \text{sgn}(1-\beta)\left(\frac{x_1^{1-\beta} + x_2^{1-\beta}}{(x_1+x_2)^{1-\beta}}\right)^{\frac{1}{\beta}} (x_1+x_2)^{\lambda}.$$

Figure 3 illustrates the approaches to multi-resource fairness. Each row of the matrix shown represents one user's resource requirements. One simplistic approach would assume perfectly substitutable resources; in that case, this matrix immediately collapses into a vector of users' single resource requirements. However, resources are often not directly substitutable, e.g. CPUs and memory.

FDS and GFJ represent alternative approaches to the scalarization of each row in Fig. 3's matrix. FDS and its limiting case DRF choose a dominant entry from the row vector of users' requirements. GFJ, on the other hand, scalarizes each row by the number of jobs processed with a bundle of different resources. These row-by-row scalarizations then yield another vector of users' scalars; evaluating fairness with $f_{\beta,\lambda}^{\text{FDS}}$ or $f_{\beta,\lambda}^{\text{GFJ}}$ further reduces this vector to a final scalar quantifying fairness.

IV. Properties of FDS and GFJ

In this section, we prove key properties of the FDS and GFJ functions introduced above. Section IV-A characterizes

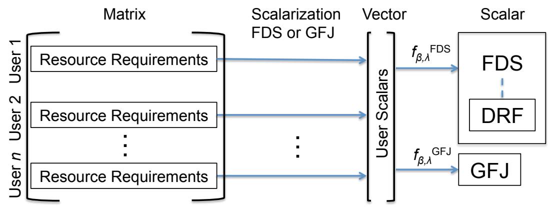


Fig. 3. Overall schematic of our multi-resource fairness approach.

the optimal fairness values in certain special cases, while Section IV-B examines the conditions of β and λ under which FDS and GFJ satisfy important properties relevant to fairness quantification and fairness-efficiency tradeoffs:

- What happens to the optimal allocations when users have the same resource requirements?
- What fairness properties do FDS and GFJ satisfy? For instance, are their optimal allocations Pareto-efficient? Sharing incentive compatible? Envy-free?
- Does there always exist a fairness-efficiency tradeoff?

Finally, Section IV-C examines the conditions under which a fairness-efficiency tradeoff exists.

We consider n users and m different resources. Users have the same resource requirements when they are homogeneous, or their heterogeneity is zero. If $n = 2$ or $m = 2$, user heterogeneity may be visualized as in Fig. 2 in Section III-A.

A. Values of FDS and GFJ

Heterogeneity is measured by the variance in the slopes σ_i or τ_i of Fig. 2. When all of users have the same ratios of multi-resource requirements (i.e., the variance of the $\{\sigma_i\}$ and $\{\tau_i\}$ is zero), the problem reduces to that of a single resource:

Proposition 1 (Reduction to Single-Resource Case): Suppose that the resource constraints may be written as

$$\eta_i\left(\mu_1 x_1 + \mu_2 x_2 + \ldots + \mu_n x_n\right) \leq 1, \tag{9}$$

$i = 1, 2, \ldots, m$. Let $\eta_{\max} = \max_i \eta_i$. Then the problem reduces to single-resource fairness on resource 1. Moreover, FDS and DRF both yield the allocation $x_j = (\eta_{\max}\mu_j n)^{-1}$. GFJ yields the allocation $x_j = \left(\mu_j^{-\frac{1}{\beta}}\right) \Big/ \left(\eta_{\max}\sum_{i=1}^{n}\mu_i^{\frac{\beta-1}{\beta}}\right)$.

Definition 1 (Efficiency): Let $X = x_1 + x_2 + \ldots + x_n$ denote the allocation efficiency.

In this special case, we also have the following corollary:

Corollary 1: For allocations that maximize DRF and FDS,

$$\frac{\partial X}{\partial \mu_j} = -\left(n\eta_{\max}\right)^{-1}\mu_j^{-2}$$

and the efficiency of these allocations increases the fastest if $\min_j \mu_j$ is decreased. For allocations that maximize GFJ,

$$\frac{\partial X}{\partial \mu_j} = \frac{-\mu_j^{-\frac{1+\beta}{\beta}}}{\eta_{\max}\beta\sum_{i=1}^{n}\mu_i^{\frac{\beta-1}{\beta}}} + \frac{(1-\beta)\mu_j^{-\frac{1}{\beta}}\sum_{i=1}^{n}\mu_i^{-\frac{1}{\beta}}}{\eta_{\max}\beta\left(\sum_{i=1}^{n}x_i^{\frac{\beta-1}{\beta}}\right)^2}.$$

In other words, the system's efficiency will increase if the user with the lowest μ_j gives up some resources.

