Building Sequential Resource Allocation Mechanisms without Payments

Anonymous authors
Paper under double-blind review

Keywords: Sequential resource allocation, mechanism design, incentive compatibility.

Summary

We study allocating divisible resources of limited quantities to agents who submit requests for the resources one or multiple times over a finite horizon. This is referred to as the sequential resource allocation problem, as irrevocable allocations need to be made as the requests arrive, without observations on the future requests. Existing works on sequential resource allocation (in the payment-free setting) mainly focus on optimizing social welfare and design mechanisms under the assumption that the agents make truthful requests. Such mechanisms can be easily exploitable – strategic agents may misreport their requests and inflate their allocations. Our aim in this work is to design sequential resource allocation mechanisms that balance the competing objectives of social welfare maximization (promoting the overall agent satisfaction) and incentive compatibility (ensuring that the agents do not have incentives to misreport). We do not design these mechanisms from scratch. As the incentive compatible mechanism design problem has been well studied in the *one-shot* setting (horizon length equals one), we propose a general *meta-algorithm* of transforming a one-shot mechanism into its sequential counterpart. The meta-algorithm can plug in any one-shot mechanism and approximately carry over the properties that the one-shot mechanism already satisfies to the sequential setting. We establish theoretical results validating these claims and also illustrate the superior performance of the proposed method through numerical simulations.

Contribution(s)

1. We propose a meta-algorithm, which we name **S**equential **A**llocation **M**eta **A**lgorithm (`SAMA`), which can be regarded as a general framework for reducing a sequential resource allocation problem into a series of one-shot problems. The key feature of `SAMA` is that it accounts for past allocation and unobserved future requests – agents with greater past allocations are more discounted against in the current round, and resources are withheld for future requests based on a confidence bound. We mathematically show that if the one-shot mechanism optimizes NSW and/or achieves incentive compatibility (IC) in the one-shot sense, `SAMA` approximately carries over the properties to the sequential setting. This implies that with suitable one-shot mechanisms plugged in as the building block, `SAMA` enjoys both approximate NSW and IC guarantees at the same time. To our knowledge, this is the first time such a result has been established for a sequential mechanism in the payment-free setting.
   Context: Prior papers on sequential resource allocation do not consider achieving IC and assume that the agents report their requests truthfully. The existing work that considers optimizing IC jointly with other metrics including social welfare and efficiency is only for the one-shot setting, in which the supplier fully observes all requests before making an allocation.
2. We numerically illustrate the superior performance of `SAMA` and its approximate NSW and IC preserving properties, with a few established one-shot mechanisms as the building block. Specifically, we plug in 1) the Proportional Fairness (PF) mechanism, which achieves the maximum possible NSW but severely violates IC, 2) the Partial Allocation (PA) mechanism, designed by Cole et al. (2013) to be exactly IC at the cost of a substantial reduction to NSW, 3) ExS-Net, which is a learned neural-network-parameterized mechanism proposed in Zeng et al. (2024b) that achieves near-optimal NSW and approximate IC simultaneously.
   Context: None.

Building Sequential Resource Allocation Mechanisms without Payments

Anonymous authors
Paper under double-blind review

Abstract

We study allocating divisible resources of limited quantities to agents who submit requests for the resources one or multiple times over a finite horizon. This is referred to as the sequential or online resource allocation problem, as irrevocable allocations need to be made as the requests arrive, without observations on the future requests. The existing work on sequential resource allocation (in the payment-free setting) mainly focuses on optimizing social welfare and designs mechanisms under the assumption that the agents make truthful requests. Such mechanisms can be easily exploitable – strategic agents may misreport their requests to inflate their allocations. Our aim in this work is to design sequential resource allocation mechanisms that balance the competing objectives of social welfare maximization (promoting the overall agent satisfaction) and incentive compatibility (ensuring that the agents do not have incentives to misreport). We do not design these mechanisms from scratch. Instead, as the incentive compatible mechanism design problem has been well studied in the *one-shot* setting (horizon length equals one), we propose a general *meta-algorithm* of transforming a one-shot mechanism into its sequential counterpart. The meta-algorithm can plug in any one-shot mechanism and approximately carry over the properties that the one-shot mechanism already satisfies to the sequential setting. We establish theoretical results validating these claims and illustrate their superior performance relative to baselines in experiments.

1 Introduction

Resource allocation is a fundamental problem in economics and computer science that studies the distribution of limited resources among requesting agents. We consider sequential (or dynamic, online) resource allocation, in which a supplier needs to distribute limited resources to a large number of agents demanding the resources *without* charging monetary payments. The interaction between the supplier and the agents occurs over multiple rounds within a finite horizon. In each round, a subset of the agents send requests for one or multiple types of the resources. Based on the demands in the current round (and the demands and allocations made previously) but not observing the future demands, the supplier needs to make an irrevocable allocation, with the goal of optimizing aggregate performance metrics. Applications of the problem framework span a wide range of domains, including telecommunication (Su et al., 2019; Guo et al., 2022), cloud computing (Vinothina et al., 2012; Belgacem, 2022), public health (Cao & Huang, 2012; Ehmann et al., 2021), and poverty relief (Yang, 2018; Gómez-Pantoja et al., 2021).

A significant challenge in sequential resource allocation stems from the uncertainty of the realized future requests, even when knowledge of their distribution is available. Successful mechanisms need to balance between consuming the resources as requests arrive and saving resources for anticipated future requests. The existing literature handles the uncertainty leveraging techniques such as confidence bounds (Sinclair et al., 2020; 2022; Hassanzadeh et al., 2023) and dynamic programming (Powell & Topaloglu, 2006; Forootani et al., 2020), and focuses on designing payment-free resource allocation mechanisms to optimize/achieve the following objectives: 1) *Nash social welfare* (NSW), defined as the product of all agents' utilities, 2) *efficiency*, measuring the utilization rate of resources, 3) *competitive ratio*, measuring the agents' utilities compared against those from some optimal mechanism with hindsight knowledge, 4) *envy-freeness*, where each agent prefers its own allocation over the allocation of any other.

