Dynamic Fair Resource Division

Shai Vardi
Krannert School of Management, Purdue University, West Lafayette, IN. svardi@purdue.edu

Christos-Alexandros Psomas
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA. cpsomas@cs.cmu.edu

Eric Friedman
ICSI and Electrical Engineering and Computer Sciences Department, Berkeley, CA. ejf@icsi.berkeley.edu

A single homogeneous resource needs to be fairly shared between users dynamically arriving and departing over time. While good allocations exist for any fixed number of users, implementing these allocations dynamically is impractical: it typically entails adjustments in the allocation of *every* user in the system whenever a new user arrives. We introduce a dynamic fair resource division problem where there is a limit in the number of users that can be disrupted when a new user arrives, and study the trade-off between fairness and reallocation, using a fairness metric: the *fairness ratio*. We almost completely characterize this trade off, and an algorithm for obtaining the optimal fairness as a function of the amount of reallocation allowed. Finally, we give a number of instructive qualitative bounds on the fairness ratio.

Key words: fair division, dynamic resource allocation

---

1. Introduction

Resource division is a fundamental problem that arises in many shared systems. The most well-studied objective is arguably maximizing efficiency. In many applications of interest, however, the system must address more delicate issues, such as *fairness*. A canonical example is a software company such as Google, Microsoft and Facebook, that need to allocate their internal clusters in a fair way among their employees.

Traditionally, research on fair division has focused on static allocations. In practice, however, allocation protocols are usually dynamic, and must accommodate users that arrive and depart from the system over time. A major difficulty in dynamic resource allocation settings stems from the need to balance fairness, efficiency and reallocation. Any two are straightforward to achieve if the third is not required. As an example, consider a cloud computing platform that can support one million users; the users arrive one by one, and the platform wishes to allocate some resource such as RAM. If memory reallocation could be done instantly and without cost, it would be trivial to always maintain a fair and efficient allocation: whenever a user arrives, simply redistribute all of the RAM evenly. If fairness is of no concern, one can simply allocate all of the RAM to the first user that arrives. Finally, without concern for efficiency, the platform could simply allocate each user a millionth of its total RAM; a perfectly fair solution, and no reallocation is ever required. Naturally, if only there are only a few users present this leads to a substantial underutilization of the platform's resources.

In this paper we study the trade-off between fairness and reallocation in a single resource system that wishes to maintain a high level of efficiency (we expand on precisely what is meant by a "high level" below), with dynamically arriving and departing users. We almost completely characterize this trade off, and design algorithms for obtaining the optimal fairness as a function of the amount of reallocation allowed. Slightly more formally, our algorithms take as input a vector of allowed disruptions: a user is disrupted if the amount of resource allocated to her is reduced. We call such vectors *control vectors*. Our algorithms are asked to maximize fairness, while respecting the number of disruptions specified by the control vector. The fairness benchmark we compare to is *proportional* allocations: in the presence of k agents, each user is allocated a k-th of the resource. The goal of our algorithms is to maximize the *fairness ratio* — the ratio of the smallest share to the proportional share at any time. Note that proportionality also captures some notion of efficiency in this setting: a proportional allocation is fully efficient, as the entire resource is allocated. This is in contrast to other notions of fairness, such as envy-freeness and equitability, which are independent of efficiency. We expand upon this in Section 1.1.

Example 1. Consider a system with capacity for three users, where every time a new user arrives a single disruption is allowed. One possible algorithm divides the largest available share equally at each arrival. When the first user arrives, she is allocated the entire resource; when the second user arrives, she is given half of the resource, and the first user's resource is halved; when the third user arrives, only one user can be disrupted, and so one user is allocated half of the resource and the two other users are allocated one quarter of the resource each. To gauge the fairness of this algorithm, we compare each of the allocations with the proportional shares: when there are one or two users in the system, the allocations of the algorithm and the proportional shares are identical; when there are three users in the system, the proportional share would be one third, but the smallest share is one quarter, 3/4 of the proportional share. The fairness ratio of this algorithm is therefore 3/4. It is easy to verify that no algorithm can achieve a fairness ratio of 1 even in this simple scenario. We will see in Section 4 that the optimal algorithm in this toy example has a fairness ratio of 6/7. $\triangleleft$

Our main result is an optimal algorithm; that is, an algorithm that achieves the best possible fairness ratio for any input (control vector). To achieve this, we solve a closely related problem: given a control vector and a fairness ratio, output allocations that guarantee this fairness ratio for this control vector, or conclude that no such allocations exist. We design an algorithm for the second problem; our original problem now reduces to computing the optimal fairness ratio of the input control vector.

We show that if there is a limit on the number of users a system can accommodate (i.e., the control vector has finite size), it is possible to precisely compute the optimal fairness ratio, and hence to compute optimal allocations. In addition, we show that we can handle uncapped systems (i.e., ones without an a-priori upper bound on the number of users) as well. As long as the set of disruptions can be described succinctly (e.g., a single disruption is allowed whenever a new user arrives), we can compute the optimal fairness ratio, and the algorithm is asymptotically optimal. Our results and proofs give several insights to the limitations that disruption requirements (or alternatively fairness requirements) impose on systems.

The first insight is that a very small number of disruptions suffices to guarantee good fairness—five disruptions per arrival, for example, guarantee a fairness ratio of 0.914. In other words, the system can allocate the resource in a way that at every point in time, every user will always have at least 0.914 of the amount she would have received if the resource was always divided equally among the users present in the system, even for unbounded systems. The second insight is that there is a very small difference in the fairness of systems of different sizes (for a fixed number of disruptions per arrival). As an example, the fairness ratio of the algorithm that allows a single disruption per arrival in a system with capacity for one hundred users is 0.727, while for a system with one billion users it is 0.721. In addition, the system does not need to compute all of the allocations in advance: our algorithm can compute the allocations "on the fly", as users arrive and depart. This is especially important for very large or unbounded systems. Finally, we show that—as long as at least one disruption is allowed per arrival—we do not need to sacrifice efficiency for fairness: there always exist allocations that are Pareto efficient (i.e., always allocate the entire resource) and achieve the optimal fairness.

At first glance, it may appear that at least one reallocation per arrival is necessary in order to provide any fairness guarantee. However, even a single reallocation per arrival may be too much for real-world systems. We therefore consider the case with fewer disruptions than arrivals. For example, consider the case where the algorithm is allowed to disrupt a single user once every c arrivals, for $c > 1$. We show that our previous optimal algorithm (for at the case of at least on disruption per arrival) gives the optimal fairness ratio in this case as well. Unfortunately, it is not (and indeed we show that it cannot be) fully efficient.

Computing meaningful qualitative bounds on the fairness ratio for uncapped systems when there are fewer disruptions that arrivals is much more involved, despite the fact that we know an algorithm that can compute the optimal fairness for any finite control vector. This task becomes even more daunting if we allow some flexibility: if the algorithm is guaranteed that it will be allowed to disrupt a user at least once every three arrivals, for example, one could simply fix the input to the algorithm to be a repeating vector of (say) a one followed by two zeros: $(1, 0, 0, 1, 0, 0, 1, 0, 0, \ldots)$. This seems unnecessarily rigid: what if, when the fifth user arrives, it suddenly becomes convenient to perform a reallocation? It seems reasonable to allow the system to add a disruption here, as long as it does not disrupt more than one user every three arrivals. Furthermore, we would like to allow the system to make such decisions as the need arises, and not commit to the entire vector ahead of time. In order to accommodate these requirements, we need to bound the fairness ratio of *all* possible control vectors simultaneously. We compute almost matching upper and lower bounds on the fairness ratio of all control vectors that allow one disruption for every $c > 1$ users. Our analysis shows that the fairness ratio decays logarithmically with the number of arrivals one has to wait between disruptions. For example, allowing (at least) one disruption every five, fifty and five thousand arrivals gives fairness ratios of approximately 0.44, 0.25 and 0.12 respectively. This implies that even with very few disruptions, we can guarantee (surprisingly) large fairness ratio: even if we allow a single disruptions for every 20,000 arrivals, each user can still be guaranteed at least 1/10 of the amount of resource she would receive if there was no cost to reallocation. These bounds hold for all control vectors such that the algorithm is allowed to disrupt a single user once every c arrivals, for $c > 1$. Practically, this means that the algorithm does not need to know the control vector a-priori. In addition, we show that in many cases, the worst allocations do not occur when there are many users in the system. This means that allowing some flexibility at the beginning—adding a handful of additional disruptions to handle the first few arrivals—can lead to improved fairness ratios and efficiency.

1.1. Related Work

Fair division has been a central topic in Operations Research, e.g., (Bertsimas et al. 2013, Correa et al. 2007, Deng et al. 2012, Fishburn and Sarin 1994, Karsten et al. 2015, Thomson 1983), Economics (Budish 2011, Pazner and Schmeidler 1978, Steinhaus 1948), Management Science (Boiney 1995, Fishburn and Sarin 1994, Haitao Cui et al. 2007), Mathematics (Alon 1987, Brams and Taylor 1995) and Computer Science (Aziz and Mackenzie 2016, Othman et al. 2014).

There are several well-studied and accepted notions of fairness. The three most studied notions are arguably (1) *proportionality*: when k agents are present, each agent receives a piece worth at least $1/k$ of her valuation for the entire resource, (2) *envy-freeness*: no agent prefers any other agent's share to her own, and (3) *equitability*: the valuation of all agents for the share they receive is equal. In the static version of the problem studied in this paper (as the resource is homogeneous), all three notions are equivalent if one additionally asks for the allocation to be Pareto optimal.

Recently, fair division has received a surge of attention due to its applications to resource sharing in data centers and the cloud, e.g., (Bhattacharya et al. 2013, Dolev et al. 2012, Friedman et al. 2014, Ghodsi et al. 2011, Popa et al. 2012, Wang et al. 2014). This has led to more research on fairness in the dynamic setting, e.g., (Aleksandrov et al. 2015, Kash et al. 2014, Walsh 2011): there is some finite amount of resource(s), users arrive and depart, and the goal is to constantly maintain allocations that are "fair".

Walsh (2011) was the first to study the problem of online fair cake cutting when users arrive, receive a piece and depart. He showed how several well-known fair division solutions (cut-and-choose, Dubins-Spanier, etc) can be adapted to satisfy desirable properties in an online setting with a single (heterogeneous) divisible cake. More recently, and closer to the problem studied here, Kash et al. (2014) introduced a model of dynamic allocations. However, their model only considers arrivals and their main algorithm reserves resources for future arrivals; it does not allow the reallocation of resources, or users to depart. This leads to allocations that satisfy neither fairness nor Pareto efficiency, as resources are left idle. Subsequent to the present work, Li and Li (2018) provide partial answers to the problem of dynamically sharing a non-homogeneous resource. They show that algorithms that are allowed one disruption per arrival can still guarantee a logarithmically decaying fairness ratio.

Guo et al. (2009) studied the problem of repeatedly allocating a single item between competing users. They give allocation algorithms that do not allow monetary transfers, with good competitive ratios with respect to optimal allocation algorithms with payments. Segal-Halevi (2018) studied the problem of re-dividing a two-dimensional resource, subject to fairness and "geometric" constraints on the allocations. Isard et al. (2009) considered scheduling with locality and fairness constraints. They evaluate different algorithms with/without requiring fairness and with preemption enabled/disabled. They find that requiring fairness and allowing preemption gives the best overall performance regardless of the scheduler implementation that is used. Freeman et al. (2018) study the trade-off between strategyproofness, efficiency and fairness in a setting with dynamically changing preferences.

Many papers study the dynamic resource allocation without the restriction of fairness, e.g., (Ahmadi et al. 1992, Huh et al. 2013). Topaloglu and Powell (2005) study the dynamic allocation of indivisible resources to tasks from a multi-agent decision-making perspective, when the tasks arrive from some known distribution. Ciocan and Farias (2012) consider a different model in which there is volatile demand for resources, and the goal is to maximize revenue. While their model and results are completely disjoint from ours, they too point out that an unattractive solution to the dynamic version is simply to solve simple "offline" versions of the allocation problem at hand.

Benade et al. (2018) study a complementary setting where users are static and resources arrive over time. Finally, the networking community has studied the problem of fairly allocating a single homogeneous resource in a queuing model where each agent's task requires a (given) number of time units to be processed. In these models, even though tasks are processed over time, demands stay fixed and there are no other dynamics such as agent arrivals and departures. The well-known fair queuing solution (Demers et al. 1989) — which has been also analyzed by economists (Moulin and Stong 2002) — allocates one unit per agent in a successive round-robin fashion.

2. The Model

One unit of a homogeneous, infinitely divisible resource is shared among users that arrive and depart over time. We assume that users only arrive, and refer to the time period when there are t users in the system as time t; in Section 6 we show that this is without loss of generality, and all of the results extend to the case where users also depart. The amount of resource is normalized to be 1. Denote an allocation for t users by a vector $\mathrm{X}^t \in [0,1]^t$. For notational convenience, we assume that the vector is sorted in non-increasing order, and denote the j-th largest allocation of

X^t by X_j^t. The users' utility is linear in the amount of resource they are allocated. An allocation is *feasible* if $\sum_{j=1}^t \mathrm{X}_j^t \leq 1$. X is usually defined with respect to a resource allocation algorithm; this is omitted from the notation when the algorithm is obvious from context. An allocation algorithm is feasible if it always outputs a feasible allocation, i.e., if $\sum_{j=1}^t \mathrm{X}_j^t \leq 1$ for all t. An allocation is *Pareto optimal* if there is no user i whose utility can be improved without strictly decreasing the utility of some other user j. Equivalently an allocation X^t is Pareto optimal if $\sum_{j=1}^t \mathrm{X}_j^t = 1$. An allocation algorithm is Pareto optimal if it always outputs Pareto optimal allocations.

The following measure will be used to compare allocations.

Definition 1 (Domination). Let $A = (a_1, \ldots, a_t)$ and be $B = (b_1, \ldots, b_t)$ be two vectors. We say that A *dominates* B, denoted $A \succeq B$, if $\forall i \in \{1, \ldots, t\}$, $a_i \geq b_i$.

Control vectors. The input to our problem is a vector defining how many disruptions are allowed: a *control vector*.

Definition 2 (Control vector). A control vector ψ is a vector, where $\psi[i]$ denotes the number of users that may be disrupted when there are $i-1$ users in the system and another one arrives.

The first entry in any control vector ψ, i.e., $\psi[1]$, is redundant, as it refers to the empty system; nevertheless we feel the notation is clearer when it is included. Note that the definition is not restricted to finite control vectors, and can be used to define infinite control vectors as well. We sometimes refer to a user whose allocation is reduced as a *donor*.

Example 2. Consider the control vector $\psi = (1, 0, 1)$. When there is a single user in the system and another user arrives, the first user's allocation cannot be disrupted. When a third user arrives, the allocation of one of the two users currently in the system can be disrupted. For all practical purposes, the control vectors $(0, 0, 1)$ and $(1, 0, 1)$ are identical. $\triangleleft$

If $\psi[i] = d$ for all i, we say ψ is a $d-$*uniform control vector*. For example $(2, 2, 2, 2)$ is a $2-$uniform control vector. The vector $(2, 2, 2, \ldots)$ is called the infinite $d-$uniform control vector.