We now consider heterogeneous users, and assume that their resource requirements R_{ij} are uniformly distributed in $[0, \nu C_i]$ for some $\nu > 0$. Then, as the number of users $n \to \infty$, the optimal FDS and GFJ values converge as follows:

Proposition 2 (Optimal FDS and GFJ Values): The optimal FDS value converges in probability as

$$\lim_{n\to\infty} \left(\max f^{\text{FDS}}_{\infty,-1}\right)^{-1} \cdot \frac{2m}{n(m+1)} = 1. \tag{10}$$

Thus, users' asymptotic dominant share is $\frac{1}{n} \cdot \frac{2m}{m+1}$. In contrast, the optimal GFJ value converges in probability as

$$\lim_{n\to\infty} \left(\max f^{\text{GFJ}}_{\infty,-1}\right)^{-1} \cdot \frac{2}{\nu\left(\sqrt{mn/3} + n\right)} = 1. \tag{11}$$

Users are asymptotically allocated resources for $\frac{2}{\nu n}$ jobs. We note that ν appears in (10) but not (11), since FDS uses dominant shares to evaluate fairness. Scaling the resource requirements by ν will scale the optimal allocations by ν^{-1}; these cancel in calculating the dominant shares $\mu_j x_j$.

We thus see that in the limit of a large number of heterogeneous users, with $\beta = \infty$ and $\lambda = -1$, the optimal FDS value increases while the optimal GFJ value decreases as more resources are added to the system. This proposition highlights the fundamental difference between FDS and GFJ.

B. Three Key Properties of Fairness

We next turn our attention to fairness and its relationship with efficiency, using three widely-used properties of fairness functions (see e.g., [1] and the many references therein):

Definition 2: A function f is **Pareto-efficient** if, whenever $\mathbf{x}$ Pareto-dominates $\mathbf{y}$ (i.e., $x_i \geq y_i$ for each index i, and $x_j > y_j$ for some j), $f(\mathbf{x}) > f(\mathbf{y})$.

Definition 3: **Sharing incentive** is the property that no user's dominant share is less than $\frac{1}{n}$: each user has an incentive to not simply split the resources equally.

Definition 4: **Envy-freeness** holds if no user envies another user's allocation. User j envies user k if $\gamma_{ik} x_k \geq \gamma_{ij} x_j$ for all resources i, with at least one strict inequality.

We investigate if and when these properties are satisfied by FDS and GFJ. Our results show that the answer depends on several factors, e.g. the values of the parameters β and λ.

We first consider Pareto-efficiency, which evidently holds for large λ. Based on [4], we can in fact specify a threshold for λ above which Pareto-efficiency holds:

Proposition 3 (Pareto-efficiency for FDS and GFJ): The fairness functions (3) and (7) are Pareto-efficient if and only if $|\lambda| \geq |(1-\beta)/\beta|$.

The absolute value signs are necessary, as for $\beta > 1$, (3) and (7) are negative. For this range of β, a more negative λ therefore emphasizes efficiency. As Pareto-efficiency is a highly desirable property for fairness functions (both single and multi-resource), the following analysis considers only values of λ satisfying $|\lambda| \geq |(1-\beta)/\beta|$.

Proposition 4 (Sharing Incentive for FDS): Sharing incentive is satisfied by the FDS-optimal allocation when $\lambda = (1-\beta)/\beta$ and $\beta > 1$. For $0 \leq \beta \leq 1$ and $\lambda = (1-\beta)/\beta$, sharing incentive may not be satisfied.

We can further bound the allocation efficiency:

Corollary 2 (Bounds on Allocation Efficiency for FDS): If $\beta > 0$ and $\lambda = (1-\beta)/\beta$, the efficiency $X \geq (\max_j \mu_j)^{-1}$.

In contrast to FDS, GFJ need not always satisfy sharing incentive even for $\beta > 1$:

Corollary 3 (Sharing Incentive for GFJ): If exactly one resource constraint $\sum_{j=1}^{n} R_{ij} x_j \leq C_i$ is tight at optimality and for any $\beta > 0$, $\lambda = (1-\beta)/\beta$, GFJ may not satisfy the sharing incentive property.

For $\lambda = \frac{1-\beta}{\beta}$, the FDS function becomes equivalent to the isoelastic α-fair utility in economics; then β corresponds to a measure of constant relative risk-aversion for individual users.[2] As β increases, individual risk-averse users find the resource allocation more equitable and become collectively envy-free. The following proposition establishes that this interesting envy-free behavior emerges (for FDS) at a *threshold* of $\beta > 1$.