A critical assumption made in these works is that the agents report their requests truthfully. Mechanisms designed under this assumption are highly exploitable when it does not hold, allowing a strategic agent to substantially increase its allocation by sending untruthful requests. In real-life applications, the agents are usually self-interested humans and/or entities that are unlikely to be always truthful, which is rarely prioritized in academic literature. In this work, our goal is to bridge this gap by designing mechanisms that (approximately) achieve both NSW and incentive compatibility (IC) in the sequential setting. IC is a property of a resource allocation mechanism which guarantees that no agent can obtain a strictly more preferable allocation by misreporting requests, and is formed as the unilateral deviation in their utility from its rational optimal, a quantity referred to as exploitability.

To the best of our knowledge, IC has not been considered in prior art on payment-free sequential resource allocation. Even in the one-shot allocation setting (horizon length equals one), ensuring incentive compatibility necessarily leads to unfair mechanisms (in terms of NSW) (Hartline & Roughgarden, 2008), and balancing between NSW and exploitability in the sequential setting raises intrinsic questions regarding scalability with respect to the problem horizon, which are identified in this work for the first time. Our approach to this problem class is to design a general meta-algorithm for assembling a one-shot allocation mechanism into its sequential version, which ensures that the desirable properties of the one-shot mechanism – NSW and IC – are inherited by their sequential extension. This allows us to avoid designing a mechanism from scratch for the sequential setting, while exploiting advances in the (better-studied) one-shot resource allocation literature.

Main Contributions

- We propose a meta-algorithm, named **S**equential **A**llocation **M**eta-**A**lgorithm (`SAMA`), which can be regarded as a general framework for reducing a sequential resource allocation problem into a series of one-shot problems. The key feature of `SAMA` is that it accounts for past allocation and unobserved future requests – agents with greater past allocations are more heavily discounted against in the current round, and resources are withheld for future requests based on a confidence bound.

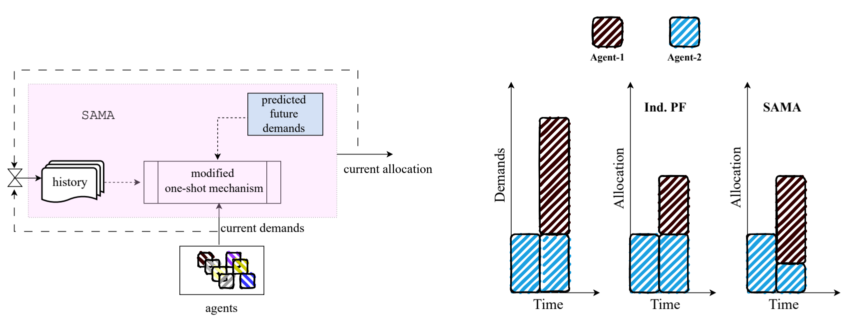

Figure 1: `SAMA` Algorithm & Performance Comparison (see Example 2 for discussion).

- We establish theoretically that if a mechanism optimizes NSW and/or achieves IC in the one-shot sense, `SAMA` approximately carries over the properties to the sequential setting. This implies that with a suitable one-shot mechanisms plugged in as the building block, `SAMA` enjoys approximate NSW and IC guarantees at the same time. To our knowledge, this is the first time such a result has been established for a sequential mechanism in the payment-free setting.

- We further illustrate the superior performance of `SAMA` and its approximate NSW and IC preserving properties using experiments on synthetic data. We plug in the following well-known one-shot mechanisms for validation: 1) the Proportional Fairness (PF) mechanism, which achieves the maximum possible NSW but severely violates IC, 2) the Partial Allocation (PA) mechanism, designed by Cole et al. (2013) to be exactly IC at the cost of a substantial reduction to NSW, 3) ExS-Net, which is a learned neural-network-parameterized mechanism proposed in Zeng et al. (2024b) that achieves near-optimal NSW and approximate IC simultaneously.

1.1 Related Work

We note two (inter-connected) lines of approaches to sequential resource allocation in the literature. The first formulates the problem in a general online decision making framework with (possibly non-convex) reward and resource consumption functions (Mirrokni et al., 2012; Balseiro et al., 2020; 2021b; 2023; An et al., 2024), which may model various payment-based and payment-free problems with proper choices of the reward function. Most works in this direction consider stochastic (i.i.d.) and/or adversarial request models, and some do not require distributional knowledge of the future requests. The algorithm performance is measured by a regret/competitive ratio defined with respect to the optimal allocation in hindsight. The algorithms developed for such general frameworks often have a strong connection to bandit algorithms (Zhalechian et al., 2022; Molina et al., 2023). The latest representative works (Balseiro et al., 2023; An et al., 2024) take a primal-dual approach where the dual variable is associated with the budget constraints, and they establish strong performance guarantees in terms of regret/competitive ratio which matches the worst-case lower bounds. The works so far are restricted to the setting where the agents report requests truthfully.

The second line of work approaches specific resource allocation problems with grounded formulation (Walsh, 2011; Sinclair et al., 2020; 2022; Liao et al., 2022; Hassanzadeh et al., 2023; Yang et al., 2024). Typically the existing work assumes a linear additive agent utility function, and the optimization objectives include social welfare (fairness), efficiency, and/or envy-freeness. Envy-freeness and certain notions of social welfare may not be conveniently modeled by the reward function considered in the general frameworks. Therefore, tailored analyses are usually carried out. It is worth noting again that incentive compatibility has not been considered in this line of work.

Finally, we point out that incentive compatible mechanism design has been studied in the payment-based (auction) setting (Tan et al., 2020; Deng et al., 2021; Balseiro et al., 2021a), where the monetary exchange acts an important tool for eliminating the incentive to misreport. This tool is unavailable in the payment-free case.