If the maximal number of consecutive zeros in a control vector ψ is c, we call ψ a $c-$*gap control vector*. We define a *basic* $c-$*gap control vector* to be an infinite control vector in which there is exactly one donor every $c+1$ arrivals; otherwise the control vector is non-basic. There are exactly $c+1$ possible basic $c-$gap control vectors.

Example 3. The three possible basic 2−gap control vectors are:

1. (0,1,0,0,1,0,0,1,0,...), also denoted $(0,1,0)^\infty$,
2. (0,0,1,0,0,1,0,0,1,...), also denoted $(0,0,1)^\infty$,
3. (1,0,0,1,0,0,1,0,0,...), also denoted $(1,0,0)^\infty$. ◁

Fairness. We define the fairness ratio of an allocation to be the ratio between the smallest share and the *proportional* share: $1/t$ when there are t users in the system. The fairness ratio of an allocation algorithm is the minimal fairness ratio over all possible allocations. More formally, denote the (exact) fairness ratio of an algorithm $\mathcal{A}$ with input control vector ψ, when there are t users in the system ($t \le |\psi|$) by $\textsc{fairness}(A,\psi,t)$. That is, if X^t is the allocation vector of $\mathcal{A}$ with control vector ψ when there are t users present,

$$\textsc{fairness}(\mathcal{A},\psi,t) = \frac{\min\limits_{j=1,\dots,t}\{\mathrm{X}^t_j\}}{1/t}.$$

An allocation algorithm $\mathcal{A}$ is $\sigma-$fair for a control vector ψ if $\textsc{fairness}(A,\psi,t) \le \sigma$ for all $t \in \{1,\dots,|\psi|\}$. We call the supremum of all such possible σs $\mathcal{A}$'s *fairness ratio* for ψ. We overload the notation and write $\textsc{fairness}(\mathcal{A},\psi) = \min_{t\in\{1,\dots,|\psi|\}} \textsc{fairness}(\mathcal{A},\psi,t)$. If ψ is an infinite vector, we take the infimum with respect to t: $\textsc{fairness}(\mathcal{A},\psi) = \inf_{t>0} \textsc{fairness}(\mathcal{A},\psi,t)$. The fairness ratio of a control vector ψ is defined as the maximal fairness ratio over all possible allocation algorithms: $\textsc{fairness}(\psi) = \max_{\mathcal{A}} \{\textsc{fairness}(\mathcal{A},\psi)\}$.

We are interested in the fairness ratio of (possibly infinite) $d-$uniform control vectors. Then

$$\textsc{fairness}(d-\text{uniform}) = \min_{\psi:\psi \text{ is a } d-\text{uniform control vector}} \{\textsc{fairness}(\psi)\}.$$

Similarly, we define the fairness ratio of $c-$gap control vectors.

$$\textsc{fairness}(c-\text{gap}) = \min_{\psi:\psi \text{ is a } c-\text{gap control vector}} \{\textsc{fairness}(\psi)\}.$$

We require one more definition. For some infinite control vectors, it may be the case that the worst fairness ratio occurs when there are only a few users in the system (see e.g., Figure 2 in Section 9). In these cases, it may be reasonable for the system designer to allow a few more disruptions earlier on to guarantee that the worst case fairness ratio occurs at the limit. We would therefore like to compute the worst fairness ratio for control vectors at the limit, ignoring the outliers early on. To that end, we define $\text{FAIRNESS}_{\geq n}(\psi) = \max_{\mathcal{A}} \{\inf_{t \geq n} \{\text{FAIRNESS}(\mathcal{A}, \psi, t)\}\}$, and $\text{FAIRNESS}_{\geq n}(c-\text{gap}) = \min_{\psi:\psi \text{ is a } c-\text{gap control vector}}\{\text{FAIRNESS}_{\geq n}(\psi)\}$. We note that this phenomenon does not occur for $d-$uniform control vectors, for any d; in other words, for $d-$uniform control vectors the fairness ratio always decreases as the number of users in the system increases (Lemma 2). We therefore do not need to study limit behavior separately for $d-$uniform control vectors.

Controlled dynamic fair division. The algorithm designer's goal is to design an algorithm that receives as input a control vector ψ and outputs allocations that maximize the fairness ratio, subject to performing disruptions within the bounds given by ψ. Upon a user's departure, the algorithm is not allowed to reduce the allocation of any user except, of course, the departing user (this property is sometimes called *population monotonicity*). Augmenting the resource of any user is always allowed, for both arrivals and departures. We call this problem *controlled dynamic fair division.*

EXAMPLE 4 (EXAMPLE 1 REPHRASED). Consider a system with a single resource and a capacity of 3 users. The control vector is $\psi = (1,1,1)$. Let $\mathcal{A}$ be the naïve algorithm that divides the largest available share equally at each arrival. When the first user arrives, she is allocated the entire resource; $X^1 = (1)$, $\text{FAIRNESS}(\mathcal{A}, \psi, 2) = 1$. When the second user arrives, she is given half of the resource, and the first user's resource is halved: $X^2 = (\frac{1}{2}, \frac{1}{2})$, $\text{FAIRNESS}(\mathcal{A}, \psi, 1) = 1$. When the third user arrives, one user is allocated half of the resource and the two other users are allocated one quarter of the resource each. $X^3 = (\frac{1}{2}, \frac{1}{4}, \frac{1}{4})$, therefore $\text{FAIRNESS}(\mathcal{A}, \psi, 3) = 3/4$, as the smallest allocation is $1/4$, while the proportional share is $1/3$. Hence, $\text{FAIRNESS}(\mathcal{A}, \psi) = 3/4$. ◁

3. An optimal algorithm (for a related problem)

The input to our problem is a control vector ψ, and a solution consists of a fairness ratio σ and a set of allocations $\mathrm{X}^1, \ldots, \mathrm{X}^{|\psi|}$, where X^t represents the allocation with t users in the system, such that: (i) $\frac{\min_{j=1,\ldots,t}\{\mathrm{X}^t(j)\}}{1/t} \geq \sigma$, (ii) the total number of users whose allocation is reduced from time t to time $t+1$ is at most $\psi[t+1]$. We distinguish between two cases: finite and infinite ψ. For both, the main building block is an algorithm for the following problem: given a control vector ψ and a fairness ratio σ, find feasible allocations with fairness ratio σ, or decide that σ is impossible to achieve. We shortly describe an optimal algorithm—the Frugally Fair Algorithm (FFA)—for this related problem, but first let us see how we can use it to design algorithms for the general problem. In both the finite and infinite case, if we know the optimal fairness ratio for ψ, FAIRNESS(ψ), we can invoke FFA with ψ and FAIRNESS(ψ) and generate optimal allocations. We note that FFA can generate the allocations "on the fly"; it doesn't need to generate all allocations up front, and so it can easily handle infinite control vectors. In Section 4 we show how to compute the optimal fairness for any finite control vector. In Section 5 we compute almost tight bounds for the fairness ratio of infinite control vectors. We note that, with fewer than one disruption per arrival, Pareto optimality is impossible.

LEMMA 1. *For any ψ, for which $\psi[t]=0$ for some $t \geq 2$, there is no σ-fair Pareto optimal algorithm for controlled dynamic fair division for any $\sigma > 0$.*

Proof. Assume that such an algorithm exists. Let t be the first coordinate for which $\psi[t]=0$. As the algorithm is Pareto optimal, there is no available resource, and therefore user t receives nothing. □

We now describe FFA, and note that we allow FFA to return infeasible allocations, as this will be useful for computing the optimal fairness ratio of finite control vectors. It is straightforward to modify FFA to return "infeasible" if there is some time t where $S^t > 1$.

Frugally Fair Algorithm (FFA).

Input: a control vector ψ and a fairness ratio σ.

Let $N = |\psi|$. When user $1 \leq t \leq N$ arrives, allocate $\frac{\sigma}{t}$ of the resource to her. Reduce the shares of the users with the $\psi[t]$ largest shares to $\frac{\sigma}{t}$ of the resource as well.

First we prove that FFA is optimal among all feasible allocation algorithms, for any control vector: if some allocation algorithm $\mathcal{A}$ is $\sigma-$fair for some control vector ψ then FFA is as well.

THEOREM 1. *For any allocation algorithm $\mathcal{A}$ and any (finite or infinite) control vector ψ, if* FAIRNESS$(\mathcal{A}, \psi) \geq \sigma$*, then the allocations produced by* FFA *when given ψ and σ as inputs are feasible.*

By Lemma 1, FFA cannot always be Pareto optimal. However, if there are no zeros in the control vector, it is easy to convert FFA to a Pareto optimal algorithm without loss in fairness, e.g., by simply adding the left over resource (if there is any) to the user with the largest share.

Proof of Theorem 1. Let ψ be an input for any allocation algorithm $\mathcal{A}$ and FFA. Let FAIRNESS$(\mathcal{A}, \psi) = \sigma$, i.e., $\mathcal{A}$ with input ψ produces feasible allocations with fairness ratio σ. We consider FFA when executed with input σ and ψ and show that FFA is feasible as well. We assume w.l.o.g. that $\mathcal{A}$ never increases the allocation of any user that is already in the system.

Let $\mathcal{A}^t$ and FFA^t be the sorted allocations of $\mathcal{A}$ and FFA, respectively, for t users in the system, with input ψ. As the allocations of $\mathcal{A}$ are always feasible, it suffices to show that $\mathcal{A}^t \succeq \text{FFA}^t$ for all $t \leq |\psi|$. We prove this by induction on the number of users in the system, t.

The base case: By the definition of FFA, $\text{FFA}^1(1) = \sigma$. It must clearly hold that $\mathcal{A}^1(1) \geq \sigma$, by the definition of FAIRNESS$(\mathcal{A}, \psi)$.

The inductive step: Assume the statement holds for $t-1$ users. We show it holds for t. There are two cases: $\psi[t] = 0$ and $\psi[t] \geq 1$.

Case $\psi[t]=0$ **:** Let the allocation of FFA at time $t-1$ be $(\text{FFA}_1^{t-1},\dots,\text{FFA}_{t-1}^{t-1})$. Then $\text{FFA}^t = (\text{FFA}_1^{t-1},\dots,\text{FFA}_{t-1}^{t-1},\frac{\sigma}{t})$. Algorithm $\mathcal{A}$ will allocate an $x\geq\frac{\sigma}{t}$ amount of the resource to the incoming user. Let the allocation of $\mathcal{A}$ at time $t-1$ be $(\mathcal{A}_1^{t-1},\mathcal{A}_2^{t-1},\dots,\mathcal{A}_{t-1}^{t-1})$. Then

$$\mathcal{A}^t = (\mathcal{A}_1^{t-1},\mathcal{A}_2^{t-1},\dots,\mathcal{A}_j^{t-1},x,\mathcal{A}_{j+1}^{t-1},\dots,\mathcal{A}_{t-1}^{t-1}),$$

for some $0\leq j\leq t-1$. It is easy to see that $\mathcal{A}^t\succeq\left(\mathcal{A}_1^{t-1},\mathcal{A}_2^{t-1},\dots,\mathcal{A}_{t-1}^{t-1},\frac{\sigma}{t}\right)$ for any such value of j. Furthermore, by the induction hypothesis,

$$\left(\mathcal{A}_1^{t-1},\mathcal{A}_2^{t-1},\dots,\mathcal{A}_{t-1}^{t-1},\frac{\sigma}{t}\right)\succeq(\text{FFA}_1^{t-1},\dots,\text{FFA}_{t-1}^{t-1},\frac{\sigma}{t})=\text{FFA}^t.$$

Case $\psi[t]\geq 1$ **:** Starting with the allocation vector at time $t-1$, $\mathcal{A}^{t-1}$, we break the allocation changes at time t into $\psi[t]+1$ steps

1. In step ℓ, for $\ell=1,\dots,\psi[t]$, reduce the share of the ℓ-th donor to get $\hat{\mathcal{A}}_\ell^{t-1}$.
2. In step $\psi[t]+1$, allocate x to the incoming user, to obtain $\mathcal{A}^t$.

Denote $\mathcal{A}_\ell^{\min}=(\mathcal{A}_{\ell+1}^{t-1},\dots\mathcal{A}_{t-1}^{t-1},\frac{\sigma}{t},\dots,\frac{\sigma}{t})$, where $\frac{\sigma}{t}$ appears ℓ times. We show that for all $\ell=1,\dots,\psi[t]$, $\hat{\mathcal{A}}_\ell^{t-1}\succeq\mathcal{A}_i^{\min}$. The last step (step $\psi[t]+1$) is identical to the $\psi[t]=0$ case, and hence, by combining the two observations, we can conclude that $A^t\succeq\text{FFA}^t$.

We focus on the $\ell=1$ case; the other cases are identical. To simplify notation we drop the subscript and write $\hat{\mathcal{A}}^{t-1}$ and $\mathcal{A}^{\min}$ instead of $\hat{\mathcal{A}}_1^{t-1}$ and $\mathcal{A}_1^{\min}$. To show that $\hat{\mathcal{A}}^{t-1}\succeq\mathcal{A}^{\min}$, assume algorithm $\mathcal{A}$ reduced the allocation of a user from $\mathcal{A}_j^{t-1}$ to y, where $1\leq j\leq t-1$ (possibly $\mathcal{A}_j^{t-1}=y$). There is some k, $j\leq k\leq t-1$ such that $\mathcal{A}_k^{t-1}\geq y\geq\mathcal{A}_{k+1}^{t-1}$. Then,

$$\hat{\mathcal{A}}^{t-1}=(\mathcal{A}_1^{t-1},\mathcal{A}_2^{t-1},\dots,\mathcal{A}_{j-1}^{t-1},\mathcal{A}_{j+1}^{t-1},\mathcal{A}_{j+2}^{t-1},\dots,\mathcal{A}_k^{t-1},y,\mathcal{A}_{k+1}^{t-1},\dots\mathcal{A}_{t-1}^{t-1}).$$

For $i\in[1,j-1]$, $\hat{\mathcal{A}}_i^{t-1}\geq\mathcal{A}_i^{\min}$. For $i\in[j,k]$, $\hat{\mathcal{A}}_i^{t-1}=\mathcal{A}_i^{\min}$. Then, by definition of y, $\hat{\mathcal{A}}_{k+1}^{t-1}=y\geq\mathcal{A}_{k+1}^{t-1}=A_{k+1}^{\min}$. Similarly, $\hat{\mathcal{A}}_i^{t-1}\geq\mathcal{A}_i^{\min}$, for all $i\in[k+1,t-2]$. For the last term, note that since $\mathcal{A}$ is $\sigma-$fair, $\mathcal{A}_{t-1}^{t-1}\geq\frac{\sigma}{t-1}\geq\frac{\sigma}{t}$ (possibly y is the last share, but then $y\geq\frac{\sigma}{t}$ as $\mathcal{A}$ is $\sigma-$fair). □

4. Finite control vectors

If the input control vector ψ is finite, computing the optimal fairness is straightforward. We invoke FFA with ψ and $\sigma = 1$. As we cannot expect perfect fairness, some of these allocations will typically be infeasible. We note that when FFA is run with the same ψ but different values of σ, the allocations are different, but their relative size remains the same; we can therefore compute the smallest amount by which we need to scale the allocations produced by FFA with ψ and $\sigma = 1$ down so that the largest allocation is 1, and this is exactly the optimal fairness ratio. The optimal algorithm for finite control vectors, FiniteFFA, is therefore the following: Execute FFA twice; once to compute the optimal fairness ratio and then with this fairness ratio as the input, to obtain the actual allocations.