Proposition 5 (Envy-freeness for FDS): For $\beta > 0$ and $\lambda = (1-\beta)/\beta$, i.e., the FDS function is the α-fair utility function with $\alpha = \beta$, envy-freeness holds if $\beta > 1$.

In contrast, GFJ-optimal allocations need not be envy-free for any value of β:

Corollary 4 (Envy-Freeness for GFJ): For any $\beta > 0$ and $\lambda = (1-\beta)/\beta$, envy-freeness may not be satisfied.

C. Fairness-Efficiency Tradeoff

We now consider two ways in which a fairness-efficiency tradeoff does not exist: first, an increased emphasis on fairness need not decrease efficiency. Second, the efficiency-maximizing allocation may also be the "most fair."

Traditionally, a larger parameter α in α-fairness functions is thought to be "more fair;" this statement is made mathematically precise in [4]. In [11], however, it is shown that when a network allocates bandwidth so as to maximize α-fairness, total throughput in the network may increase with α and may even decrease as capacity increases. These "counter-intuitive" results hold in the general multi-resource problem:

Consider a family of utility functions $U(\mathbf{x}, \alpha)$; here α is a parameter indexing the family of functions, and the specific functional form of U is not specified. For instance, we could use the functions in (3) with $\alpha = \beta$ and $\lambda = (1-\beta)/\beta$, i.e.,

[2] **Isoelasticity** and **relative risk-aversion** in economics are defined as $\frac{\partial u(x)}{\partial x} \frac{x}{u(x)}$ and $-\frac{x u''(x)}{u'(x)}$ respectively, where u is the utility function.

"α-fair" utility functions. We incorporate the resource capacity constraints in the matrix inequality $\mathbf{Rx} \leq \mathbf{C}$, and assume that $\mathbf{R}$ is a full-rank matrix consisting only of constraints which are tight at the optimal allocation for the given value of α.

We let $\mathbf{S}$ be an $(n-m) \times n$ dimensional matrix whose columns form a basis for the nullspace of $\mathbf{R}$. The negative of the utility function's Hessian matrix is denoted by $\mathbf{D}$, and we define $\mathbf{b} = \frac{\partial^2 U}{\partial x \partial \alpha}$, $\mathbf{A} = \mathbf{S}^T\mathbf{DS}$, $\mathbf{v}_j = \mathbf{s}_j{}^T\mathbf{b}$ and $\beta_j = -\mathbf{1}^T\mathbf{s}_j$, where the $\mathbf{s}_j$ are the columns of the matrix $\mathbf{S}$. Let $\overline{\mathbf{A}}_i$ denote the matrix $\mathbf{A}$ with the ith row replaced by $\beta = [\beta_1\ \beta_2 \cdots \beta_n]$. We use δ to denote a direction of perturbation of the capacity vector $\mathbf{C}$ and $\mathbf{D}X(\delta)$ to denote the derivative of X in the direction of δ. From [11], we have

$$\frac{\partial X}{\partial \alpha} = \mathbf{1}^T\mathbf{S}\mathbf{A}^{-1}\mathbf{S}^T\mathbf{b} \tag{12}$$

$$\mathbf{D}X(\delta) = \mathbf{1}^T\frac{\partial x}{\partial C}\delta = \mathbf{1}^T\mathbf{D}^{-1}\mathbf{R}^T(\mathbf{R}\mathbf{D}^{-1}\mathbf{R}^T)^{-1}\delta. \tag{13}$$

We can further prove the following proposition:

Proposition 6 (Efficiency Non-Monotonicity): Efficiency increases with α if and only if

$$\sum_{i=1}^{N-L} \mathbf{v}_i \det \overline{\mathbf{A}}_i \geq 0. \tag{14}$$

Moreover, efficiency may decrease with an increase in the capacity vector $\mathbf{C}$. If capacity increases proportionally, i.e., $\delta = \epsilon\mathbf{C}$ for some small ϵ, then $\mathbf{D}X(\delta) \geq 0$.

As a special case, when only one capacity constraint is tight (e.g., one resource), efficiency always increases with capacity. The technical report [8] contains a numerical example in which efficiency increases with β.

We next examine the conditions under which an equal allocation (equal dominant shares for FDS or an equal number of jobs for GFJ) maximizes efficiency. In such scenarios, there is no fairness-efficiency tradeoff; the most fair allocation maximizes the total number of jobs processed. As this property is an ideal case, it will likely be satisfied only under rather stringent conditions. Indeed, our results show that this ideal case occurs only when the resource constraints "line up."