2 Problem Formulation – Sequential Resource Allocation

We consider the sequential resource allocation problem, which is a generalization of the single-period resource allocation problem with stochastic requests arriving over time. A supplier needs to allocate a finite number M of divisible resources to N agents over a horizon of T discrete time intervals. Any agent may come to the supplier in any number of intervals and submit a request for one or multiple types of the resources every time. We use $x_{i,m}^{[t]} \in [0, \bar{x}]$ (for some $\bar{x} < \infty$) to denote the quantity of resource $m \in [M]$ requested by agent $i \in [N]$ in time interval $t \in [T]$[1]. We assume a *clipped* linear utility – each unit of resource m increases the utility of agent i by $v_{i,m}$ up to the demand, with $v_{i,m} \in [\underline{v}, \bar{v}]$, $\forall i, m$ for some $0 < \underline{v}, \bar{v} < \infty$. This is a standard assumption in the literature (Cole et al., 2013) – see (1). Both $x_{i,m}^t$ and $v_{i,m}$ are privately known only to agent i.

An agent submits a request by reporting these values to the supplier (possibly untruthfully). The valuation is only reported the first time an agent submits a request and fixed for the entire horizon. This is a reasonable assumption that captures the real-world static preferences for resources, with only changing demand over time. Observing all requests in time interval t, the supplier makes an irrevocable allocation $a_{i,m}^{[t]} \geq 0$ to every agent i for every resource m. The supplier may take historical information into account when making a decision, including the total past allocation denoted by $\widetilde{a}$,

[1]We use $[M]$ to represent $\{1, 2, \cdots, M\}$.

where $\widetilde{a}_{i,m}^{[t]}$ represents the total allocation of resource m made to agent i until time t, i.e.

$$\widetilde{a}_{i,m}^{[1]} \triangleq 0, \quad \widetilde{a}_{i,m}^{[t]} \triangleq \sum_{t'=1}^{t-1} a_{i,m}^{[t']}, \forall t \geq 2.$$

The budget $B_m \geq 0$ is the total available quantity of resource m, known to the supplier before allocation begins and not re-stocked. We denote by $b_m^{[t]} \in \mathbb{R}_+$ the remaining quantity of resource m at the beginning of interval t, which satisfies the relation

$$b_m^{[1]} = B_m, \quad b_m^{[t]} = b_m^{[t-1]} - \sum_{i=1}^{N} a_{i,m}^{[t-1]} = B_m - \sum_{t'=1}^{t-1} \sum_{i=1}^{N} a_{i,m}^{[t']}, \forall t \geq 2.$$

We may aggregate valuations, demands, and budgets across agents, resources, and/or intervals. For a list of the notations, see Table 3. In particular, we use the bold notation $\boldsymbol{x}, \boldsymbol{a}$ to denote the aggregation of demands and allocations over time. The valuations v, demands $\boldsymbol{x}$, budgets B are random variables following a known joint distribution. Let $I^{[t]}$ represent the historical information observed by the supplier up to the beginning of time interval t, i.e. $I^{[1]} = \{b^{[t]}\}$ and $I^{[t]} \triangleq \{v, x^{[1]}, \cdots, x^{[t-1]}, a^{[1]}, \cdots, a^{[t-1]}, b^{[1]}, \cdots, b^{[t]}\}$ for $t \geq 2$. We denote by $\mathcal{I}^{[t]}$ the space of historical information at time t. For simplicity, we assume that the demands of time t are not affected by allocations made prior to t, a common setting considered in a number of existing works (Sinclair et al., 2022; Liao et al., 2022; Hassanzadeh et al., 2023). Given demands $\boldsymbol{x} \in \mathbb{R}^{TNM}$, valuations $v \in \mathbb{R}^{NM}$, and allocations $\boldsymbol{a} \in \mathbb{R}^{TNM}$, we use $\boldsymbol{u}_i$ to represent the utility of agent i from its total allocation over the horizon

$$\boldsymbol{u}_i(\boldsymbol{a}, v, \boldsymbol{x}) \triangleq \sum_{t=1}^{T} u_i(a^{[t]}, v, x^{[t]}), \tag{1}$$

where u_i is the single-interval utility function defined as $u_i(a, v, x) \triangleq \sum_{m=1}^{M} v_{i,m} \min\{a_{i,m}, x_{i,m}\}$. To allow for the degree of freedom in discounting certain agents, we introduce a bias matrix $\tilde{a} \in \mathbb{R}^{NM}$, where $\tilde{a}_i$ models the total allocation made to agent i in the past interactions. A sequential mechanism is a policy that determines a valid allocation in each time interval based on the current demands and historical information. A valid allocation must satisfy the budget constraint across time and be no more than the demand.

Definition 1 (Sequential Allocation Mechanism) *A mapping $\boldsymbol{f} = \{f^{[t]} : \mathbb{R}_+^{NM} \times \mathbb{R}_+^{NM} \times \mathcal{I}^{[t]} \to \mathbb{R}_+^{NM}\}_{t \in [T]}$ is said to be a sequential allocation mechanism if for all $t, v, x^{[t]}, I^{[t]}$*

$$\sum_{i=1}^{N} f_{i,m}^{[t]}(v, x^{[t]}, I^{[t]}) \leq b_m^{[t]}, \ \forall m; \quad 0 \leq f_{i,m}^{[t]}(v, x^{[t]}, I^{[t]}) \leq x_{i,m}^{[t]}, \ \forall i, m. \tag{2}$$

We denote $\boldsymbol{f}(v, \boldsymbol{x}, B) = [f^{[1]}(v, x^{[1]}, I^{[1]}); \cdots; f^{[T]}(v, x^{[T]}, I^{[T]})] \in \mathbb{R}_+^{TNM}$.

2.1 Mechanism Design Objectives

We study designing sequential mechanisms that balance NSW and exploitability. The NSW in the sequential setting can be defined by following the classic one-period definition (Cole et al., 2013).