Finite Frugally Fair Algorithm (FiniteFFA).

Input: a finite control vector ψ.

Let $|\psi| = N$.

1. Run FFA with input ψ and $\sigma = 1$. Let S^t be the total resource allocated by FFA when there are t users, $1 \leq t \leq N$. Set $\sigma^* = (\max_{1\leq t\leq N}\{\mathrm{S}^t\})^{-1}$.
2. Run FFA with input ψ and σ^*.

Theorem 2. FiniteFFA *is an optimal algorithm for the single-resource controlled dynamic fair division for any finite control vector.*

Proof. Fix ψ. Observe that S^t, the total resource allocated at time t when FFA is executed with input ψ and $\sigma = 1$, is equal to $\hat{\sigma} \cdot \hat{\mathrm{S}}^t$, where $\hat{\mathrm{S}}^t$ is the total resource allocated at time t by FFA with input ψ and $\hat{\sigma}$, for any $\hat{\sigma}$. Therefore, FFA executed with input ψ and $\sigma^* = (\max_{1\leq t\leq N}\{\mathrm{S}^t\})^{-1}$ is feasible. Furthermore, FFA executed with input ψ and $\hat{\sigma} > \sigma^*$ creates an infeasible allocation at some time $t \leq |\psi|$. The contrapositive of Theorem 1 then implies that there is no allocation algorithm $\mathcal{A}$ that produces feasible allocations with fairness ratio $\hat{\sigma} > \sigma^*$. It follows that FiniteFFA is optimal for ψ. $\square$

It is straightforward to compute the fairness ratio and allocations of finite control vectors, using the procedure given in Step 1 of FiniteFFA. In Table 1, we give the optimal fairness ratios for some values of d, for $d-$uniform control vectors.

$\|\psi\|$	d					
	1	2	3	5	10	50
2	1	1	1	1	1	1
3	0.857	1	1	1	1	1
5	0.811	0.909	0.952	1	1	1
10	0.774	0.868	0.913	0.957	1	1
100	0.727	0.827	0.874	0.919	0.958	0.995
1,000	0.722	0.823	0.870	0.915	0.954	0.991
100,000	0.721	0.822	0.869	0.914	0.954	0.990

Table 1 Optimal fairness ratios for finite $d-$uniform control vectors different lengths for some values of d.

In addition to the values in Table 1, we supply a simple upper bound on the fairness ratio of *any* finite control vector. We note that this applies even to adversarial control vectors, where the adversary is completely unrestricted.

Theorem 3. *Let ψ be a finite control vector, $|\psi| = n$. Then* $\text{FAIRNESS}(\psi) \geq (H_n)^{-1}$, *where H_n denotes the $n-$th harmonic number.*

Proof. Given any σ as the auxiliary fairness ratio, the allocation of FiniteFFA after the last user arrives is $\left(\sigma, \frac{\sigma}{2}, \dots \frac{\sigma}{n}\right)$. The total allocation with n users in the system will therefore be σH_n. Setting $\sigma H_n = 1$ completes the theorem. □

5. Infinite control vectors

We now turn to infinite control vectors; unfortunately, we cannot use FFA to compute the optimal fairness ratio in this case. Nevertheless, we would like to allow for systems that can accommodate an arbitrary number of users. In these cases, we can still invoke FFA with the infinite control vector ψ and a fairness ratio σ as input, but we must find another way of computing σ. As long as σ is upper bounded by the worst fairness ratio possible for any number of users, the allocations produced by FFA will be feasible. We would like to give FFA the tightest σ possible, and towards that end, we precisely compute $\textsc{fairness}(d-\text{uniform})$ for any $d \geq 1$ (Subsection 5.1). In Subsection 5.2, we show that for $c \geq 3$, there is no unique optimal ratio (i.e., different basic $c-$gap control vectors have different optimal fairness ratios), and provide a tight asymptotic bound and almost-tight bounds that hold for *all* basic $c-$gap control vectors. In addition, we compute $\textsc{fairness}(c-\text{gap})$ for $c \in \{1, 2\}$. We note that this discrepancy between $c = 1, 2$ and $c = 3$ is unavoidable, as the asymptotic bound holds for all time periods for $c = 1$ and $c = 2$, but not for $c > 3$ (e.g., Figure 2 in Section 9). We note that a consequence of our results is that our algorithm needs to know c, the maximum gap possible between donors, but does not need to know the exact control vector a-priori.

5.1. Infinite uniform control vectors

Our main result in this section is an exact characterization of $\textsc{fairness}(d-\textit{uniform})$, the optimal fairness ratio of (possibly) infinite $d-$uniform control vectors. Note that we can easily find allocations that are feasible for this fairness ratio by executing FFA with input an infinite $d-$uniform control vector ψ and the corresponding fairness ratio $\textsc{fairness}(\psi)$.

THEOREM 4. *For any $d \geq 1$, it holds that*

$$\textsc{fairness}(d-\text{uniform}) = \frac{1}{(d+1)\ln\left(\frac{d+1}{d}\right)}.$$

Our proof will be via a characterization of the allocations output by FFA, when the input is a $d-$uniform control vector (and some σ), in Proposition 1. We then show that the fairness ratio is non-increasing in the length of the control vector in Lemma 2. The proofs of Proposition 1 and Lemma 2 appear in Appendix B. Using these two results, Theorem 4 is then proved by bounding the allocations described in Proposition 1 in the limit.

PROPOSITION 1. *Let* X^t *denote the allocation of* FFA *when there are* t *users in the system, when* FFA*'s inputs are the infinite* $d-$*uniform control vector and* σ*. For all* $t = \ell \cdot (d+1) + i$*,* $i \in [0,d]$*,*

$$X^t = \left(\underbrace{\frac{\sigma}{t-\ell}, \ldots, \frac{\sigma}{t-\ell}}_{i \ terms}, \underbrace{\frac{\sigma}{t-\ell+1}, \ldots, \frac{\sigma}{t-\ell+1}}_{d+1 \ terms}, \ldots, \underbrace{\frac{\sigma}{t-1}, \ldots, \frac{\sigma}{t-1}}_{d+1 \ terms}, \underbrace{\frac{\sigma}{t}, \ldots, \frac{\sigma}{t}}_{d+1 \ terms} \right).$$

LEMMA 2. *Let* ψ^N *denote a* $d-$*uniform control vector of length* N*,* $d \geq 1$*. Then,* FAIRNESS(FFA, ψ^N) *is monotone non-increasing in* N*.*

Proof of Theorem 4. Let X^t denote allocation of FFA on input the infinite $d-$uniform control vector when there are t users in the system. Denote the total amount of resource allocated by $S^t = \sum_{i=1}^t X_i^t$. We first show that S^t increases with t. Let $t = \ell \cdot (d+1) + i$, for some ℓ and i. Then $S^t = i \cdot \frac{\sigma}{t-\ell} + (d+1) \cdot \sum_{k=1}^{\ell} \frac{\sigma}{t-\ell+k} = i \cdot \frac{\sigma}{\ell d+i} + (d+1) \cdot \sum_{k=1}^{\ell} \frac{\sigma}{\ell d+i+k}$. Let $t' = \ell \cdot (d+1) + i + 1$.

$$\begin{aligned} S^{t'} - S^t &= \left(\frac{\sigma(i+1)}{\ell d+i+1} + (d+1) \cdot \sum_{k=1}^{\ell} \frac{\sigma}{\ell d+i+1+k} \right) - \left(i \cdot \frac{\sigma}{\ell d+i} + (d+1) \cdot \sum_{k=1}^{\ell} \frac{\sigma}{\ell d+i+k} \right) \\ &= \frac{(d+1)\sigma}{\ell d+i+1+\ell} - i \cdot \frac{\sigma}{\ell d+i} - \frac{\sigma((d+1)-(i+1))}{\ell d+i+1} \\ &= \sigma \frac{d-i}{(\ell d+i)(\ell d+i+1)(\ell d+i+\ell+1)}, \end{aligned}$$

which is non-negative for all $i \leq d$. Therefore, $S^{(\ell+1)(d+1)} \geq S^{\ell(d+1)+i}$, for all $i \in [0,d]$. The same argument shows that $S^{\ell(d+1)+1} \geq S^{\ell(d+1)}$. Thus, it suffices to bound S^t at the limit. Without loss of generality we analyze $t = \ell(d+1)$.

$$\begin{aligned} \lim_{\ell\to\infty} S^{\ell(d+1)} &= \lim_{\ell\to\infty} \sum_{k=1}^{\ell} \frac{\sigma(d+1)}{\ell d+k} \\ &= \sigma(d+1) \cdot \lim_{\ell\to\infty} \sum_{k=\ell d+1}^{\ell(d+1)} \frac{1}{k} \\ &= \sigma(d+1) \cdot \lim_{\ell\to\infty} \left(H_{\ell(d+1)} - H_{\ell d} \right) \\ &= \sigma(d+1) \cdot \lim_{\ell\to\infty} \left(\ln(\ell(d+1)) - \ln(\ell d) \right) \\ &= \sigma(d+1) \cdot \ln\left(\frac{d+1}{d} \right). \end{aligned}$$

As S^t is increasing in t, we set $\lim_{\ell\to\infty} S^{\ell(d+1)} = 1$. From Lemma 2, this characterizes the infimum of the fairness ratio of all $d-$uniform control vectors. □

5.2. General infinite control vectors

Showing bounds for FAIRNESS($c-$gap) is much more involved than for FAIRNESS($d-$uniform) for several reasons. First, there are infinitely many possible infinite $c-$gap control vectors, as opposed to exactly one for constant $d \geq 1$. For example, the only infinite $2-$uniform control vector is $(2,2,2,\ldots)$. Furthermore, a bound for $2-$uniform control vectors also immediately holds for control vectors where all elements are *at least* 2, as one can always simply execute FFA on a $2-$uniform control vector, ignoring the extra available donors. This is not the case when there are steps where no donor is allowed; we cannot simply use 0 donors every round and obtain any meaningful result. In this section we consider *binary* control vectors. This is without loss of generality, as we would like to compute the worst case fairness ratio (and, similarly to the argument above, any allocation algorithm can always treat an element in the control vector that is greater than one as one).

We note that even if we only considered *basic* $c-$gap control vectors, the allocations produced by FFA for each of them are very different; furthermore, they are not as "well behaved" as the allocations for $d \geq 1$. Compare Figures 1 and 2 in Section 9. The $x-$axis is the number of users and the $y-$axis is the total resource allocated at step t in the execution of FFA with input ψ and $\sigma = 1$. The total resource allocated is, of course, at least 1, since the optimal fairness ratio is the inverse of the maximal total allocation created, which is at most 1. Figure 1 shows the first 100 allocations created by FFA when given the infinite $d-$control vector and $\sigma = 1$ as input. Figure 2 shows the first 30 allocations created by FFA given the 4 basic $3-$gap control vectors and $\sigma = 1$ as input. The complexity in the second setting is due to several reasons: the allocations are not monotone, they are not pointwise comparable, and they are not simple transformations one of the other. Furthermore, the allocations do not necessarily take their maxima at the limit. Still, in both cases, the total allocation converges to a limit as the number of users grows (in the second case, *all* infinite allocations that obey certain natural requirements converge to the *same* limit (Theorem 6)). The horizontal line in both figures denotes this limit. By Lemma 2, the first set of allocations takes its maximum at the limit, however Figure 2 shows that the second case does not.

It might be tempting to think that, as in Figure 2, there always exists *some* basic $c-$gap control vector that takes its maximum at the limit, and this is indeed the case for $c<8$ (we do not prove this, but it is easy to verify). However, for $c\geq 8$, *every* basic $c-$gap control vector has its maximum at a finite number of users (see Figure 3 in Appendix 9 for a pictorial example).

The main results of this section are almost matching upper and lower bounds for the optimal fairness ratio attainable for any (possibly infinite) $c-$gap control vector.

THEOREM 5. *The optimal fairness ratio for any* $c-$gap *control vector,* $c\geq 1$*, is bounded by*

$$(H_{c+1})^{-1}\geq \text{FAIRNESS}(c-\text{gap})\geq \left(H_{2c+3}-\tfrac{1}{2}\right)^{-1},$$

where H_n *denotes the* $n-th$ *harmonic number.*

To prove this, we consider a set of allocations that are pointwise greater than the allocations for all basic vectors simultaneously, and bound them. In addition, we show that as the number of users grows, the optimal fairness ratio for all basic $c-$gap control vectors converge to the same value, which is given in Theorem 6.

THEOREM 6. *For every* $c\geq 1$*, there exists a number* n_0 *such that,*

$$\text{FAIRNESS}_{\geq n_0}(c-\text{gap})=(c+1)\left((c+2)\ln(c+2)\right)^{-1}.$$

In addition, we show that we can compute $\text{FAIRNESS}(1-gap)$ precisely for $c=1$ and $c=2$, as it holds that the worst allocations are always at the limit. The proof appears in Appendix C.1.

THEOREM 7. *The optimal fairness ratios for* $1-$ *and* $2-$gap *control vectors are*

1. $\text{FAIRNESS}(1-\text{gap})=2(3\ln 3)^{-1}$,
2. $\text{FAIRNESS}(2-\text{gap})=3(4\ln 4)^{-1}$.

The proofs of Theorems 5 and 6 involve the following two steps:

1. (Subsection 5.2.1). Showing that basic control vectors are the worst; that is, that for every $c-$gap control vector ψ, there is a basic $c-$gap control vector ψ' whose fairness ratio is at least as bad (i.e., $\text{FAIRNESS}(\psi')\leq\text{FAIRNESS}(\psi)$).

2. (Subsection 5.2.2). Bounding the fairness ratio of *all* basic $c-$gap control vectors concurrently.

5.2.1. Reduction to basic control vectors. We would like to compute a lower bound on the fairness ratio for *all* $c-$gap control vectors, but the optimal fairness ratio for any control vector depends on the vector itself. We first show that basic control vectors have the worst fairness ratio; hence in order to provide a lower bound on the fairness ratio, it suffices to analyze basic control vectors. Denote by $\text{FFA}(\psi,\sigma)$ the (possibly infinite) set of allocations of FFA on input ψ and σ. It will be useful to think of the unallocated resource as a "bank". Let X^t denote allocation of FFA given the infinite $d-$uniform control vector and some σ as input, at time t. As before, set $\text{S}^t = \sum_{i=1}^{t} \text{X}_i^t$. Let $\textsc{bank}(\psi, t) = 1 - \text{S}^t$.

Lemma 3. *For every $c-$gap control vector ψ, there exists some basic $c-$gap control vector $\hat{\psi}$ such that* $\textsc{fairness}(\psi) \geq \textsc{fairness}\left(\hat{\psi}\right)$.