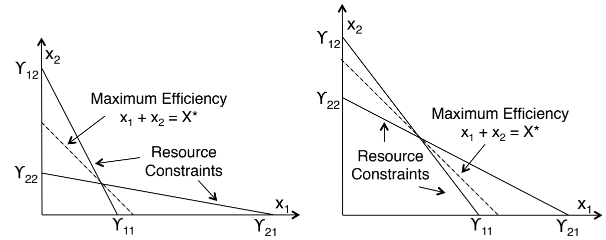
Proposition 7 (Maximizing Fairness and Efficiency (I)): Suppose that $m = n$ constraints are tight at the maximum-efficiency allocation. Then this allocation equalizes the dominant shares (maximizing FDS fairness) if and only if all resources i satisfy

$$\sum_{j=1}^{n} \frac{\gamma_{ij}}{\mu_j} = \rho \tag{15}$$

for some constant ρ. The number of jobs per user is equalized (maximizing GFJ fairness) if

$$\sum_{j=1}^{n} \gamma_{ij} = r \tag{16}$$

for some constant r and all resources i.



(a) FDS fairness. (b) GFJ fairness.

Fig. 4. Illustration of Prop. 7 for two resources. In the left graph, the FDS-maximizing allocation maximizes efficiency and equalizes dominant shares. In the right graph, the GFJ-maximizing allocation maximizes efficiency and equalizes the number of jobs processed.

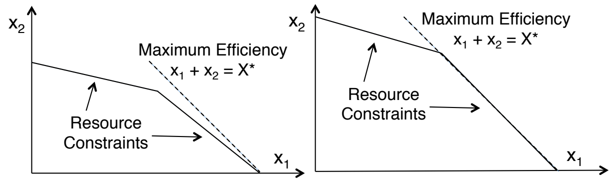


Fig. 5. Illustration of Prop. 8 in two dimensions. In the left graph, $x_2 = 0$ at the unique efficiency-maximizing allocation. In the right graph, multiple efficiency-maximizing allocations exist.

Figure 4 illustrates these conditions in two dimensions. In the left figure 4a, dominant shares are equal at the efficiency-maximizing allocation, while in the right figure 4b the numbers of jobs are equalized at the efficiency-maximizing allocation.

Our conclusions are more subtle when $m < n$ constraints are tight at an efficiency-maximizing allocation:

Proposition 8 (Maximizing Fairness and Efficiency (II)): Suppose that $m < n$ constraints are tight at an efficiency-maximizing allocation $\mathbf{x}^*$. If this allocation is the unique allocation maximizing efficiency, then at least one of the $x_j^* = 0$ and one user is allocated no jobs. If other allocations also maximize efficiency, an allocation equalizing either the dominant shares or number of jobs processed maximizes efficiency if and only if at the equal allocation, the constraint set intersects the hyperplane $\sum_{j=1}^{n} x_j = \sum_{j=1}^{n} x_j^*$ on a set of dimension at least 1.

Figure 5 illustrates these statements in two dimensions. The left graph shows a unique efficiency-maximizing allocation with only 1 tight resource constraint, and the bottom graph shows a set of multiple efficiency-maximizing allocations.

We can use this proposition to derive a sufficient condition for the efficiency-maximizing allocation to equalize the dominant shares or number of jobs for each user:

Corollary 5: Suppose $m < n$ resource constraints hold at the efficiency-maximizing allocation. Then if $R_{ij} > R_{ik}$ for some users j and k and all resources i, $x_j = 0$ (user j is allocated no jobs) at any efficiency-maximizing allocation.

If $m = 1$ (the single-resource case), this result implies the following:

Corollary 6: The maximum efficiency allocation equalizes the dominant shares (FDS) or jobs per user (GFJ) if and only if $\mu_j = \mu \; \forall$ users j. In other words, each user needs the same amount of the single resource to process one job.

V. Applications and Illustrations

We consider an illustrative example of a datacenter with CPU and RAM constraints. There are two users, each of whom requires a fixed amount of each resource to accomplish a job. Jobs are assumed to be infinitely divisible [15], [16]. In order to benchmark performance, we use the same parameters as [1]: user 1 requires 1 CPU and 4 GB of RAM for each job, and user 2 requires 3 CPUs and 1 GB for each job. At first, we assume 9 CPUs and 18 GB of RAM; we later vary these constraint values to observe their impact on fairness.

Suppose that the fairness function is given by f (e.g. FDS (3), DRF (4), GFJ (7)). Then the allocation problem is

$$\max_{x,y} \; f(x,y) \quad \text{s.t. } x + 3y \leq 9, \; 4x + y \leq 18 \tag{17}$$

where x and y are the number of jobs allocated to users 1 and 2 respectively.