Definition 2 (Sequential NSW) *Given $v \in \mathbb{R}_+^{NM}$, $\boldsymbol{x} \in \mathbb{R}_+^{TNM}$, $B \in \mathbb{R}_+^{M}$, the Nash social welfare of a sequential mechanism $\boldsymbol{f}$ is defined as*

$$\mathbf{NSW}(\boldsymbol{f}, v, \boldsymbol{x}, B) \triangleq \prod_{i=1}^{N} \boldsymbol{u}_i(\boldsymbol{a}, v, \boldsymbol{x}), \quad \textit{where } \boldsymbol{a} = \boldsymbol{f}(v, \boldsymbol{x}, B).$$

The definition states that the agents evaluate their satisfaction based on the total allocation they receive over the horizon, on which an aggregate NSW is computed. A mechanism that maximizes this NSW aims to ensure a "fair" cumulative allocation over time for all agents. We believe this is one such definition that matches the objective usually applicable in real-world problems where the performance is evaluated based on cumulative outcomes, such as in computational resource allocation and wireless networks.

Definition 3 (Exploitability) *For mechanism $\boldsymbol{f}$ and $v \in \mathbb{R}_+^{NM}$, $\boldsymbol{x} \in \mathbb{R}_+^{TNM}$, and $B \in \mathbb{R}_+^{M}$, we define*

$$\mathbf{expl}_i^{online}(\boldsymbol{f}, v, \boldsymbol{x}, B) \triangleq \max_{t, v_i' \in \mathbb{R}_+^M, x_i' \in \mathbb{R}_+^M} u_i\left(f^{[t]}\left((v_i', v_{-i}), (x_i', x_{-i}^{[t]}), I^{[t]}\right), v, x^{[t]}\right) - u_i\left(f^{[t]}\left(v, x^{[t]}, I^{[t]}\right), v, x^{[t]}\right),$$

$$\mathbf{expl}_i^{full}(\boldsymbol{f}, v, \boldsymbol{x}, B) \triangleq \max_{v_i' \in \mathbb{R}_+^M, \boldsymbol{x}_i' \in \mathbb{R}_+^{TM}} \boldsymbol{u}_i\left(\boldsymbol{f}\left((v_i', v_{-i}), (\boldsymbol{x}_i', \boldsymbol{x}_{-i}), B\right), v, \boldsymbol{x}\right) - \boldsymbol{u}_i\left(\boldsymbol{f}(v, \boldsymbol{x}, B), v, \boldsymbol{x}\right),$$

where $I^{[t]}$ is generated under $\boldsymbol{f}$.

Conceptually, the online exploitability measures the maximum possible utility increase obtained by an agent in any interval t when it misreports its parameters *only in interval* t. The full exploitability is a more ambitious metric – it measures the maximum total utility increase of agent i when it misreports its parameters *across all intervals*. Note that $\mathbf{expl}_i^{\text{full}}$ may be far larger than $T \cdot \mathbf{expl}_i^{\text{online}}$. A small $\mathbf{expl}_i^{\text{full}}$ necessarily implies a small $\mathbf{expl}_i^{\text{online}}$, but the converse is not true (see Example 1 below). We say that a sequential mechanism $\boldsymbol{f}$ is ϵ-online/full incentive compatible if $\mathbf{expl}_i^{\text{online}}(\boldsymbol{f}, v, \boldsymbol{x}, B) \le \epsilon$ or $\mathbf{expl}_i^{\text{full}}(\boldsymbol{f}, v, \boldsymbol{x}, B) \le \epsilon$ for all $i, v, \boldsymbol{x}, B$.

2.2 One-shot Allocation ($T = 1$)

We quickly discuss the special case when $T = 1$, as these will feature in the key allocation component of Algorithm 1. The definition of a one-shot allocation mechanism is given as follows.

Definition 4 (One-Shot Allocation Mechanism) *A mapping $f : \mathbb{R}_+^{NM} \times \mathbb{R}_+^{NM} \times \mathbb{R}_+^{M} \times \mathbb{R}^{NM} \to \mathbb{R}_+^{NM}$ is said to be a one-shot mechanism if for all $v \in \mathbb{R}_+^{NM}, x \in \mathbb{R}_+^{NM}, B \in \mathbb{R}_+^{M}$, and $\tilde{a} \in \mathbb{R}^{NM}$*

$$\begin{aligned} &\textstyle\sum_{i=1}^N f_{i,m}(v, x, B, \tilde{a}) \le B_m, \quad \forall m, \\ &0 \le f_{i,m}(v, x, B, \tilde{a}) \le x_{i,m}, \forall i, m. \end{aligned}$$

One-shot allocation mechanism design is well-studied in the literature with standard mechanisms like (i) ***proportional fairness*** (PF): By definition f^{PF} achieves the maximum possible NSW, but is shown to incur a substantial exploitability (Zeng et al., 2024a).

$$\begin{aligned} f^{PF}(v, x, B, \tilde{a}) = \operatorname{argmax}_{a \in \mathbb{R}^{NM}} &\textstyle\sum_{i=1}^N \log u_i(a + \tilde{a}, v, x + \tilde{a}) \\ \text{s.t. } \ 0 \le a \le x; \quad &\textstyle\sum_{i=1}^N a_{i,m} \le B_m, \ \forall m \in [M]. \end{aligned} \tag{3}$$

(ii) ***partial allocation*** (PA): Motivated to design an "unexploitable" mechanism with guarantees on NSW, Cole et al. (2013) proposes the Partial Allocation (PA) mechanism, which is built upon the PF mechanism. PA mechanism assigns to each agent the allocation they would receive under the PF mechanism scaled by a discount ratio (between 0 and 1), computed according to the externality each agent introduces to the system. We represent the PA mechanism by f^{PA} and note that the aforementioned discount ratio is guaranteed to be at least 1/e in the worst case when $\tilde{a} = 0$, i.e., we have for any v, x, B

$$\frac{f_{i,m}^{PA}(v, x, B, 0)}{f_{i,m}^{PF}(v, x, B, 0)} \ge 1/e. \tag{4}$$

However, defined as the product of agents' utilities, the NSW of the PA mechanism deteriorates exponentially with N and is numerically shown in Zeng et al. (2024b) to be negligibly low (less than $1/1000$ of that of the PF mechanism) in 10-agent systems.