Proof. Define a series of control vectors $\hat{\psi} = \psi_1, \psi_2, \ldots, \psi_k = \psi$ such that if $\text{FFA}(\psi_i,\sigma)$ is feasible, then $\text{FFA}(\psi_{i+1},\sigma)$ is feasible, for all $1 \leq i < k$. We define these vectors inductively: ψ_1 is the (infinite) basic $c-$gap control vector whose first "disagreement" with ψ is as late as possible. For $i > 1$, let t_i^* be the leftmost coordinate on which ψ_i and ψ differ. The first $t_i^* - 1$ elements of ψ_{i+1} are the same as ψ_i, the t_i^*th element becomes the same as $\psi[t_i^*]$ (the t_i^*th element of ψ), and the remainder continues as $(0^c1)^\infty$.

Example 5. Let $\psi = (0,1,1,1,0,1,0,0,1)$. We derive the ψ_is.

$$\hat{\psi} = \psi_1 = (0,1),(0,0,1)^\infty,$$

$$\psi_2 = (0,1,1),(0,0,1)^\infty,$$

$$\psi_3 = (0,1,1,1),(0,0,1)^\infty$$

$$\psi_4 = (0,1,1,1,0,1),(0,0,1)^\infty = \psi(0,0,1)^\infty$$

Note that, $\hat{\psi}$ is an infinite basic $c-$gap control vector and has a worse fairness ratio than any finite basic $c-$gap control vector that is a prefix of $\hat{\psi}$. We prove that $\hat{\psi}$ is worse than the control vector $\psi(0^c1)^\infty$, which is in turn worse than the control vector ψ.

In Proposition 2, we show that for every i and all steps t such that $t = t_i^* \pmod{c+1}$, if $\mathrm{FFA}(\psi_i, \sigma)$ is feasible, then so is $\mathrm{FFA}(\psi_{i+1}, \sigma)$. In Proposition 4 we show that $\mathrm{FFA}(\psi_{i+1}, \sigma)$ remains feasible for all other time steps ($t \neq t_i^* \pmod{c+1}$). The lemma follows from these two propositions, as together they cover all time steps. □

Fix σ, and let $\hat{\psi} = \psi_1, \psi_2, \ldots, \psi_k = \psi$ be as in the proof of Lemma 3. Denote the set of allocations of $\mathrm{FFA}(\psi, \sigma)$ by $\mathcal{X}_\psi$. Notice that $\mathcal{X}_{\psi_i}$ and $\mathcal{X}_{\psi_{i+1}}$, the allocation sets of $\mathrm{FFA}(\psi_i, \sigma)$ and $\mathrm{FFA}(\psi_{i+1}, \sigma)$, are identical up to step $t_i^* - 1$. Then, on step t_i^*, necessarily $\psi_i[t_i^*] = 0$, $\psi_{i+1}[t_i^*] = 1$.

Let $\mathcal{X}_\psi^t$ denote the allocation produced by FFA with inputs ψ and σ at step t. The proofs of Propositions 2 and 4 (as well as Proposition 3, which is used to prove Proposition 4), are deferred to Appendix B.

Proposition 2. *For all $1 \leq i \leq k-1$ and all $t = t_i^* \pmod{c+1}$,*

$$\mathcal{X}_{\psi_i}^t \succeq \mathcal{X}_{\psi_{i+1}}^t.$$

Showing Proposition 4—that if $\mathrm{FFA}(\psi_i, \sigma)$ is feasible, then $\mathrm{FFA}(\psi_{i+1}, \sigma)$ is feasible, for steps $t \neq t_i^* \pmod{c+1}$—is more involved. First we show that from a certain time onwards, denoted T_{ψ_i}, $\mathcal{X}_{\psi_{i+1}}^t$ is identical to $\mathcal{X}_{\psi_i}^t$, hence we only need to consider allocations prior to that point.

Proposition 3. *For all $1 \leq i \leq k-1$, there exists some (minimal) T_{ψ_i} such that, for all steps $t > T_i^{\max}$, $\mathcal{X}_{\psi_{i+1}}^t$ is identical to $\mathcal{X}_{\psi_i}^t$.*

In order to show that $\mathrm{FFA}(\psi_{i+1}, \sigma)$ is feasible, we want to show that the total resource allocated in $\mathcal{X}_{\psi_{i+1}}^t$ for all $t \in [t_i^*, T_{\psi_i}]$, $t \neq t_i^* \pmod{c+1}$ is at most one. Equivalently, if it suffices to show that for all steps when a donor is used, $\mathrm{BANK}(\psi_{i+1}, t)$ is large enough to handle c consecutive non-disruptive steps.

Proposition 4. *For $1 \leq i \leq k-1$, let T_{ψ_i} be as in Proposition 3. For all $t = t_i^* \pmod{c+1}$, $t \in [t_i^*, T_{\psi_i}]$, $\mathrm{BANK}(\psi_{i+1}, t) \geq \sum_{j=1}^{c} \frac{\sigma}{t+j}$.*

5.2.2. Bounding the fairness ratio of basic control vectors. Having shown that the basic control vectors have the worst fairness ratio, it remains to bound it. The allocations created by FFA on binary $c-$gap control vectors have a particular structure: they resemble a segment of the harmonic series, with some doubled entries (see, e.g., Table 3 in Appendix C.1). Even though we showed that in order to bound the fairness ratio, it is enough to consider only basic control vectors, each basic control vector has a different fairness ratio. Nevertheless, we would like to provide some upper bound on the fairness ratio for each c. We give two types of bounds: an upper bound that applies to all control vectors for any number of users, and an asymptotic bound. The asymptotic bound is particularly useful for systems that wish to be able to accommodate an unbounded number of users, and in which the amount of time the system will have fewer than n_0 users (for some constant n_0) is vanishingly small. In this case, one can set the fairness ratio to be the asymptotic fairness ratio, with an arbitrary "quick fix" heuristic for when there are fewer than n_0 users in the system (for example, allowing slightly more disruptions to guarantee the asymptotic fairness ratio). The following proposition bounds the fairness of all basic $c-$gap control vectors concurrently and is the key ingredient in the proofs of both Theorems 5; its proof appears in Appendix C.5.

PROPOSITION 5. *Let ψ be any basic $c-$gap control vector. Then for any $t>1$,*

$$\textsc{fairness}(\text{FFA},\psi,t) \geq \left(\arg\max_{t\in\mathbb{N}^+, j\in\{0,\ldots,c+1\}} \sum_{i=t}^{(t-1)(c+2)+j} \frac{1}{i} + \sum_{i=0}^{t-2} \frac{1}{t+i(c+1)+j}\right)^{-1}.$$

We are finally ready to prove Theorems 5 and 6.

Proof of Theorems 5 and 6. We restate the theorems before their proof for convenience.

THEOREM 5. *The optimal fairness ratio for any $c-$gap control vector, $c\geq 1$, is bounded by*

$$(H_{c+1})^{-1} \geq \textsc{fairness}(c-\text{gap}) \geq \left(H_{2c+3} - \tfrac{1}{2}\right)^{-1},$$

where H_n denotes the $n-th$ harmonic number.

Proof. We use the bound of $\textsc{fairness}(\text{FFA},\psi,t)$ from Proposition 5. We consider two cases: $t=1$ and $t>1$.

Case $t=1$ **:** The total allocation is simply $\sigma+\frac{\sigma}{2}+\cdots+\frac{\sigma}{j}$. This is simply the first j elements of the harmonic progression multiplied by σ. The largest value it can take is when $j=c+1$, hence $(H_{c+1})^{-1}$ provides an upper bound on the fairness ratio.

Case $t>1$ **:** From Proposition 5, the total resource allocated by FFA at time t when executed with ψ and $\sigma=1$ is:

$$\sum_{i=t}^{(t-1)(c+2)+j}\frac{1}{i}+\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)+j}\leq\sum_{i=t}^{(t-1)(c+2)+c+1}\frac{1}{i}+\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)}.$$

We bound each term separately. The term on the left, $\sum_{i=t}^{(t-1)(c+2)+c+1}\frac{1}{i}$, is decreasing in t. This is restated as Lemma 5 and proved in Appendix B. Plugging in $t=2$, we therefore have that this term is upper bounded by $\sum_{i=2}^{2c+3}\frac{1}{i}=H_{2c+3}-1$.

The term on the right, $\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)}$, is upper bounded by $\frac{1}{2}$ for all $c>1$ (Lemma 6 in Appendix B). Combined, we get that the total resource allocated is at most $H_{2c+3}-\frac{1}{2}$. Hence $\left(H_{2c+3}-\frac{1}{2}\right)^{-1}$ is a lower bound on FAIRNESS$(c-\text{gap})$. □

THEOREM 6. *For every $c\geq 1$, there exists a number n_0 such that,*

$$\text{FAIRNESS}_{\geq n_0}(c-\text{gap})=(c+1)\left((c+2)\ln(c+2)\right)^{-1}.$$

Proof. In Appendix A, we show the following result about harmonic sums: $\sum_{x=a}^{b}\frac{1}{x}\leq\ln\left(\frac{b}{a-1}\right)$. We use this to upper bound the total allocation at step t, as it appears in Proposition 5, $\sum_{i=t}^{(t-1)(c+2)+j}\frac{1}{i}+\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)+j}$. For the first term:

$$\sum_{i=t}^{(t-1)(c+2)+j}\frac{1}{i}\leq\ln\left(\frac{(t-1)(c+2)+j}{t-1}\right)=\ln\left(c+2+\frac{j}{t-1}\right),$$

which approaches $\ln(c+2)$ as $t\to\infty$. For the second term:

$$\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)+j}\leq\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)}=\frac{1}{c+1}\sum_{i=0}^{t-2}\frac{1}{\frac{t}{c+1}+i}=\frac{1}{c+1}\sum_{i=\frac{t}{c+1}}^{\frac{t}{c+1}+t-2}\frac{1}{i}$$
$$\leq\frac{1}{c+1}\ln\left(\frac{\frac{t}{c+1}+t-2}{\frac{t}{c+1}-1}\right)=\frac{1}{c+1}\ln\left(\frac{t+(t-2)(c+1)}{t-c-1}\right),$$

which approaches $\frac{1}{c+1}\ln(c+2)$ as $t \to \infty$. Combining gives an upper bound of $(c+1)^{-1}((c+2)\ln(c+2))$ on the total allocation at step t, as t goes to infinity. The lower bound is virtually identical, using the result $\ln\left(\frac{b}{a-1}\right) - \frac{1}{2a-2} \le \sum_{x=a}^{b} \frac{1}{x}$, also shown in Appendix A, and is omitted. □

6. Accommodating Departures

Thus far, we have only considered arrivals; when allowing for departures of users, the situation can get more complex. The optimal fairness ratio could theoretically depend on a number of parameters, for example the allocation of the departing user. In this section we prove that this is not the case.

For concreteness, we formally state the problem. The input is a control vector ψ. Users arrive and depart in an order a-priory unknown to the algorithm. $\psi[t] = d$ means that the algorithm is allowed to reduce the allocation of d users when there are $t-1$ users in the system and a new one arrives. Note that as departures are allowed, this can happen multiple times. When a user departs, the algorithm is not allowed to decrease the allocation of any user (except the departing user). We call this the *arrivals-departures model*. We show that is is straightforward to augment FFA such that its fairness ratio for a given control vector ψ is the same as when users are not allowed to depart.

THEOREM 8. *There exists an algorithm $\mathcal{A}$ such that* FAIRNESS$(\mathcal{A}, \psi)$ *in the arrivals-departures model equals* FAIRNESS(FFA, ψ) *in the arrivals-only model.*

Proof. Let X^t and X^{t+1} be the (sorted) allocations of FFA for a given control vector ψ. It suffices to show that when an arbitrary user from X^{t+1} departs, it is possible to distribute her share in a way that the sorted allocation is X^t. If $\psi[t+1] = 0$, this is straightforward: if the user with the j-th largest allocation leaves, we simply augment the smallest allocation to X_j^{t+1}. Without loss of generality, we consider $\psi[t+1] = 1$; It is straightforward to extend the proof to $\psi[t+1] > 1$. The two allocations we consider are therefore $\mathrm{X}^t = (a_1, a_2, \ldots, a_t)$ and $\mathrm{X}^{t+1} = \left(a_2, a_3, \ldots, a_t, \frac{\sigma}{t+1}, \frac{\sigma}{t+1}\right)$.

Assume that one of the last two users, with a $\frac{\sigma}{t+1}$ share, departs. Since both X^t and X^{t+1} are feasible, $a_1 - 2\frac{\sigma}{t+1}$ is equal to the difference $\text{BANK}^{t+1} - \text{BANK}^t$. Therefore, there must be a way to combine the departing user's share $\frac{\sigma}{t+1}$, with the other share equal to $\frac{\sigma}{t+1}$, and the unallocated amount $\textsc{bank}^{t+1}$, to get a_1. Assume for contradiction this is not the case; then $a_1 > 2\frac{\sigma}{t+1} + \textsc{bank}^{t+1}$, which implies that $a_1 - 2\frac{\sigma}{t+1} > \textsc{bank}^{t+1}$. The LHS is equal to $\textsc{bank}^{t+1} - \textsc{bank}^{t}$. Combining gives that $\textsc{bank}^{t} < 0$, a contradiction.

If the departing user is some user $j \in [2, t-1]$, we can do the following: allocate the difference $a_j - \frac{\sigma}{t+1}$ (which is positive since the allocation is sorted) to one of the last two users. The amount left to distribute is exactly $\frac{\sigma}{t+1}$; we've already shown this is sufficient to increase the share of the other of the last two users to a_1. □

7. Conclusion and future directions

In this paper we introduced and studied the trade-off between fairness and the amount of allowed disruptions in a dynamic fair division setting. We described an algorithm that receives as input a control vector and outputs a set of allocations, and showed that it is optimal when the control vector is finite (and known a-priori) or is $d-$uniform. In addition, we gave upper and lower bounds on the fairness ratio in systems that allow fewer disruptions than arrivals and showed that the fairness ratio decays logarithmically with the number of consecutive arrivals in which a disruption is not allowed. Our results give insights on how to design shared-resource systems that need to maintain a high degree of fairness, while still allowing for smooth operation.

There are various directions in which our research can be extended. *Fairness ratio* is a very strong notion of fairness; it requires that the smallest share received by any user at any time be sufficiently large. This may be too strong a requirement for many systems. For example, most users would probably not mind being allocated slightly fewer resources at some time periods if they are compensated at other times. This is especially true if jobs have deadlines; a user might be willing to accept a slightly more "unfair" allocation at some time period if it means her job will finish earlier overall. In addition, one may wish to take into consideration the system requirements. In a software company, for example, if the system requires a few more seconds to complete the execution of an important program, and one hundred agents join, the system might prefer to delay the agents' entry into the system for a while. Under our definition, this system would have a fairness ratio of zero. This example bring us to another direction for future research - balancing fairness and social welfare (the sum of the agents' utilities). It would be interesting to characterize the loss of social welfare caused by fairness requirements. Bertsimas et al. (2011) study an axiomatic such trade-off in static settings.

Finally, real world systems typically consist of multiple resources (e.g., CPU, RAM, disk space and I/O resources), which users require in different proportions. It is interesting to extend our results to the multi-resource setting, which possesses many complications not found in the single-resource setting. For example, the users' demands need to be input to the algorithm, and in addition to adding additional parameters, this also causes incentive issues. While the notion of a fair allocation in the (homogeneous) single resource case is fairly intuitive—the only truly fair allocation assigns each user an equal amount of the resource—when there are multiple resources this is not the case. There are several notions of fairness in the literature, such as the Nash bargaining solution (Nash Jr 1950) and Dominant Resource Fairness (DRF) (Ghodsi et al. 2011); when considered on a single resource, they all reduce to proportional allocations. We briefly addressed the question of maximizing the fairness ratio with respect to DRF in a similar setting in a recent working paper (Friedman et al. 2017).