We use DRF as the benchmark fairness to compare the performance of our FDS and GFJ functions. We define **percent fairness** as the percentage difference between the optimal DRF fairness value (i.e., the minimum dominant share) and the DRF fairness value of the allocation obtained from FDS or GFJ. The **percent efficiency** is defined as the percentage difference between the total number of jobs processed in the given allocation and the maximum number of jobs that can be processed, given the same capacity constraints. We also introduce another efficiency measure, the **leftover capacity** (i.e., the amount of unused resources).

We investigate the outcomes of the proposed fairness measures along two dimensions:

- Comparing the achieved efficiency when user heterogeneity and resource capacity are varied.
- Examining the range of attainable fairness-efficiency tradeoffs for different values of the parameters β and λ.

A. Efficiency

We first use our two efficiency measures–leftover capacity and percent efficiency–to investigate user heterogeneity's effect on achieved efficiency. Heterogeneity is measured by the variance in the slopes τ_i and σ_i of users' resource requirements, as introduced in Section III-A's Fig. 2. If two users have identical resource requests, they become homogeneous, and both variances are 0. At the other extreme, the users do not share any resource requirements and the variances are infinite.

We calculate the optimal FDS, GFJ and DRF allocations for $\beta = 2$, $\lambda = -0.5$. First, Fig. 6 examines the leftover capacity as a function of the variance in τ. The heterogeneity was varied by changing user 2's RAM requirement from 1 GB to 13 GB. Thus, the RAM constraint line in Fig. 2's representation tilts from very steep to very flat. This tilting geometrically explains

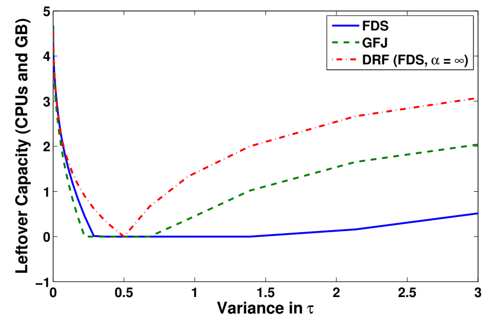


Fig. 6. *Too much or too little variance in τ leads to inefficiency from leftover capacity:* Leftover capacity versus variance in user heterogeneity in a datacenter example. Variances below 0.5 have only leftover CPUs; variances above 0.5 have only leftover RAM.

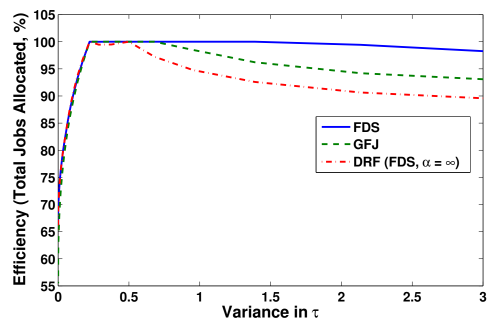


Fig. 7. *Greater variance in τ leads to DRF inefficiency in the number of jobs processed:* Percentage efficiency versus variance in user heterogeneity.

the overall "V" trend in Fig. 6. When the RAM requirement is below 3 GB (a steep constraint line), the variance of τ is over 0.5: only RAM is leftover. When the RAM requirement is above 3 GB (a flatter line), the variance of τ is less than 0.5: only CPUs are leftover. The change in the leftover resource is due to the changing shape of the feasible region.

In this example, we see that for low heterogeneity in users' resource requirements, FDS, GFJ and DRF have similar efficiency values. In fact, we confirm Prop. 1's result that at zero heterogeneity, DRF and FDS are optimized at the same allocation. As heterogeneity increases in Fig. 6, DRF has significant leftover capacity compared to GFJ and FDS. DRF trades off efficiency to preserve users' minimum dominant share with increasingly heterogeneous resource requirements. Even GFJ performs worse than FDS, which yields the lowest leftover capacity. As FDS includes resource requirements in its fairness function, we intuitively expect such a result.

We next examine the percent efficiency in jobs processed as a function of the variances in τ in Fig. 7. As in the previous figures, for low heterogeneity in users' resource requirements, FDS, GFJ and DRF perform at similar efficiency levels. All three achieve full efficiency for a τ variance near 0.5. Again, the efficiency attained is much higher (about 15%) for FDS and GFJ than for DRF as the variance increases. Similar plots for the variance of σ are shown in the technical report [8].