(iii) ***ExS-Net***: Balancing between the two ends of the spectrum, Zeng et al. (2024b) introduces a neural-network-parameterized mechanism ExS-Net. Trained with samples from a distribution of truthful parameters, the mechanism ensures that no agent can benefit from untruthful reporting by more than a user-specified parameter $\epsilon > 0$. With a suitable choice of ϵ, ExS-Net substantially reduces the exploitability relative to the PF mechanism, while still achieving near-optimal NSW. We denote the mechanism as f^{ExS} in the rest of paper.

Example 1 *We discuss a simple mechanism which incurs zero online exploitability but a non-zero full exploitability. Suppose that $T = 2$ and we run the mechanism $\boldsymbol{f} = \{f^{[1]}, f^{[2]}\}$ defined as follows*

$$a_{i,m}^{[1]} = f_{i,m}^{[1]}(v, x^{[1]}, I^{[1]}) = f_{i,m}^{PA}(v, x^{[1]}, \frac{1}{2}B, 0),$$
$$f_{i,m}^{[2]}(v, x^{[2]}, I^{[2]}) = \begin{cases} f_{i,m}^{PA}(v, x^{[2]}, \frac{1}{2}B, 0), & \text{if } a_{i,m}^{[1]} \leq \frac{1}{4}x_{i,m}^{[1]}, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Eq. (5) says that we allocate according to the PA mechanism in the first interval, with half of the total available budget. In the second interval, we do not allocate anything unless the allocation made in the first interval is much smaller than what the agent requests – in that case, we allocate to the specific agent on the specific resource according to the PA mechanism. This is a valid sequential mechanism, as the budget constraint is never violated. It can also be seen that the online exploitability of the mechanism is zero, as f^{PA} satisfies IC. However, the full exploitability is non-zero, as an agent supposed to receive zero allocation in the second interval with a truthful report can suitably under-report its demand in the first interval to increase its second-round allocation.

Why is sequential setting with IC & NSW non-trivial? First, consider the one-shot setting. NSW as an objective can be optimized by considering the allocation that solves (3). IC on the other hand, can only be evaluated and optimized *given a mechanism*. This is what makes it challenging to address both these simultaneously, and the literature on hand-designed mechanisms with exact guarantees solve either NSW (ex. PF) or IC (ex. PA). The sequential version only exacerbates this challenge. Ensuring full IC over multiple rounds increases the difficulty, as it is unclear how to prevent agents from manipulating future allocations by adjusting their current reports. We take the first step in tackling this problem by instead designing mechanisms that approximately preserve the properties of the well-understood one-shot mechanisms. In doing so, however, we reveal potentially unimprovable dependence on the problem horizon, unless additional structure is assumed regarding the interaction between demand and time, which we defer to future work.

3 Meta-Algorithm for Sequential Allocation

In this section, we introduce the Sequential Allocation Meta-Algorithm (`SAMA`), a framework for applying one-shot mechanisms to the sequential setting. The key challenge of sequential resource allocation lies in the future request uncertainty. `SAMA` is designed to account for the worst case in the face of uncertainty by following the simple idea of pre-allocating to future requests pretending that they will arrive exactly as their lower confidence bounds. A similar idea has been considered in Hassanzadeh et al. (2023) in the design of their SAFFE algorithm. Interestingly, `SAMA` with the PF mechanism plugged in can be regarded as a generalization of SAFFE to the multi-resource setting.

We denote the one-shot/ single-period allocation mechanism as $f^{\text{one-shot}}$, which takes arguments v (valuation), x (demand), B (budget), and $\tilde{a}$ (past allocation) and produces an allocation outcome a. In the rest of the paper, we use the notation $\text{SAMA}(f^{\text{one-shot}})$ to represent the sequential mechanism built from $f^{\text{one-shot}}$ according to Algorithm 1. Formally presented in Algorithm 1 and illustrated in Figure 1, `SAMA` initializes the budget $b^{[1]} = B \in \mathbb{R}^M$ and total past allocation $\tilde{a}^{[1]} = \mathbf{0} \in \mathbb{R}^{MN}$, and operates in every interval t as follows. First, `SAMA` determines an "allocation factor" $\beta_i^{[t]}$ as the ratio between the expected demand in the current interval and the total expected demand from t till the end of the horizon. This factor is used to scale the current demand to produce $y_{i,m}^{[t]}$ as an estimate of the total demands for the remaining intervals. For simplicity of presentation, we assume that the expected demand $\mathbb{E}[x_{i,m}^{[t]}]$ is always positive, which makes the allocation factor well-defined, but we note that the generalization can be easily made by fixing $\beta_{i,m}^{[t]}$ to 1 when the denominator of (6) is zero. Second, we apply the one-shot mechanism to calculate an tentative allocation $c^{[t]} \in \mathbb{R}^{MN}$ based on $y^{[t]}$ using the full remaining budget. However, we cannot allocate $c^{[t]}$ as it contains a portion associated with future requests. We determine the actual allocation by scaling $c^{[t]}$ back with the allocation factor $\beta^{[t]}$, update the remaining budget and past allocation information, and proceed to the next iteration. As the allocation in every iteration only uses the remaining budget and the allocation factor $\beta_i^{[t]}$ always lies between 0 and 1, `SAMA` always returns a feasible allocation for every t.

Remark 1 *When the exact future request distribution is unknown,* `SAMA` *can be applied using expectations and standard deviations estimated from data. If no such data is available, we can use* `SAMA` *with* $\beta_i^{[t]} = 1$*, ensuring that at least past allocations are considered when making current decisions. Although constantly setting* $\beta_i^{[t]} = 1$ *results in a loss of the mathematical guarantees on NSW, the approach remains preferable to independently applying one-shot mechanism in each iteration, as it still accounts for past allocations.*