References

Ahmadi RH, Dasu S, Tang CS (1992) The dynamic line allocation problem. *Management Science* 38(9):1341–1353, URL `http://dx.doi.org/10.1287/mnsc.38.9.1341`.

Aleksandrov M, Aziz H, Gaspers S, Walsh T (2015) Online fair division: analysing a food bank problem. *Proceedings of the 24th International Conference on Artificial Intelligence*, 2540–2546 (AAAI Press).

Alon N (1987) Splitting necklaces. *Advances in Mathematics* 63(3):247–253.

Aziz H, Mackenzie S (2016) A discrete and bounded envy-free cake cutting protocol for four agents. *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing* 454–464, URL `http://dx.doi.org/10.1145/2897518.2897522`.

Benade G, Kazachkov AM, Procaccia AD, Psomas CA (2018) How to make envy vanish over time. *Proceedings of the Nineteenth ACM Conference on Economics and Computation*, 593–610 (ACM).

Bertsimas D, Farias VF, Trichakis N (2011) The price of fairness. *Operations Research* 59(1):17–31, URL http://dx.doi.org/10.1287/opre.1100.0865.

Bertsimas D, Farias VF, Trichakis N (2013) Fairness, efficiency, and flexibility in organ allocation for kidney transplantation. *Operations Research* 61(1):73–87, URL http://dx.doi.org/10.1287/opre.1120.1138.

Bhattacharya AA, Culler D, Friedman E, Ghodsi A, Shenker S, Stoica I (2013) Hierarchical scheduling for diverse datacenter workloads. *Proceedings of the Fourth Annual Symposium on Cloud Computing*, 4:1–4:15, SOCC '13.

Boiney LG (1995) When efficient is insufficient: Fairness in decisions affecting a group. *Management Science* 41(9):1523–1537, URL http://dx.doi.org/10.1287/mnsc.41.9.1523.

Brams SJ, Taylor AD (1995) An envy-free cake division protocol. *American Mathematical Monthly* 9–18.

Budish E (2011) The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy* 119(6):1061–1103.

Ciocan DF, Farias V (2012) Model predictive control for dynamic resource allocation. *Mathematics of Operations Research* 37(3):501–525, URL http://dx.doi.org/10.1287/moor.1120.0548.

Correa JR, Schulz AS, Stier-Moses NE (2007) Fast, fair, and efficient flows in networks. *Operations Research* 55(2):215–225, URL http://dx.doi.org/10.1287/opre.1070.0383.

Demers A, Keshav S, Shenker S (1989) Analysis and simulation of a fair queueing algorithm. *ACM SIGCOMM Computer Communication Review*, volume 19, 1–12 (ACM).

Deng X, Qi Q, Saberi A (2012) Algorithmic solutions for envy-free cake cutting. *Operations Research* 60(6):1461–1476, URL http://dx.doi.org/10.1287/opre.1120.1116.

Dolev D, Feitelson DG, Halpern JY, Kupferman R, Linial N (2012) No justified complaints: On fair sharing of multiple resources. *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 68–75, ITCS '12 (New York, NY, USA: ACM), ISBN 978-1-4503-1115-1.

Fishburn PC, Sarin RK (1994) Fairness and social risk i: Unaggregated analyses. *Management Science* 40(9):1174–1188, URL http://dx.doi.org/10.1287/mnsc.40.9.1174.

Freeman R, Zahedi SM, Conitzer V, Lee BC (2018) Dynamic proportional sharing: A game-theoretic approach. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 2(1):3.

Friedman E, Ghodsi A, Psomas CA (2014) Strategyproof allocation of discrete jobs on multiple machines. *Proceedings of the Fifteenth ACM conference on Economics and Computation*, 529–546 (ACM).

Friedman E, Psomas CA, Vardi S (2017) Controlled dynamic fair division. *Proceedings of the Eighteenth ACM Conference on Economics and Computation*, 461–478, EC '17 (New York, NY, USA: ACM), ISBN 978-1-4503-4527-9, URL http://dx.doi.org/10.1145/3033274.3085123.

Ghodsi A, Zaharia M, Hindman B, Konwinski A, Shenker S, Stoica I (2011) Dominant resource fairness: Fair allocation of multiple resource types. *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, 24–24, NSDI'11.

Guo M, Conitzer V, Reeves DM (2009) Competitive repeated allocation without payments. *International Workshop on Internet and Network Economics*, 244–255 (Springer).

Haitao Cui T, Raju JS, Zhang ZJ (2007) Fairness and channel coordination. *Management Science* 53(8):1303–1314, URL http://dx.doi.org/10.1287/mnsc.1060.0697.

Huh WT, Liu N, Truong VA (2013) Multiresource allocation scheduling in dynamic environments. *Manufacturing & Service Operations Management* 15(2):280–291, URL http://dx.doi.org/10.1287/msom.1120.0415.

Isard M, Prabhakaran V, Currey J, Wieder U, Talwar K, Goldberg A (2009) Quincy: fair scheduling for distributed computing clusters. *Proceedings of the ACM SIGOPS Twenty-Second Symposium on Operating Systems Principles*, 261–276 (ACM).

Karsten F, Slikker M, van Houtum GJ (2015) Resource pooling and cost allocation among independent service providers. *Operations Research* 63(2):476–488, URL http://dx.doi.org/10.1287/opre.2015.1360.

Kash IA, Procaccia AD, Shah N (2014) No agent left behind: Dynamic fair division of multiple resources. *J. Artif. Intell. Res.* 51:579–603, URL http://dx.doi.org/10.1613/jair.4405.

Li B, Li Y (2018) Dynamic fair division problem with general valuations. *arXiv preprint arXiv:1802.05294* .

Moulin H, Stong R (2002) Fair queuing and other probabilistic allocation methods. *Mathematics of Operations Research* 27(1):1–30.

Nash Jr JF (1950) The bargaining problem. *Econometrica: Journal of the Econometric Society* 155–162.

Othman A, Papadimitriou C, Rubinstein A (2014) The complexity of fairness through equilibrium. *Proceedings of the Fifteenth ACM conference on Economics and computation*, 209–226 (ACM).

Pazner EA, Schmeidler D (1978) Egalitarian equivalent allocations: A new concept of economic equity. *The Quarterly Journal of Economics* 671–687.

Popa L, Kumar G, Chowdhury M, Krishnamurthy A, Ratnasamy S, Stoica I (2012) Faircloud: sharing the network in cloud computing. *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 187–198 (ACM).

Segal-Halevi E (2018) Redividing the cake. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, 498–504, URL `http://dx.doi.org/10.24963/ijcai.2018/69`.

Steinhaus H (1948) The problem of fair division. *Econometrica* 16(1).

Thomson W (1983) The fair division of a fixed supply among a growing population. *Mathematics of Operations Research* 8(3):319–326, URL `http://dx.doi.org/10.1287/moor.8.3.319`.

Topaloglu H, Powell WB (2005) A distributed decision-making structure for dynamic resource allocation using nonlinear functional approximations. *Operations Research* 53(2):281–297, URL `http://dx.doi.org/10.1287/opre.1040.0166`.

Walsh T (2011) Online cake cutting. *Algorithmic Decision Theory*, 292–305 (Springer).

Wang W, Li B, Liang B (2014) Dominant resource fairness in cloud computing systems with heterogeneous servers. *INFOCOM, 2014 Proceedings IEEE*, 583–591 (IEEE).

8. Summary of results

We summarize our main results in Table 2.

Control vectors	**Bound on the fairness ratio**
$d-$uniform, $d \geq 1$	$\left((d+1)\ln\left(\frac{d+1}{d}\right)\right)^{-1}$ (tight)
$1-$gap	$2\,(3\ln 3)^{-1}$ (tight)
$2-$gap	$3\,(4\ln 4)^{-1}$ (tight)
$c-$gap, $c \geq 3$	$(H_{c+1})^{-1} \geq \text{FAIRNESS}(c-\text{gap}) \geq \left(H_{2c+3} - \frac{1}{2}\right)^{-1}$
	$(c+1)\left((c+2)\ln(c+2)\right)^{-1}$ (asymptotic bound, tight)

Table 2 Results for single resource dynamic fair division. H_n denotes the n-th harmonic number.

9. Figures

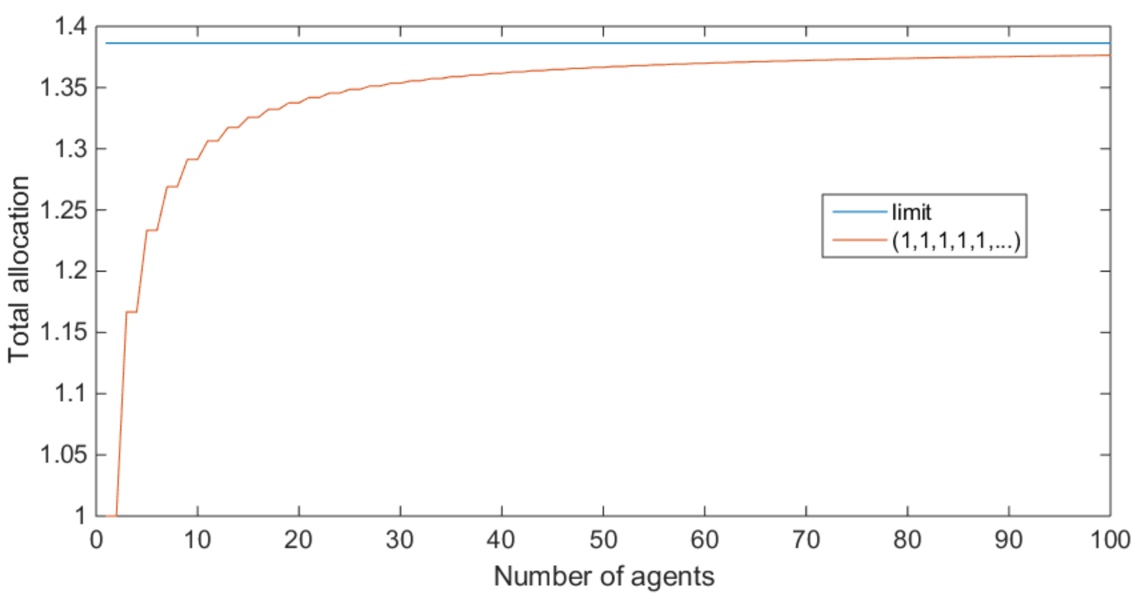

Figure 1 Allocations of FFA for $1-$uniform control vectors and $\sigma = 1$.

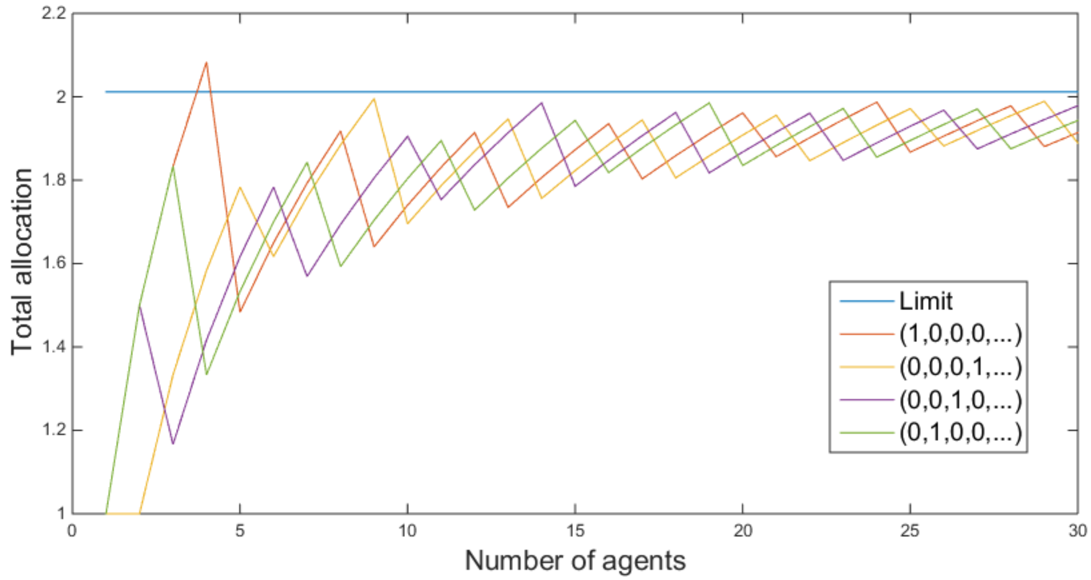

Figure 2 Allocations of FFA for the basic 2−gap control vectors with $\sigma = 1$.

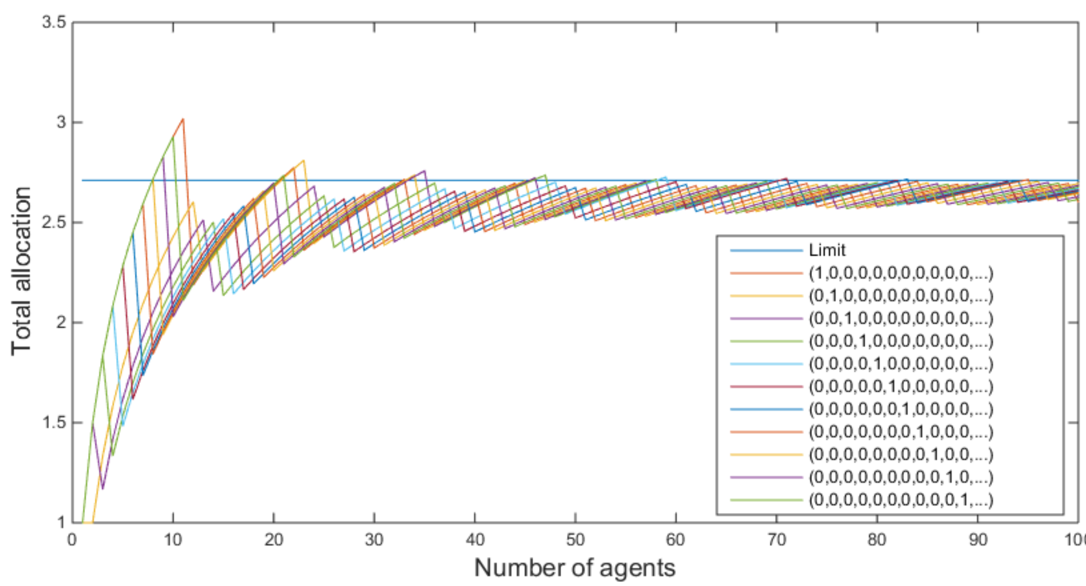

Figure 3 Allocations of FFA for the basic 10−gap control vectors with $\sigma = 1$.