In summary, enforcing DRF can significantly reduce efficiency as measured by either leftover capacity or percent efficiency. In the technical report [8], we discuss a scenario in which the number of users grows and their corresponding τ_i

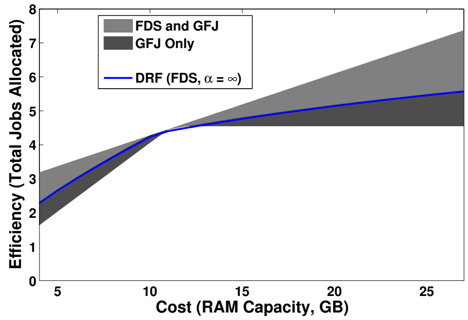


Fig. 8. *Capacity expansion can increase the range of FDS and GFJ operating efficiencies over DRF:* Attainable efficiency for varying capacity constraint, given different implicit realizations of $\beta \in (-5, 5)$ and $\lambda \in (0.01, 1.91)$ for $\beta < 0$, $\lambda \in \left(0.005\left(\beta^{-1}-2\right), 0.955\left(\beta^{-1}-2\right)\right)$ for $\beta > 0$.

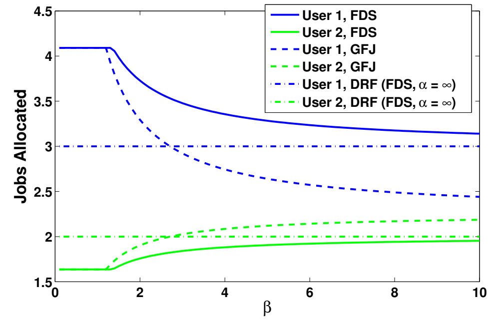


Fig. 9. *Larger β values lead to more equitable allocations:* Optimal allocations for various fairness measures, using $\alpha = \beta$ fairness for FDS and GFJ.

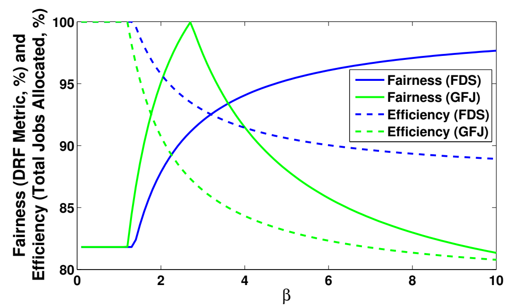


Fig. 10. *The fairness-efficiency tradeoff can be tuned by changing β:* Percentage of fairness and efficiency achieved for various fairness measures, using $\alpha = \beta$ fairness for FDS and GFJ.

are uniformly distributed. In this situation, the FDS-optimal allocation's efficiency becomes even more desirable.

Finally, we examine the impact of changing RAM capacity on the attainable efficiency, as shown in Figure 8. We see that when the dominant shares for both users are equal, at $6\sqrt{3}$ GB of RAM capacity, GFJ and FDS have the same range of achievable efficiency. In general, β and λ may be chosen so that FDS and GFJ are more efficient than DRF; in fact, the FDS-optimal allocation is always more efficient than DRF.

The impact of capacity expansion also highlights an interesting dimension of the *economy of scale* in large networks. The standard view is that a large scale helps smoothen temporal demand fluctuations through statistical multiplexing, e.g., at an aggregation point in a broadband access network. In addition to temporal "heterogeneity" (bursting at different times), network users may have resource type heterogeneity: some applications need more CPU processing while others need more storage or bandwidth. Can this heterogeneity be exploited to more efficiently utilize different types of resources? The answer depends on how these different resources are allocated among the users. If DRF is used, for example, efficiency can be quite low. However, the appropriate FDS parametrization can indeed leverage user heterogeneity along with increases in resource capacity, creating another type of economy of scale.

B. Fairness-Efficiency Tradeoffs

The previous section established that when users are very heterogeneous, FDS and GFJ outperform DRF, achieving a much greater efficiency. However, we expect that this larger efficiency comes at a cost of decreased fairness. This section examines the general behavior of fairness when a larger efficiency is achieved. Here we measure fairness as percent fairness with the DRF metric and efficiency as percent efficiency on the number of jobs processed.

Figure 9 shows the optimal job allocations for different values of β, $\lambda = \frac{1-\beta}{\beta}$. FDS and GFJ become α-fair on the dominant shares of and jobs allocated to each user, respectively, for $\alpha = \beta$. As β increases, λ decreases, so that fairness is emphasized more than efficiency and FDS asymptotes to DRF. For small β (more relative emphasis on efficiency), the optimal FDS allocation maximizes efficiency. In the case of GFJ, which emphasizes the fairness on jobs allocated, larger β values yield a more fair allocation of jobs across users than FDS, as expected. Thus, the total number of jobs processed (i.e., efficiency) is lower for GFJ than for FDS.