Algorithm 1 Sequential Allocation Meta-Algorithm (`SAMA`)

```
Initialize: budget b^[1] = B ∈ R^M, past allocations ã^[1] = 0 ∈ R^{MN}
for interval t = 1, ..., T do
    Receive reported valuation v_i for all i (only the first time that agent i reports) and demand x^[t]
    Calculate β^[t], y^[t] ∈ R^N such that
```

$$\beta_{i,m}^{[t]} = \frac{\mathbb{E}[x_{i,m}^{[t]}]}{\mathbb{E}[x_{i,m}^{[t]}] + \sum_{\tau>t} \max\left\{\mathbb{E}[x_{i,m}^{[\tau]}] - \lambda^{[\tau]}\,\mathrm{std}(x_{i,m}^{[\tau]}), 0\right\}},$$
$$\beta_i^{[t]} = \tfrac{1}{M}\sum_{m=1}^{M} \beta_{i,m}^{[t]}, \quad y_{i,m}^{[t]} = x_{i,m}^{[t]}/\beta_i^{[t]}. \tag{6}$$

```
    Apply one-shot allocation mechanism with bias and allocate a_i^[t] to agent i
```

$$c^{[t]} = f^{\text{one-shot}}(v, y^{[t]}, b^{[t]}, \tilde{a}^{[t]}),$$
$$a_{i,m}^{[t]} = \min\{\beta_i^{[t]} c_{i,m}^{[t]}, x_{i,m}^{[t]}\}. \tag{7}$$

```
    Update remaining budget b_m^[t+1] = b_m^[t] - Σ_{i=1}^N a_{i,m}^[t] for all m ∈ [M]
    Update past allocation
```

$$\tilde{a}_{i,m}^{[t+1]} = \tilde{a}_{i,m}^{[t]} + a_{i,m}^{[t]}, \quad \forall i, m$$

```
end for
```

Example 2 *Consider the following simple case of* 2 *agents requesting a single resource over* $T = 2$ *time periods. Let the total budget* $B = 6$ *units and the demands be as follows:* $a_1^{[1]} = 2, a_2^{[1]} = 0$ *and* $a_1^{[2]} = 2$, $a_2^{[2]} = 4$ *units over the two time periods; as illustrated in the first sub-figure in Fig. 1. Suppose we are interested in maximizing the NSW over the two time periods. We know that PF allocation achieves the largest welfare in a single time-period (Cole et al., 2013; Zeng et al., 2024b). We known that for a single source allocation problem, PF can be seen as a water-filling solution (Hassanzadeh et al., 2023). If we myopically solve for PF allocations in each period, we obtain the allocation in the second sub-figure in Fig. 1. Intuitively, first period allocation of* 2 *units goes to Agent-2. In the second period, with a remaining budget of* 4 *units, following a water-filling strategy each of the agents get* 2 *units. On the other hand, if we use,* `SAMA`*, which accounts for the past and future allocations we obtain the third sub-figure in Fig. 1. The first period allocation proceeds as it is. In the second period,* `SAMA` *employs a bias-adjusted water-filling strategy where Agent-2 having already received* 2 *units can receive at most* 1 *unit, while the remaining* 3 *units goes to Agent-1. Comparing the two allocations, we see that independent PF has* 4 *units for Agent-2 and* 2 *units for Agent-1, while* `SAMA` *has* 3 *units for each overall achieving a higher welfare.*

4 Theoretical Guarantees

The important feature of SAMA is that it achieves approximate IC and NSW maximization, provided that the one-shot mechanism from which it is built upon enjoys such properties. In this section, we establish a few bounds for SAMA on 1) the online incentive compatibility, 2) the full incentive compatibility under a "correction" condition, and 3) the optimality gap (regret) in NSW compared against the NSW maximization allocation in hindsight.

Theorem 1 (Online Incentive Compatibility) *Suppose that the one-shot mechanism is ϵ-incentive compatible, i.e. it satisfies for all agent i*

$$\mathbf{expl}_i^{\textit{one-shot}}(f^{\textit{one-shot}}, v, x, B, \tilde{a}) \leq \epsilon, \quad \forall v, x, B, \tilde{a}. \tag{8}$$

Then, we have for any valuation and demand and budget profile $v, \boldsymbol{x}, B$ and agent i

$$\mathbf{expl}_i^{\textit{online}}(\mathrm{SAMA}(f^{\textit{one-shot}}), v, \boldsymbol{x}, B) \leq \epsilon.$$

The first theorem states that if the one-shot mechanism is ϵ-incentive compatible, SAMA is guaranteed to build a sequential mechanism that is ϵ-online incentive compatible in the sense of Definition 3. We defer the all proofs to the supplementary material, but point that Theorem 1 follows from a simple argument – SAMA straightforwardly inherits the online IC property from the one-shot mechanism as it applies a scaled version of the one-shot mechanism in each interval.

Theorem 2 (Full Incentive Compatibility) *Suppose that the one-shot is ϵ-incentive compatible in the sense of* (8) *and satisfies the correction condition. Then, we have for any valuation and demand profile $v, \boldsymbol{x}$, budget B, and agent i*

$$\mathbf{expl}_i^{\textit{full}}(\boldsymbol{f}, v, \boldsymbol{x}, B) \leq T\epsilon.$$

This result importantly says that if our aim is to design a sequential mechanism with Δ full exploitability and the horizon is T, we simply need to enforce that the $f^{\text{one-shot}}$ is $\frac{\Delta}{T}$-IC. We make use of the following "correction" property of $f^{\text{one-shot}}$ to rule out the possibility of the worst case and show that the full exploitability of SAMA is only linear in T. Given $\tilde{a}, \tilde{a}' \in \mathbb{R}^M$, suppose the one-shot mechanism satisfies for all i, v, x, B

$$\begin{aligned}|u_i(f^{\text{one-shot}}(v, x, B - \sum_i \tilde{a}_i, \tilde{a}) + \tilde{a}, v, x + \tilde{a}) - u_i(f^{\text{one-shot}}(v, x, B - \sum_i \tilde{a}'_i, \tilde{a}') + \tilde{a}', v, x + \tilde{a}')| \\ \leq |u_i(\tilde{a}, v, \tilde{a}) - u_i(\tilde{a}', v, \tilde{a}')|.\end{aligned} \tag{9}$$

We argue that the correction property is a mild condition, which conceptually says the following. Consider the same agent in two scenarios. In scenario 1, the agent is over-allocated in the past and has a high utility resulting from the past allocation. In scenario 2, the agent is less allocated and has a lower utility. After a new round of allocation is made by the one-shot mechanism (accounting for the past allocation), the difference in the utilities between the two scenario should be "corrected" and not become larger. Note that establishing this bound requires more than simply applying the online IC bound across time. As we have seen in Example 1, it can happen that an exactly online-IC sequential mechanism has a non-zero full exploitability. Even with SAMA, there is the possibility in the worst case that the full exploitability scales exponentially with respect to T, as an earlier misreport can have a long-lasting and recurring effect on later allocations (since the allocation mechanism needs to account for the past allocation).