Appendix A: Number theory

We require the following well-known inequalities for the n^{th} harmonic number H_n, (where $\gamma' = 0.57721\dots$ is the Euler-Mascheroni constant):

$$H_n \geq \ln n + \gamma' + \frac{1}{2n} - \frac{1}{12n^2}, \tag{1}$$
$$H_n \leq \ln n + \gamma' + \frac{1}{2n}, \tag{2}$$
$$H_n \geq \ln n + \gamma' + \frac{1}{2n} - \frac{1}{12n^2}. \tag{3}$$

Using these, it is easy to derive the following:

LEMMA 4. *For any natural numbers $b > a > 1$,*

$$\ln(b) - \ln(a-1) - \frac{1}{2a-2} \leq \sum_{x=a}^{b} \frac{1}{x} \leq \ln(b) - \ln(a-1).$$

Proof. We only prove the upper bound; the proof for the lower bound is similar. From Inequalities (2) and (3) (for the lower bound proof, use (1) and (2)),

$$\begin{aligned}\sum_{x=a}^{b} \frac{1}{x} &= \sum_{x=1}^{b} \frac{1}{x} - \sum_{x=1}^{a-1} \frac{1}{x} \\ &\leq ln(b) + \gamma' + \frac{1}{2b} - \left(ln(a-1) + \gamma' + \frac{1}{2(a-1)} - \frac{1}{12(a-1)^2} \right) \\ &\leq \ln(b) - \ln(a-1),\end{aligned}$$

for $b > a > 1$. □

LEMMA 5. *The function*

$$f(t) = \sum_{i=t}^{(t-1)(c+2)+c+1} \frac{1}{i}$$

is monotone decreasing in t, for integer values of t and integer $c > 0$.

Proof.

$$\begin{aligned}f(t) - f(t+1) &= \sum_{i=t}^{(t-1)(c+2)+c+1} \frac{1}{i} - \sum_{i=t+1}^{t(c+2)+c+1} \frac{1}{i} \\ &= \frac{1}{t} - \sum_{i=t(c+2)}^{t(c+2)+c+1} \frac{1}{i} \\ &> \frac{1}{t} - (c+2)\frac{1}{t(c+2)} = 0.\end{aligned}$$

□

Lemma 6. *For integer $c>1$ and integer $t\geq 2$,*

$$\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)}\leq 1/2.$$

Proof. It suffices to prove the Lemma for $c=2$, as the sum decreases as c increases.

$$\begin{aligned}\sum_{i=0}^{t-2}\frac{1}{t+3i} &= \frac{1}{t}+\sum_{i=1}^{t-2}\frac{1}{t+3i}\\ &\leq \frac{1}{t}+\int_0^{t-2}\frac{1}{t+3x}dx\\ &= \frac{1}{t}+\frac{1}{3}\ln(4-6/t)\\ &\leq \frac{1}{t}+\frac{1}{3}\ln 4\\ &\leq 1/2,\end{aligned}$$

for $t\geq 27$. It is easy to computationally verify the Lemma holds for smaller t. □

Appendix B: Supplementary material for Section 5.1

B.1. Proof of Proposition 1

Proposition 1. *Let* X^t *denote the allocation of* FFA *when there are t users in the system, when* FFA*'s inputs are the infinite $d-$uniform control vector and σ. For all $t=\ell\cdot(d+1)+i$, $i\in[0,d]$,*

$$\mathrm{X}^t=\left(\underbrace{\frac{\sigma}{t-\ell},\ldots,\frac{\sigma}{t-\ell}}_{i\ terms},\underbrace{\frac{\sigma}{t-\ell+1},\ldots,\frac{\sigma}{t-\ell+1}}_{d+1\ terms},\ldots,\underbrace{\frac{\sigma}{t-1},\ldots,\frac{\sigma}{t-1}}_{d+1\ terms},\underbrace{\frac{\sigma}{t},\ldots,\frac{\sigma}{t}}_{d+1\ terms}\right).$$

Proof. The proof is by induction on ℓ. For the base case, $\ell=0$, i.e., $t=i$, $i\in[0,d]$, there are at least as many donors as users, and so the allocation is

$$\mathrm{X}^i=\left(\underbrace{\frac{\sigma}{t-\ell},\ldots,\frac{\sigma}{t-\ell}}_{i\text{ terms}}\right).$$

For the inductive step, assume the statement holds for some ℓ and all $i\in[0,d]$. Specifically, for $i=d$, we have

$$\begin{aligned}\mathrm{X}^{\ell(d+1)+d}&=\left(\underbrace{\frac{\sigma}{t-\ell},\ldots,\frac{\sigma}{t-\ell}}_{d\text{ terms}},\underbrace{\frac{\sigma}{t-\ell+1},\ldots,\frac{\sigma}{t-\ell+1}}_{d+1\text{ terms}},\ldots,\underbrace{\frac{\sigma}{t-1},\ldots,\frac{\sigma}{t-1}}_{d+1\text{ terms}},\underbrace{\frac{\sigma}{t},\ldots,\frac{\sigma}{t}}_{d+1\text{ terms}}\right)\\ &=\left(\underbrace{\frac{\sigma}{\ell d+d},\ldots,\frac{\sigma}{\ell d+d}}_{d\text{ terms}},\underbrace{\frac{\sigma}{\ell d+d+1},\ldots,\frac{\sigma}{\ell d+d+1}}_{d+1\text{ terms}},\ldots,\underbrace{\frac{\sigma}{\ell(d+1)+d},\ldots,\frac{\sigma}{\ell(d+1)+d}}_{d+1\text{ terms}}\right).\end{aligned}$$

When the next user arrives, FiniteFFA disrupts the d users with the largest shares, so the next allocations will be:

$$\mathrm{X}^{(\ell+1)(d+1)} = \left(\underbrace{\frac{\sigma}{\ell d+d+1},\dots,\frac{\sigma}{\ell d+d+1}}_{d+1 \text{ terms}},\dots,\underbrace{\frac{\sigma}{\ell(d+1)+d+1},\dots,\frac{\sigma}{\ell(d+1)+d+1}}_{d+1 \text{ terms}} \right)$$
$$= \left(\underbrace{\frac{\sigma}{(\ell+1)(d+1)-\ell},\dots,\frac{\sigma}{(\ell+1)(d+1)-\ell}}_{d+1 \text{ terms}},\dots,\underbrace{\frac{\sigma}{(\ell+1)(d+1)},\dots,\frac{\sigma}{(\ell+1)(d+1)}}_{d+1 \text{ terms}} \right),$$

as required. It is now straightforward to confirm that the characterization holds for $i \geq 1$, since d of the first $d+1$ terms are disrupted, and $d+1$ new terms are added at the end. □

B.2. Proof of Lemma 2

Lemma 2. *Let ψ^N denote a $d-$uniform control vector of length N, $d \geq 1$. Then,* FAIRNESS(FFA, ψ^N) *is monotone non-increasing in N.*

Proof. Towards a contradiction, let $M > N$ and assume FAIRNESS(FFA, ψ^N) $= \sigma < \sigma' =$ FAIRNESS(FFA, ψ^M). Denote by $\mathrm{X} = \{\mathrm{X}^1,\dots,\mathrm{X}^N,\mathrm{X}^{N+1},\dots,\mathrm{X}^M\}$ the allocations of FFA on ψ^M. We can simply set the allocations of FFA on ψ^N to be $\{\mathrm{X}^1,\dots,\mathrm{X}^N\}$ in order to obtain a better fairness ratio: the resource allocated is never greater than 1 and the number of donors used in each step is legal. This is in contradiction to the optimality of FFA. □

Appendix C: Supplementary material for Section 5.2

C.1. Proof of Theorem 7.

Theorem 7. *The optimal fairness ratios for* $1-$ *and* $2-$gap *control vectors are*

1. FAIRNESS$(1-\text{gap}) = 2(3\ln 3)^{-1}$,
2. FAIRNESS$(2-\text{gap}) = 3(4\ln 4)^{-1}$.

There are two basic $1-$gap control vectors: $(1,0,1,0,\dots)$ and $(0,1,0,1,\dots)$. There are three basic $2-$gap control vectors (Example 3). We prove the bound in Theorem 7 for each basic control vector separately. The theorem then follows from Lemma 3. Here we only present the proof for one of the two basic $1-$gap control vectors. We omit the proof for the other control vector as it is virtually identical. Furthermore, we only present the proof for $c=1$; the proof for $c=2$ is by similar (slightly more involved) case analysis.

Proposition 6. *The allocations created by* FiniteFFA*, with basic $1-$gap control vector $\psi^1 = (0,1,\dots)$ (from step 6 onward) are*

1. *At time* $t = 0 \pmod 6$*:* $\frac{\sigma}{(t/3)+1}, \frac{\sigma}{(t/3)+2}, \frac{\sigma}{(t/3)+2}, \frac{\sigma}{(t/3)+3}, \dots, \frac{\sigma}{t-2}, \frac{\sigma}{t-2}, \frac{\sigma}{t-1}, \frac{\sigma}{t}, \frac{\sigma}{t}.$

step	allocation for $(0,1,\dots)$	sum	allocation for $(1,0,\dots)$	sum
1	σ	σ	σ	σ
2	$\frac{\sigma}{2},\frac{\sigma}{2}$	σ	$\sigma,\frac{\sigma}{2}$	$\frac{90\sigma}{60}$
3	$\frac{\sigma}{2},\frac{\sigma}{2},\frac{\sigma}{3}$	$\frac{80\sigma}{60}$	$\frac{\sigma}{2},\frac{\sigma}{3},\frac{\sigma}{3}$	$\frac{70\sigma}{60}$
4	$\frac{\sigma}{2},\frac{\sigma}{3},\frac{\sigma}{4},\frac{\sigma}{4}$	$\frac{80\sigma}{60}$	$\frac{\sigma}{2},\frac{\sigma}{3},\frac{\sigma}{3},\frac{\sigma}{4}$	$\frac{85\sigma}{60}$
5	$\frac{\sigma}{2},\frac{\sigma}{3},\frac{\sigma}{4},\frac{\sigma}{4},\frac{\sigma}{5}$	$\frac{92\sigma}{60}$	$\frac{\sigma}{3},\frac{\sigma}{3},\frac{\sigma}{4},\frac{\sigma}{5},\frac{\sigma}{5}$	$\frac{79\sigma}{60}$
6	$\frac{\sigma}{3},\frac{\sigma}{4},\frac{\sigma}{4},\frac{\sigma}{5},\frac{\sigma}{6},\frac{\sigma}{6}$	$\frac{82\sigma}{60}$	$\frac{\sigma}{3},\frac{\sigma}{3},\frac{\sigma}{4},\frac{\sigma}{5},\frac{\sigma}{5},\frac{\sigma}{6}$	$\frac{89\sigma}{60}$

Table 3 Allocations for the first 6 steps of FFA with basic 1−gap control vectors and some σ as input.

2. *At time* $t=2 \pmod 6$*:* $\frac{\sigma}{((t+1)/3)+1},\frac{\sigma}{((t+1)/3)+2},\frac{\sigma}{((t+1)/3)+2},\dots,\frac{\sigma}{t},\frac{\sigma}{t}$.
3. *At time* $t=4 \pmod 6$*:* $\frac{\sigma}{((t+2)/3)},\frac{\sigma}{((t+2)/3)+1},\frac{\sigma}{((t+2)/3)+2},\frac{\sigma}{((t+2)/3)+2},\dots,\frac{\sigma}{t},\frac{\sigma}{t}$.

On odd time steps t*, add* $\frac{\sigma}{t}$ *to the previous step (note the denominators have "changed" relative to the new time* t*).*

Proof. The proof is by induction on the time t. The base case (the first 6 time steps) appears in Table 3. The move from even to odd steps is immediate, as there is no donor. Because only the user with the highest utility has her allocation decreased, it is easy to verify the transition from odd to even steps, by renaming the denominators. For example, in the inductive step, for the case of $t=0 \pmod 6$, we know that $t-1=5 \pmod 6$, and by the inductive hypothesis, the allocation on step $t-1$ is equal to: $\frac{\sigma}{(t/3)},\frac{\sigma}{(t/3)+1},\frac{\sigma}{(t/3)+2},\frac{\sigma}{(t/3)+2},\frac{\sigma}{(t/3)+3},\frac{\sigma}{(t/3)+4},\frac{\sigma}{(t/3)+4},\dots,\frac{\sigma}{t-2},\frac{\sigma}{t-2},\frac{\sigma}{t-1}$ (we get this by plugging in $t=t-2$ in the "$t=4 \pmod 6$" allocation and adding $\frac{\sigma}{t-1}$ at the end). At step t we have a disruption, therefore the leading term $\frac{\sigma}{(t/3)}$ is disrupted, and we have two additional $\frac{\sigma}{t}$ terms. The remaining cases are virtually identical. □

In order to compute the optimal fairness ratio of $\psi^1=(0,1,0,1,\dots)$, we bound the sum of allocations at odd steps, since each odd step uses strictly more resources than the previous even step.

We bound the size of $\text{BANK}(\psi^1,t)$, noting for all $t>0$, $\text{BANK}(\psi^1,t)<1$. On odd steps, we take $\frac{\sigma}{i}$ from the bank. On all even steps except the second and fourth, we return some resource to the bank. (Note that to be able to reach step 5, we need $\frac{23\sigma}{15}$ in the bank; this immediately implies that $\sigma \le \frac{15}{23}$.) First, we show that the resource allocated monotonically increases, when we look at it through a slightly wider lens:

Lemma 7. *Denote by* $\mathrm{S}(\psi^1,t)$ *the total resource allocated at time* t *by* FFA *with input* ψ^1 *and* σ*. For any* $t=0 \pmod 6$, $\mathrm{S}(\psi^1,t+5)=\max_{i\in\{0,\ldots,5\}}\mathrm{S}(\psi^1,t+i)$*. Furthermore,* $\mathrm{S}(\psi^1,t+5)$ *is monotone non-decreasing in* t *for* $t=0 \pmod 6$*.*

Proof. Let $t=0 \pmod 6$. It is easy to verify the following using Proposition 6:

1. At the $t+1$-st arrival we add $\frac{\sigma}{t+1}$ to $\mathrm{S}(\psi^1,t)$.
2. At the $t+2$-nd arrival we add $\frac{2\sigma}{t+2}$ and subtract $\frac{\sigma}{t/3+1}$ from $\mathrm{S}(\psi^1,t+1)$.
3. At the $t+3$-rd arrival we add $\frac{\sigma}{t+3}$ to $\mathrm{S}(\psi^1,t+2)$.
4. At the $t+4$-th arrival we add $\frac{2\sigma}{t+4}$ and subtract $\frac{\sigma}{t/3+2}$ from $\mathrm{S}(\psi^1,t+3)$.
5. At the $t+5$-th arrival we add $\frac{\sigma}{t+5}$ to $\mathrm{S}(\psi^1,t+4)$.