Figure 10 gives a representative plot of how this tradeoff varies with β, $\lambda = \frac{1-\beta}{\beta}$. As β increases, the percent efficiency from the FDS measure drops, approaching DRF as $\beta \to \infty$. GFJ fairness increases until $\beta = 2.6$, at which point the GFJ-optimal allocation is also DRF-optimal. (In Fig. 9, the GFJ and DRF allocations intersect at $\beta = 2.6$). For larger values of β, GFJ quickly converges to an allocation with a more equal number of jobs per user, decreasing efficiency. Efficiency in FDS decreases more slowly since FDS attempts to make the dominant shares, not the number of jobs, more equitable.

Finally, we show the interaction between capacity constraints and the range of fairness-efficiency tradeoffs achieved. The shaded region in Fig. 11 shows the attained tradeoffs for a large range of β and λ values; each point corresponds to some β and λ values in the FDS function, which achieve the shown operating tradeoff. This achieved tradeoff depends on the available capacity, with contour lines for various RAM capacities shown in the figure. As RAM capacity increases from 4 GB to $6\sqrt{3}$ GB, the tradeoff stopped: one can increase both fairness and efficiency. At a RAM capacity of $6\sqrt{3}$ GB, the conditions of Prop. 7 are satisfied, and efficiency is maximized when the dominant shares are equal. When the RAM capacity goes above $6\sqrt{3}$ GB up to 25 GB, user 1's dominant share $\frac{4x_1}{\text{RAM capacity}}$ decreases. Thus, an increase in fairness requires an increase in x_1 and user 1's CPU allocation. User 2 is then allocated fewer jobs, decreasing efficiency. In

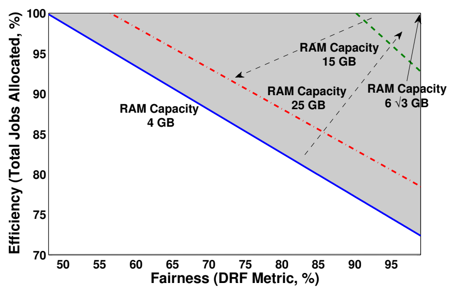


Fig. 11. *Capacity expansion allows different FDS fairness-efficiency tradeoff contours:* Attainable efficiency vs. fairness tradeoffs from different implicit realizations of $\beta \in (-5, 5)$ and $\lambda \in (0.01, 1.91)$ for $\beta < 0$, $\lambda \in \left(0.005\left(\beta^{-1} - 2\right), 0.955\left(\beta^{-1} - 2\right)\right)$ for $\beta > 0$ values. DRF is used as the fairness benchmark and metric.

this figure, one can achieve 100% efficiency and fairness when RAM capacity is $6\sqrt{3}$ GB, but such an ideal operating point does not always exist. An analogous plot for GFJ functions is shown in the technical report [8].

VI. Future Work

Some preliminary work shows that both FDS and GFJ can be unified into a single framework. The idea is to use a p-norm function $g(\gamma_{1,j}, \ldots, \gamma_{n,j}) = \left(\sum_i \gamma_{i,j}^p\right)^{\frac{1}{p}}$ to scalarize the resource requirement vector of user j, and then evaluate the resulting fairness by $f_{\beta,\lambda}$. This method leads to a new family of fairness measures, parameterized p, β, and λ, i.e.,

$$f_{p,\beta,\lambda} = \text{sgn}(1-\beta)\left(\sum_{j=1}^{m}\left(\sum_{k=1}^{n} R_{kj}^p\right)^{\frac{1-\beta}{p}} x_j^{1-\beta}\right)^{\frac{1}{\beta}} \times \left(\sum_{j=1}^{m}\left(\sum_{k=1}^{n} R_{kj}^p\right)^{\frac{1}{p}} x_j\right)^{\lambda+1-\frac{1}{\beta}} \quad (18)$$

Fairness $f_{p,\beta,\lambda}$ includes many fairness measures as special cases. It is easy to verify that $f_{0,\beta,\lambda} = f_{\beta,\lambda}^{GFJ}$ and $f_{\infty,\beta,\lambda} = f_{\beta,\lambda}^{FDS}$. Further, $f_{1,\beta,\lambda}$ gives the system's total resource usage.

This function again satisfies the four axioms of [4], as do FDS and GFJ. Moreover, Pareto-efficiency is satisfied for $|\lambda| \geq \left|\frac{1-\beta}{\beta}\right|$. We expect that, in analogy with Props. 4-5 and their corollaries, conditions on p, β and λ can be found, under which sharing incentive and envy-freeness are satisfied.