Theorem 3 (Nash Social Welfare) *Suppose that the one-shot mechanism $f^{\textit{one-shot}}$ satisfies the correction property in* (9) *and is δ-NSW optimal in the sense that the difference between the allocation under $f^{\textit{one-shot}}$ and that under the PF mechanism f^{PF} is at most δ, i.e. for any i*

$$\|f_i^{\textit{one-shot}}(v, x, B, a) - f_i^{PF}(v, x, B, a)\| \leq \delta. \tag{10}$$

Let $\mathrm{std}^{\max} = \max_{i,m,t} \mathrm{std}(x_{i,m}^{[t]})$*. Given a target failure probability* $\xi > 0$*, let* $\lambda^{[\tau]} = \sqrt{(T-\tau)/\xi}$ *in* (6)*. With the number of resources* $M = 1$*, it holds with probability at least* $1 - \xi$

$$\mathbf{regret}^{\mathbf{NSW}}(\mathrm{SAMA}(f^{\textit{one-shot}}) \leq \frac{2T^{3/2}N\bar{v}}{\sqrt{\xi}}\,\mathrm{std}^{\max} + T\bar{v}\delta,$$

where $\mathbf{regret}^{\mathbf{NSW}}(\boldsymbol{f}) = \mathbb{E}_{v,\boldsymbol{x},B}[\mathbf{NSW}^{\textit{one-shot}}(f^{PF}, v, \sum_{t=1}^{T} x^{[t]}, B, 0) - \mathbf{NSW}(\boldsymbol{f}, v, \boldsymbol{x}, B)]$*. With* $\mathbf{NSW}^{\textit{one-shot}}$ *defined in* (12) *in the supplementary material, the first term of the regret expresses the maximum possible NSW that can be achieved in hindsight.*

This theorem establishes a bound on the optimality gap (regret) in NSW, in the special case of a single resource. We define regret by comparing against the maximum achievable NSW with the complete and truthful observation of $v, \boldsymbol{x}$, attainable by the PF mechanism with hindsight knowledge – we simply need to apply the PF mechanism on the demands aggregated over time. Similar to full exploitability, we note that in the worst case the sequential NSW may scale exponentially with T, which we rule out by leveraging the correction property. The bound states that a NSW maximizing one-shot mechanism can be used to built an approximate NSW optimal sequential one, up to a gap scaling with the standard deviation of the demand distribution.

Mechanism	NSW	Efficiency (%)	Full Exploitability
SAMA(PF)	2.28±1.19	95.41±6.27	5.47e-2±2.78e-2
SAMA(PA)	1.00±0.77	54.39±13.64	0.0±0.0
SAMA(ExS-Net)	2.14±1.14	95.33±6.28	2.55e-2±1.78e-2
Independent(PF)	1.96±1.03	90.96±8.72	6.12e-2±4.84e-2
Independent(PA)	8.20e-1±6.63e-1	49.83±14.02	0.0±0.0
Independent(ExS-Net)	1.89±9.95e-1	90.66±8.71	3.25e-2±1.96e-2

Table 1: Mechanism performance in 2x2 system.

Mechanism	NSW	Efficiency (%)	Full Exploitability
SAMA(PF)	1.74e+4±1.67e+4	100.0±0.0	1.62e-1±4.00e-2
SAMA(PA)	8.63±9.08	39.52±4.58	0.0±0.0
SAMA(ExS-Net)	2.71e+3±2.03e+3	99.89±0.16	2.83e-3±1.16e-3
Independent(PF)	9.15e+3±9.05e+3	99.72±0.67	1.61e-1±3.02e-2
Independent(PA)	3.73±3.89	36.84±5.20	0.0±0.0
Independent(ExS-Net)	2.29e+3±1.71e+3	98.48±1.07	3.79e-3±1.27e-3

Table 2: Mechanism performance in 10x3 system.

5 Numerical Simulations

The purpose of this section is to provide insight into the performance of SAMA through a range of simulations. Specifically, we examine 1) how SAMA performs relative to the baseline sequential mechanism built by applying one-shot mechanisms independently in each interval until the budget runs out, 2) the behavior of SAMA as the budget level and horizon length vary. Given $f^{\text{one-shot}}$, this baseline sequential mechanism, which we denote as $\mathrm{Independent}(f^{\text{one-shot}})$, operates as follows. In each interval t, the supplier observes $v, x^{[t]}$, allocates $a^{[t]} = f^{\text{one-shot}}(v, x^{[t]}, b^{[t]}, 0)$, and updates the budget $b^{[t+1]} = b^{[t]} - \sum_{i=1}^{N} a_i^{[t]}$ with $b^{[1]} = B$.