Clearly, $\mathrm{S}(\psi^1,t+5)\geq\mathrm{S}(\psi^1,t+4)$, $\mathrm{S}(\psi^1,t+3)\geq\mathrm{S}(\psi^1,t+2)$ and $\mathrm{S}(\psi^1,t+1)\geq\mathrm{S}(\psi^1,t)$. It is easy to verify using simple calculus that $\mathrm{S}(\psi^1,t+5)=\mathrm{S}(\psi^1,t+4)+\frac{\sigma}{t+5}=\mathrm{S}(\psi^1,t+3)+\frac{2\sigma}{t+4}-\frac{\sigma}{t/3+2}+\frac{\sigma}{t+5}$ is larger than $\mathrm{S}(\psi^1,t+3)$ for all t. Also, $\mathrm{S}(\psi^1,t+5)=\mathrm{S}(\psi^1,t+1)+\frac{\sigma}{t+3}+\frac{2\sigma}{t+2}-\frac{\sigma}{t/3+1}+\frac{2\sigma}{t+4}-\frac{\sigma}{t/3+2}+\frac{\sigma}{t+5}\geq\mathrm{S}(\psi^1,t+1)$. Therefore, $\mathrm{X}(\psi^1,t+5)$ is the allocation with the maximum resource out of the 6 allocations $\mathrm{S}(\psi^1,t+i):i\in\{0,\ldots,5\}$.

Furthermore, $\mathrm{S}(\psi^1,t+5)=\mathrm{S}(\psi^1,t-1)+\frac{\sigma}{t+1}+\frac{2\sigma}{t}-\frac{\sigma}{t/3}+\frac{\sigma}{t+3}+\frac{2\sigma}{t+2}-\frac{\sigma}{t/3+1}+\frac{2\sigma}{t+4}-\frac{\sigma}{t/3+2}+\frac{\sigma}{t+5}$ is greater than $\mathrm{S}(\psi^1,t-1)$, for all $t\geq 2$. Since $t-1=5 \pmod 6$, this implies that $\mathrm{S}(\psi^1,t)$ for $t=5 \pmod 6$ is monotone non-decreasing. □

We now complete the proof of Theorem 7 (for the control vector $(0,1,0,1,0,\ldots)$).

Proof of Theorem 7. From Lemma 7, the amount of resource allocated increases with t; therefore it suffices to bound the total allocation when t goes to infinity. Similarly, it suffices to analyze the case when the allocation is larger than all previous allocations, $t=5 \pmod 6$. At time $t=0 \pmod 6$, the total resource allocated is

$$\sigma\left(2\sum_{i=1}^{t/3}\frac{1}{t/3+2i}+\sum_{i=1}^{t/3}\frac{1}{t/3+2i-1}\right)\approx\sigma\left(\frac{3}{2}\sum_{i=t/3}^{t}\frac{1}{i}\right)\approx\sigma\frac{3\ln 3}{2}.$$

It must hold that $\sigma\frac{3\ln 3}{2}\leq 1$, and the allocations at times $t=0 \pmod 6$ and $t=5 \pmod 6$ are asymptotically the same. The theorem follows. □

C.2. Proof of Proposition 2.

Proposition 2. *For all* $1\leq i\leq k-1$ *and all* $t=t_i^* \pmod{c+1}$*,*

$$\mathcal{X}_{\psi_i}^t\succeq\mathcal{X}_{\psi_{i+1}}^t.$$

Proof. For clarity, instead of $\mathcal{X}_{\psi_i}^t$, we will use $\mathcal{X}(\psi_i,t)$. Recall that t_i^* is the leftmost coordinate on which ψ_i and ψ differ, hence the leftmost coordinate on which ψ_i and ψ_{i+1} differ. If $t<t_i^*$, the proposition trivially holds because the two allocations are identical up until step t_i^*-1. We prove

the claim for $t = t_i^* + j(c+1)$ by induction on j. For the base case, $j = 0$, denote $\mathcal{X}(\psi_i, t_i^* - 1) = \mathcal{X}(\psi_{i+1}, t_i^* - 1) = (a_1, a_2, \dots, a_{t_i^*-1})$.

Since $\psi_i[t_i^*] = 0$ and $\psi_{i+1}[t_i^*] = 1$, we have

$$\mathcal{X}(\psi_i, t_i^*) = (a_1, a_2, \dots, a_{t_i^*-2}, a_{t_i^*-1}, \sigma/t_i^*),$$
$$\mathcal{X}(\psi_{i+1}, t_i^*) = (a_2, a_3, \dots, a_{t_i^*-1}, \sigma/t_i^*, \sigma/t_i^*).$$

As $\mathcal{X}(\psi_i, t_i^* - 1)$ is sorted (hence $a_\ell \geq a_{\ell+1}$ for all ℓ), and $a_{t_i^*-1} = \frac{\sigma}{t_i^*-1} > \frac{\sigma}{t_i^*}$, this completes the proof of the base case.

For the inductive step, let $\mathcal{X}(\psi_i, t) = (a_1, a_2, \dots, a_t)$ and $\mathcal{X}(\psi_{i+1}, t) = (b_1, b_2, \dots, b_t)$. By the inductive hypothesis $\mathcal{X}(\psi_i, t) \succeq \mathcal{X}(\psi_{i+1}, t)$, i.e., $a_i \geq b_i$, for all $i = 1, \dots, t$. We will show that $\mathcal{X}(\psi_i, t + c + 1) \succeq \mathcal{X}(\psi_{i+1}, t + c + 1)$.

Since $t = t_i^* + j(c+1)$, we know that $\psi_i[t] = 0$, $\psi_i[t + c + 1] = 0$, and $\psi_{i+1}[t] = 1$. Furthermore, $\psi_{i+1}[t'] = 0$, for $t < t' < t + c + 1$. Let q_i be the smallest value such that $\psi_i[t + q_i] = 1$.

$$\mathcal{X}(\psi_i, t) = (a_1, \dots, a_{t-1}, a_t),$$
$$\mathcal{X}(\psi_i, t + q_i) = \left(a_2, \dots, a_t, \frac{\sigma}{t+1}, \dots, \frac{\sigma}{t+q_i-1}, \frac{\sigma}{t+q_i}, \frac{\sigma}{t+q_i}\right),$$
$$\mathcal{X}(\psi_i, t + c + 1) = \left(a_2, \dots, a_t, \frac{\sigma}{t+1}, \dots, \frac{\sigma}{t+q_i-1}, \frac{\sigma}{t+q_i}, \frac{\sigma}{t+q_i}, \frac{\sigma}{t+q_i+1}, \dots, \frac{\sigma}{t+c+1}\right).$$

and

$$\mathcal{X}(\psi_{i+1}, t) = (b_1, \dots, b_{t-1}, b_t),$$
$$\mathcal{X}(\psi_{i+1}, t + c + 1) = \left(b_2, \dots, b_t, \frac{\sigma}{t+1}, \dots, \frac{\sigma}{t+c}, \frac{\sigma}{t+c+1}, \frac{\sigma}{t+c+1}\right).$$

For the first $t - 1$ terms $a_i \geq b_i$ by the inductive hypothesis. The next q_i terms are identical ($\frac{\sigma}{t+1}$ through $\frac{\sigma}{t+q_i}$), as well as the very last term. □

C.3. Proof of Proposition 3

Proposition 3. *For all $1 \leq i \leq k - 1$, there exists some (minimal) T_{ψ_i} such that, for all steps $t > T_i^{\max}$, $\mathcal{X}_{\psi_{i+1}}^t$ is identical to $\mathcal{X}_{\psi_i}^t$.*

Proof. Once again, for clarity, instead of $\mathcal{X}_{\psi_i}^t$, we will use $\mathcal{X}(\psi_i, t)$. Consider what the allocation of a basic $c-$gap control vector looks like. Every $c + 1$ arrivals a disruption is allowed, so the "bulk" of the allocation is c single terms of the form $\left(\frac{\sigma}{j}, \frac{\sigma}{j+1}, \dots, \frac{\sigma}{j+c}\right)$, followed by two terms

$\left(\frac{\sigma}{j+c+1}, \frac{\sigma}{j+c+1}\right)$, followed by c singles, followed by a double, and so on. The leading terms (until the first double term) could be any number of singles between zero and $c+1$. (There could be exactly $c+1$ single terms in the following case: the first term of the allocation in the previous step was a double term (followed by c singles), and the next step had a disruption.) Similarly, the last terms could be any number of singles between zero and $c+1$. Knowing the specific basic $c-$gap control vector essentially only determines the steps in which the double terms appear.

Let p be number of 0s between t_i^* and the previous 1. In Example 5, for ψ_4, $p=1$. Then, $\mathcal{X}(\psi_{i+1}, t_i^*) = \left(a_1, \ldots, a_m, \frac{\sigma}{t_i^*-p-1}, \frac{\sigma}{t_i^*-p-1}, \frac{\sigma}{t_i^*-p}, \frac{\sigma}{t_i^*-p+1}, \ldots, \frac{\sigma}{t_i^*-1}, \frac{\sigma}{t_i^*}, \frac{\sigma}{t_i^*}\right)$. T_{ψ_i} is exactly the smallest time in which the first term of $\mathcal{X}(\psi_{i+1}, T_{\psi_i})$ is $\frac{\sigma}{t_i^*-p-1}$ and the second term is $\frac{\sigma}{t_i^*-p}$, i.e. only the first one of the $\frac{\sigma}{t_i^*-p-1}$ terms in $\mathcal{X}(\psi_{i+1}, t_i^*)$ has been disrupted. So, we'd like for the first $m+1$ terms of the t_i^* terms above to be disrupted. Solving for m we get that $m = t_i^* - p - 4$ (we subtract the two double terms $\frac{\sigma}{t_i^*}$ and $\frac{\sigma}{t_i^*-p-1}$, and the p single terms), hence $T_{\psi_i} = t_i^* + (c+1)(t_i^* - p - 3)$. □

It might seem that the precision of the computation of T_{ψ_i} in the above proof is unnecessary. It is easy to verify that the proof of Proposition 4 does not hold for looser bounds.

C.4. Proof of Proposition 4

PROPOSITION 4. *For $1 \le i \le k-1$, let T_{ψ_i} be as in Proposition 3. For all $t = t_i^* \pmod{c+1}$, $t \in [t_i^*, T_{\psi_i}]$,* BANK$(\psi_{i+1}, t) \ge \sum_{j=1}^{c} \frac{\sigma}{t+j}$.

Proof. Again, instead of $\mathcal{X}_{\psi_i}^t$, we will use $\mathcal{X}(\psi_i, t)$. Let $t = t_i^* + \ell(c+1)$. Let q_i be the smallest positive integer such that $\psi_i^N[t_i^* + q_i] = 1$. Clearly, $q_i \le c$, as ψ has at most c consecutive zeros. Note further that since $\psi_i^N[t_i^* + q_i] = 1$ and since $t_i^* \ge 2$, it must hold that $t_i^* + q_i \ge c+2$.

Observe that if $\mathcal{X}(\psi_i^N, t_i^*) = (a_1, a_2, \ldots, a_{t_i^*-2}, a_{t_i^*-1}, \sigma/t_i^*)$, then

$$\mathcal{X}(\psi_i^N, t_i^* + q_i) = \left(a_2, \ldots, a_{t_i^*-2}, a_{t_i^*-1}, \frac{\sigma}{t_i^*}, \frac{\sigma}{t_i^*+1}, \ldots, \frac{\sigma}{t_i^*+q_i-1}, \frac{\sigma}{t_i^*+q_i}, \frac{\sigma}{t_i^*+q_i}\right),$$
$$\mathcal{X}(\psi_{i+1}^N, t_i^*) = \left(a_2, \ldots, a_{t_i^*-2}, a_{t_i^*-1}, \frac{\sigma}{t_i^*}, \frac{\sigma}{t_i^*}\right).$$

After $c+1$ more steps the leading term a_2 will appear in neither $\mathcal{X}(\psi_i^N, t_i^* + q_i + c + 1)$ nor $\mathcal{X}(\psi_{i+1}^N, t_i^* + c + 1)$. The first allocation has additional terms $\frac{1}{t_i^*+q_i+1}$ through $\frac{1}{t_i^*+q_i+c}$ and two terms $\frac{1}{t_i^*+q_i+c+1}$. The latter allocation has additional terms $\frac{1}{t_i^*+1}$ through $\frac{1}{t_i^*+c}$ and two terms $\frac{1}{t_i^*+c+1}$. A similar pattern appears $c+1$ steps later, and so on. Denote by $\mathcal{S}(\psi, t)$ the total resource allocated by $\mathcal{X}(\psi, t)$. Therefore, at step t, we have (for some k):

$$\mathcal{S}(\psi_{i+1}^N, t) = a_k + \cdots + a_{t_i^*-1} + \underbrace{\left(\frac{1}{t_i^*+1} + \frac{1}{t_i^*+2} + \cdots + \frac{1}{t}\right)}_{\text{“single terms” plus one of the two “double terms”}} + \underbrace{\left(\frac{1}{t_i^*} + \frac{1}{t_i^*+c+1} + \cdots + \frac{1}{t}\right)}_{\text{remaining “double terms”}}$$

$$\mathcal{S}(\psi_i^N, t+q_i) = a_k + \cdots + a_{t_i^*-1} + \sum_{j=t_i^*+1}^{t+q_i} \frac{1}{j} + \left(\frac{1}{t_i^*+q_i} + \frac{1}{t_i^*+c+1+q_i} + \cdots + \frac{1}{t+q_i} \right).$$

Their difference (which is also the difference of their "banks") is:

$$\mathcal{S}(\psi_i^N, t+q_i) - \mathcal{S}(\psi_{i+1}^N, t) = \sum_{j=t+1}^{t+q_i} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right)$$

$$\text{BANK}(\psi_{i+1}^N, t) = \text{BANK}(\psi_i^N, t+q_i) + \sum_{j=t+1}^{t+q_i} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right).$$

Since ψ_i^N is feasible, we know that $\text{BANK}(\psi_i^N, t+q_i) \geq \sum_{j=1}^{c} \frac{1}{t+q_i+j}$. Our goal is to show that $\text{BANK}(\psi_{i+1}^N, t) \geq \sum_{j=1}^{c} \frac{1}{t+j}$. To that end, we prove the following:

$$\sum_{j=1}^{c} \frac{1}{t+q_i+j} + \sum_{j=t+1}^{t+q_i} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right) \geq \sum_{j=1}^{c} \frac{1}{t+j} = \sum_{j=t+1}^{t+c} \frac{1}{j}$$

This is equivalent to the following two inequalities, and we prove the latter.

$$\sum_{j=t+q_i+1}^{t+q_i+c} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right) \geq \sum_{j=t+q_i+1}^{t+c} \frac{1}{j}$$

$$\sum_{j=t+c+1}^{t+c+q_i} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right) \geq 0.$$

Let

$$f(t_i^*, \ell, c, q_i) = \sum_{j=t+c+1}^{t+c+q_i} \frac{1}{j} + \sum_{j=0}^{\ell} \left(\frac{1}{t_i^*+q_i+j(c+1)} - \frac{1}{t_i^*+j(c+1)} \right),$$

where $t = t_i^* + \ell(c+1)$. We know that $c \geq q_i \geq 1$, $t + q_i \geq c+2$, and $t_i^* + \ell(c+1) \leq T_{\psi_i} = t_i^* + (c+1)(t_i^* - p - 3)$, where p is the number of zeros between t_i^* and the previous 1 in ψ_N. Therefore, $p + q_i = c$, which implies that $\ell \leq t_i^* + q_i - c - 3$. This is the choice of parameters for which we would like to show non-negativity of $f(t_i^*, \ell, c, q_i)$.