VII. Concluding Remarks

In this paper, we introduce FDS and GFJ, two families of fairness functions for multi-resource allocations. FDS includes as a special case the recently proposed generalization of max-min fairness to multiple resources. Different parameterizations of these functions generate a range of fairness-efficiency tradeoffs, thus allowing for different degrees of emphasis on fairness and efficiency for different network operation needs.

We consider three key properties of fairness functions: Pareto-efficiency, sharing incentive, and envy-freeness. FDS and GFJ are both Pareto-efficient if $|\lambda| \geq \frac{1-\beta}{\beta}$. Neither FDS nor GFJ always satisfies sharing incentive or envy-freeness for $0 < \beta < 1$, $\lambda = \frac{1-\beta}{\beta}$. For any $\beta > 1$ and $\lambda = \frac{1-\beta}{\beta}$, sharing incentive and envy-freeness are always satisfied for FDS but may or may not be satisfied for GFJ.

Finally, we discuss a less theoretical direction of future work: estimating the β and λ values which correspond to people's preferences. We are exploring methods to estimate these parameters based on experimental results from human subjects. At present, we plan to ask subjects to rank given resource allocations in order of preference. The β and λ values calculated from these results can help give a physical meaning to these two parameters. Moreover, we plan to investigate if people give consistent results and if different demographics can be naturally grouped into different β and λ values.

Acknowledgment

This work was partly supported by NSF grant CNS-0905086, DARPA grant FA8750-11-C-0254, and AFOSR MURI grant FA9550-09-1-0643.

References

[1] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, and I. Stoica, "Dominant resource fairness: Fair allocation of multiple resource types," in *Proc. of the 8th USENIX Conf. on Networked Systems Design and Implementation*. USENIX Association, 2011, pp. 24–37.

[2] R. Jain, D. M. Chiu, and W. R. Hawe, *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System*. Eastern Research Laboratory, Digital Equipment Corp., 1984.

[3] K. J. Arrow, "The theory of risk aversion," *Essays in the Theory of Risk-Bearing*, pp. 90–120, 1971.

[4] T. Lan, D. Kao, M. Chiang, and A. Sabharwal, "An axiomatic theory of fairness in network resource allocation," in *Proc. of IEEE INFOCOM 2010*. IEEE, 2010, pp. 1–9.

[5] H. Varian, "Equity, envy, and efficiency," *Journal of Economic Theory*, vol. 9, no. 1, pp. 63–91, 1974.

[6] A. Odlyzko, "Network neutrality, search neutrality, and the never-ending conflict between efficiency and fairness in markets," in *Review of Network Economics*, vol. 8, no. 1, March 2009, pp. 40–60.

[7] M. Zukerman, L. Tan, H. Wang, and I. Ouveysi, "Efficiency-fairness tradeoff in telecommunication networks," in *IEEE Communications Letters*. IEEE, 2005, pp. 643–645.

[8] C. Joe-Wong, S. Sen, T. Lan, and M. Chiang, "Multi-resource allocation: Fairness-efficiency tradeoffs in a unifying framework," Princeton University, http://www.princeton.edu/∼chiangm/multiresourcefairness.pdf, Tech. Rep., 2011.

[9] S. K. Baruah, N. K. Cohen, C. G. Plaxton, and D. A. Varvel, "Proportionate progress: A notion of fairness in resource allocation," *Algorithmica*, vol. 15, no. 6, pp. 600–625, 1996.

[10] J. F. Nash, "The bargaining problem," *Econometrica*, vol. 18, no. 2, pp. 155–162, 1950.

[11] A. Tang, J. Wang, and S. H. Low, "Counter-intuitive throughput behaviors in networks under end-to-end control," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 355–368, 2006.

[12] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.

[13] R. Srinivasan and A. K. Somani, "On achieving fairness and efficiency in high-speed shared medium access," *IEEE/ACM Transactions on Networking*, vol. 11, no. 1, pp. 111–124, 2003.

[14] M. Zaharia, D. Borthakur, J. Sen Sarma, K. Elmeleegy, S. Shenker, and I. Stoica, "Delay scheduling: A simple technique for achieving locality and fairness in cluster scheduling," in *Proc. of the 5th European Conference on Computer Systems*. ACM, 2010, pp. 265–278.

[15] Y. Yang, "Rumr: Robust scheduling for divisible workloads," in *Proc. of the 12th IEEE Symposium on High Performance and Distributed Computing*. IEEE, 2003.

[16] M. Drozdowski. (2011, Jul.) Introduction to divisible tasks. [Online]. Available: http://www.cs.put.poznan.pl/mdrozdowski/divisible/divisible_intro/divisible_intro.html