As we noted earlier, showing that $f(t_i^*, \ell, c, q_i)$ is non-negative is a very delicate task. For example, using a more coarse upper bound for ℓ, like $\ell \leq t_i^* - 3$, $f(t_i^*, \ell, c, q_i)$ might be negative ($t_i^* = 9$, $\ell = 6$, $c = 4$ and $q_i = 1$.) We first show that the function is non-decreasing in ℓ, in Proposition 7. Then, because f decreases as ℓ increases, it suffices to show non-negativity for the maximum value of ℓ, $t_i^* + q_i - c - 3$. This implies that $t = T_{\psi_i} = t_i^* + \ell(c+1) = t_i^* + (t_i^* + q_i - c - 3)(c+1) = (c+2)t_i^* + (c+1)(q_i - c - 3)$. We overload notation and re-define f as

$$f(t_i^*, c, q_i) = \sum_{j=t_{max}+c+1}^{t_{max}+c+q_i} \frac{1}{j} + \sum_{j=0}^{t_i^*+q_i-c-3} \left(\frac{1}{t_i^* + q_i + j(c+1)} - \frac{1}{t_i^* + j(c+1)} \right).$$

We show in Proposition 8 that $f(t_i^*, c, q_i)$ is non-increasing in c. It therefore remains to show that $f(t_i^*, c, q_i)$ is non-negative in the limit. We can re-write the function as:

$$\begin{aligned}
f(t_i^*, c, q_i) &= \sum_{j=t_{max}+c+1}^{t_{max}+c+q_i} \frac{1}{j} + \sum_{j=0}^{t_i^*+q_i-c-3} \left(\frac{1}{t_i^* + q_i + j(c+1)} - \frac{1}{t_i^* + j(c+1)} \right) \\
&= \sum_{j=t_{max}+c+1}^{t_{max}+c+q_i} \frac{1}{j} + \frac{1}{c+1} \sum_{j=0}^{t_i^*+q_i-c-3} \left(\frac{1}{\frac{t_i^*+q_i}{c+1} + j} - \frac{1}{\frac{t_i^*}{c+1} + j} \right) \\
&= \sum_{j=t_{max}+c+1}^{t_{max}+c+q_i} \frac{1}{j} + \frac{1}{c+1} \left(\sum_{j=\frac{t_i^*+q_i}{c+1}}^{\frac{t_{max}+q_i}{c+1}} \frac{1}{j} - \sum_{j=\frac{t_i^*}{c+1}}^{\frac{t_{max}}{c+1}} \frac{1}{j} \right).
\end{aligned}$$

Applying the approximations for the Harmonic number from Appendix A gives:

$$\begin{aligned}
f(t_i^*, c, q_i) \geq \ln\left(\frac{t_{max} + c + q_i}{t_{max} + c} \right) - \frac{1}{2\,(T_{\psi_i} + c)} + \frac{1}{c+1} \ln\left(\frac{t_{max} + q_i}{t_i^* + q_i - c - 1} \right) \\
- \frac{1}{c+1} \cdot \frac{1}{2\,(t_i^* + q_i - c - 1)} - \frac{1}{c+1} \ln\left(\frac{t_{max}}{t_i^* - c - 1} \right)
\end{aligned}$$

Taking the limit at t_i^* goes to infinity (recall that $t_{max} = (c+2)t_i^* + (c+1)(q_i - c - 3)$) gives

$$\lim_{t_i^* \to \infty} f(t_i^*, c, q_i) = 0 - 0 + \frac{1}{c+1} \ln(c+2) - 0 - \frac{1}{c+1} \ln(c+2) = 0.$$

This completes the proof of Proposition 4. □

PROPOSITION 7. $f(t_i^*, \ell+1, c, q_i) - f(t_i^*, \ell, c, q_i) \leq 0.$

Proof.

$$\begin{aligned}
&f(t_i^*, \ell+1, c, q_i) - f(t_i^*, \ell, c, q_i) = \\
&\qquad = \sum_{j=t+2(c+1)}^{t+c+1+c+q_i} \frac{1}{j} - \sum_{j=t+c+1}^{t+c+q_i} \frac{1}{j} + \frac{1}{t_i^* + q_i + (\ell+1)(c+1)} - \frac{1}{t_i^* + (\ell+1)(c+1)}.
\end{aligned}$$

Every positive term in the first sum is smaller than the corresponding negative term in the second sum (and they have the same number of terms). The second to last positive term is smaller than the last (negative) term. □

PROPOSITION 8. $f(t_i^* + c + 1, c, q_i) - f(t_i^*, c, q_i) \leq 0.$

Proof. Increasing t_i^* by $c+1$ increases t_{max} by $(c+2)(c+1)$.

$$f(t_i^*+c+1,c,q_i)-f(t_i^*,c,q_i)=\sum_{j=t_{max}+(c+2)(c+1)+c+1}^{t_{max}+(c+2)(c+1)+c+q_i}\frac{1}{j}-\sum_{j=t_{max}+c+1}^{t_{max}+c+q_i}\frac{1}{j}$$
$$+\sum_{j=0}^{t_i^*+q_i-2}\left(\frac{1}{t_i^*+q_i+(j+1)(c+1)}-\frac{1}{t_i^*+(j+1)(c+1)}\right)$$
$$-\sum_{j=0}^{t_i^*+q_i-c-3}\left(\frac{1}{t_i^*+q_i+j(c+1)}-\frac{1}{t_i^*+j(c+1)}\right)$$

This can be re-written as:

$$\begin{aligned}
f(t_i^*+c+1,c,q_i)-f(t_i^*,c,q_i)&=\left(\frac{1}{t_{max}+(c+1)(c+2)+c+1}+\cdots+\frac{1}{t_{max}+(c+1)(c+2)+c+q_i}\right)\\
&-\left(\frac{1}{t_{max}+c+1}+\frac{1}{t_{max}+c+2}+\cdots+\frac{1}{t_{max}+c+q_i}\right)\\
&+\left(\frac{1}{t_i^*+q_i+c+1}+\frac{1}{t_i^*+q_i+2(c+1)}+\cdots+\frac{1}{t_{max}+q_i+(c+1)(c+2)}\right)\\
&-\left(\frac{1}{t_i^*+c+1}+\frac{1}{t_i^*+2(c+1)}+\cdots+\frac{1}{t_{max}+(c+1)(c+2)}\right)\\
&-\left(\frac{1}{t_i^*+q_i}+\frac{1}{t_i^*+q_i+c+1}+\cdots+\frac{1}{t_{max}+q_i}\right)\\
&+\left(\frac{1}{t_i^*}+\frac{1}{t_i^*+c+1}+\frac{1}{t_i^*+2(c+1)}+\cdots+\frac{1}{t_{max}}\right).\\
&=\sum_{j=1}^{q_i}\left(\frac{1}{t_{max}+(c+1)(c+2)+c+j}-\frac{1}{t_{max}+c+j}\right)\\
&\quad+\left(\sum_{j=1}^{c+2}\frac{1}{t_{max}+q_i+j(c+1)}\right)-\left(\sum_{j=1}^{c+2}\frac{1}{t_{max}+j(c+1)}\right)\\
&\quad-\frac{1}{t_i^*+q_i}+\frac{1}{t_i^*}\\
&=-\sum_{j=1}^{q_i}\frac{(c+1)(c+2)}{(T_{\psi_i}+(c+1)(c+2)+c+j)\,(T_{\psi_i}+c+j)}\\
&\quad-\sum_{j=1}^{c+2}\frac{q_i}{(T_{\psi_i}+(c+1)j+q_i)\,(T_{\psi_i}+(c+1)j)}+\frac{q_i}{t_i^*(t_i^*+q_i)}\\
&\le-\frac{(c+1)(c+2)q_i}{(T_{\psi_i}+(c+1)(c+2)+c+q_i)\,(T_{\psi_i}+c+q_i)}\\
&\quad-\frac{(c+2)q_i}{(T_{\psi_i}+(c+1)(c+2)+q_i)\,(T_{\psi_i}+(c+1)(c+2))}+\frac{q_i}{t_i^*(t_i^*+q_i)}.
\end{aligned}$$

To verify this by computer to be non-positive, we execute the following code in Mathematica:

```
f[t_, c_, q_] =

q/(t (t + q)) - (c + 2) q/( ( (c + 2) t + (c + 1) (q - c - 3) + (c + 1) (c + 2) )
```

```
* ( (c + 2) t + (c + 1) (q - c - 3) + (c + 1) (c + 2) + q ))
 - q (c + 1) (c + 2)/(((c + 2) t + (c + 1) (q - c - 3) + c + q)
 * ((c + 2) t + (c + 1) (q - c - 3) + c + q + (c + 1) (c + 2)));
Reduce[ {f[t, c, q] <= 0, q <= c, q >= 1, t + q >= c + 2}, t]
```

which gives the following result, thus verifying non-positivity.

```
(c == 1 && q == 1 && t >= 2) || (c > 1 && 1 <= q <= c && t >= 2 + c - q)
```

This completes the Proof of Proposition 8. □

C.5. Proof of Proposition 5

PROPOSITION 5. *Let ψ be any basic $c-$*gap *control vector. Then for any $t>1$,*

$$\text{FAIRNESS}(\text{FFA},\psi,t)\geq\left(\arg\max_{t\in\mathbb{N}^+,j\in\{0,\ldots,c+1\}}\sum_{i=t}^{(t-1)(c+2)+j}\frac{1}{i}+\sum_{i=0}^{t-2}\frac{1}{t+i(c+1)+j}\right)^{-1}.$$

In order to prove Proposition 5 it will be convenient to introduce some additional notation. Elements of an allocation that appear once are called *singletons*, and those that appear twice *doubles*. We use the following to make our notation more compact. Instead of individually analyzing each basic control vector, we define a set of allocations $Z^c=\{Z^c_{t,j}\}$, that simultaneously upper bounds all allocations created by all basic $c-$gap control vectors for any fixed $c\geq 1$.

DEFINITION 3. For any $c\geq 1$,

- $Z^c_{1,1}=(\sigma)$.
- For $j\in\{2,\ldots,c+1\}$,
$$Z^c_{1,j}=\left(\sigma,\frac{\sigma}{2},\ldots,\frac{\sigma}{j}\right).$$
- For $t>1$ and $j=0$, let $t'=(t-1)(c+2)-c$:
$$Z^c_{t,j}=\left(\underbrace{\frac{\sigma}{t},\frac{\sigma}{t}}_{\text{one double}},\underbrace{\frac{\sigma}{t+1},\ldots,\frac{\sigma}{t+c}}_{c\text{ singletons}},\underbrace{\frac{\sigma}{t+c+1},\frac{\sigma}{t+c+1}}_{\text{one double}},\ldots,\underbrace{\frac{\sigma}{t'},\frac{\sigma}{t'}}_{\text{one double}},\underbrace{\frac{\sigma}{t'+1},\ldots,\frac{\sigma}{t'+c}}_{c\text{ singletons}}\right).$$
- For $t>1$ and $j\in\{1,\ldots,c+1\}$, let $t'=(t-1)(c+2)+j-c$.
$$Z^c_{t,j}=\left(\underbrace{\frac{\sigma}{t},\ldots,\frac{\sigma}{t+j-1}}_{j\text{ singletons}},\underbrace{\frac{\sigma}{t+j},\frac{\sigma}{t+j}}_{\text{one double}},\underbrace{\frac{\sigma}{t+j+1},\ldots,\frac{\sigma}{t+j+c}}_{c\text{ singletons}},\ldots,\underbrace{\frac{\sigma}{t'},\frac{\sigma}{t'}}_{\text{one double}},\underbrace{\frac{\sigma}{t'+1},\ldots,\frac{\sigma}{t'+c}}_{c\text{ singletons}}\right).$$

EXAMPLE 6. $Z^c_{t,j}$ for some possible values of c,t,j:

1. $Z^2_{3,2} = \left(\frac{\sigma}{3}, \frac{\sigma}{4}, \frac{\sigma}{5}, \frac{\sigma}{5}, \frac{\sigma}{6}, \frac{\sigma}{7}, \frac{\sigma}{8}, \frac{\sigma}{8}, \frac{\sigma}{9}, \frac{\sigma}{10}\right)$.
2. $Z^3_{4,0} = \left(\frac{\sigma}{4}, \frac{\sigma}{4}, \frac{\sigma}{5}, \frac{\sigma}{6}, \frac{\sigma}{7}, \frac{\sigma}{8}, \frac{\sigma}{8}, \frac{\sigma}{9}, \frac{\sigma}{10}, \frac{\sigma}{11}, \frac{\sigma}{12}, \frac{\sigma}{12}, \frac{\sigma}{13}, \frac{\sigma}{14}, \frac{\sigma}{15}\right)$.

We now show that Z^c is the set of all possible allocations in the round just before a donor is used.

PROPOSITION 9. *For any $c \geq 1$, the set of allocations $Z^c = \{Z^c_{t,j} : t \geq 1, 0 \leq j \leq c+1\}$ is precisely the set of all possible allocations of* FFA *in the round before a donor is used, when the inputs to* FFA *are a basic $c-$*gap *control vector and σ.*

Proof. For $t = 1, j \in [0, \ldots, c+1]$, the corresponding basic $c-$gap control vector is the one with a prefix of exactly j zeros followed by one.

For $t > 1$, first notice that even though there are $c+1$ basic $c-$gap control vectors, there are $c+2$ possible allocations where $\frac{\sigma}{t}$ is the largest share, since one of the basic $c-$gap control vectors gives an allocation with two terms equal to $\frac{\sigma}{t}$. At some step the first of the two will be the largest share (and equal to the second largest share) and after $c+1$ arrivals it will get disrupted, resulting in an allocation that again has largest share equal to $\frac{\sigma}{t}$. Thus, it is easy to confirm that our allocations $Z^c_{t,j}$ start with the correct pattern.

It remains to show that we have the correct number of total terms, i.e., an allocation which starts with j singletons has a total of $t' + c = (t-1)(c+2) + j$ terms. Towards this statement, let ℓ be the number of times the pattern "one double followed by c singletons" appears (in $Z^2_{3,2}$ in Example 6, $\ell = 2$). This pattern has $c+2$ terms, therefore $t' + c = j + \ell(c+2) \implies t' = \ell(c+2) + j - c$. A different way to count the total number of terms $t' + c$ is to notice that the denominator of the singleton right before one double increases by $c+1$ every time the pattern "one double followed by c singletons" appears. Therefore, the last denominator $t' + c$ is equal to $t + j - 1 + \ell(c+1)$, which (together with the previous equality) implies that $\ell = t-1$, and thus $t' = (t-1)(c+2) + j - c$. □

We can now prove Proposition 5.

Proof of Proposition 5. The allocation just before a donor is used is necessarily greater than the previous c allocations, as there was no donor for the previous c rounds. Therefore, by Proposition 9, the maximal allocation of $Z^c_{t,j}$ is an upper bound on the maximal allocation of FFA, with a basic $c-$gap control vector. The total allocation of $Z^c_{t,j}$ is

$$\sum_{i=t}^{(t-1)(c+2)+j} \frac{\sigma}{i} + \sum_{i=0}^{t-2} \frac{\sigma}{t + i(c+1) + j},$$

by straightforward summation over the allocation vector of Definition 3, where the first is a sum of all the values appearing in the allocation vector and the second part is a sum of the values that have duplicates. □