Data Generation. In all experiments, we consider valuations and demands that element-wise follow the uniform and Bernoulli-uniform distributions within the range $[0.1, 1]$. Specifically, for all i, m, t

$$v_{i,m} \sim \text{Unif}(0.1, 1),\ \breve{x}_{i,m}^{[t]} \sim \text{Unif}(0.1, 1), \quad \hat{x}_{i,m}^{[t]} \sim \text{Bern}(0.5), \quad x_{i,m}^{[t]} = \breve{x}_{i,m}^{[t]}\hat{x}_{i,m}^{[t]}.$$

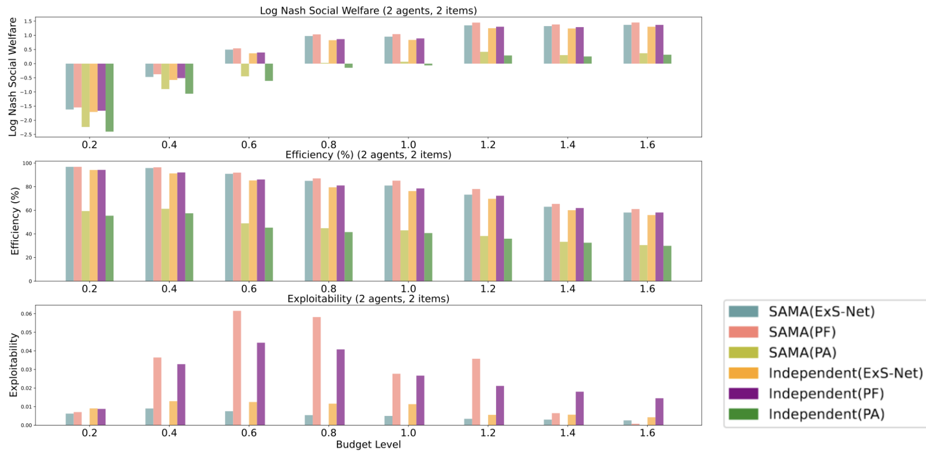

Figure 2: Algorithm Performance in 2x2 System under Varying Budget.

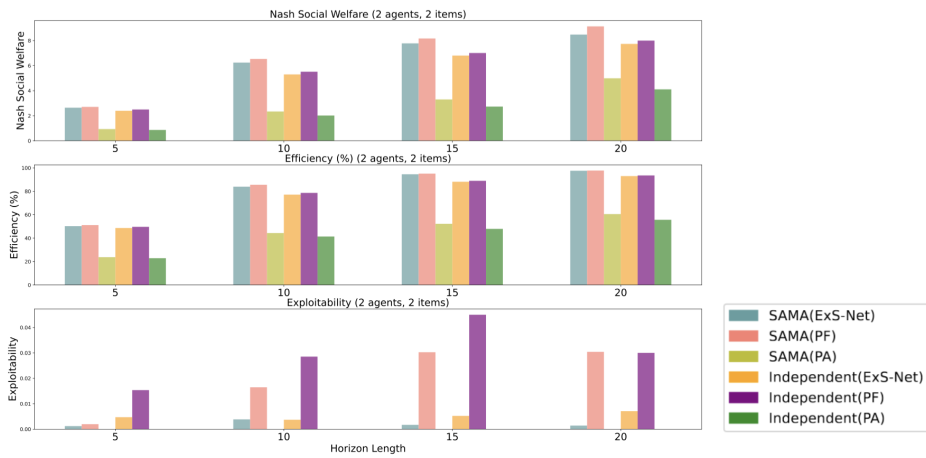

Figure 3: Algorithm Performance in 2x2 System with Varying Horizon.

Unless otherwise noted, we set the budget for each resource to $\frac{NT}{4}$, which means that on average every agent expects to receive an allocation slightly lower than a half of its demand. This budget level creates reasonable competition for the resources.

Metrics. Our evaluation metrics include NSW and full exploitability introduced in Section 2, as well as efficiency. Given $v \in \mathbb{R}_+^{TNM}$, $\boldsymbol{x} \in \mathbb{R}_+^{TNM}$, and $B \in \mathbb{R}_+^{M}$, the efficiency of a sequential mechanism $\boldsymbol{f}$ on resource m is

$$\textbf{efficiency}_m(\boldsymbol{f}, v, \boldsymbol{x}, B) \triangleq \frac{1}{B_m} \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,m}^{[t]}(v, x^{[t]}, I^{[t]}),$$

where $I^{[t]}$ is generated under $\boldsymbol{f}$. Mechanisms with high efficiency reduces the waste of resources and are hence preferable. In our simulations, we report the averaged efficiency over resources.

We first present in Tables 1 (2-agent 2-resource system) and 2 (10-agent 3-resource system) the performance of `SAMA` against Independent with PF mechanism, PA mechanism, and properly trained ExS-Net as the one-shot mechanism backbone. Note that the to exactly calculate the full exploitability an optimization program needs to be solved to find the optimal misreported parameters for each agent. We approximate the optimal misreports by a local grid search around the true parameters.

Across meta-algorithms, we see that `SAMA` outperforms Independent across all metrics. Within `SAMA`, it is observed that the properties of the one-shot mechanism are preserved. In the one-shot setting, the PF mechanism achieves the largest NSW, the PA mechanism has zero exploitability, and ExS-Net strikes a balance between them. This relationship remains consistent in the sequential setting.

Varying Budget Level. We also visualize the mechanism performance as a budget scaling α parameter, which leads to the budget $B_m = \frac{\alpha NT}{2}$ for every resource m, varies from 0.2 (scarce) to 1.6 (abundant). The budget for The expected behavior in terms of NSW, efficiency, and exploitability is 1) that NSW should constantly move up as more resources are available, 2) that the efficiency drops as the chance of the budget exceeding the total demand increases, thus creating a waste, 3) that the exploitability exhibit an increase-then-decrease movement, as misreporting helps little under a small budget and is unnecessary when the resources are excessive. The simulation results for the 2-agent 2-resource problem, plotted in Figure 2, match the expectation and show that `SAMA` again consistently achieves better metrics than Independent. We note that experimental results on the 10-agent 3-resource problem can be found in Section 9 of the supplementary material.

Varying Horizon. We also investigate the effect of varying horizon on the mechanism performance. Shown in Figure 3 for the 2-agent 2-resource problem, NSW increases as T goes up as the overall budget increases with T, while the full exploitability also increases, matching the behavior predicted by the bound in Theorem 2. The trend is consistently observed in the 10-agent 3-resource problem as well, and we defer the plot to Section 9 of the supplementary material.

6 Conclusion & Future Work

There is a gap in the literature on sequential mechanisms that can (approximately) optimize both IC and NSW without monetary payments. We proposed a simple method that builds sequential mechanisms from one-shot mechanisms approximately preserving their properties.

A interesting future direction is to learn sequential IC mechanism. In the one-shot setting, Dütting et al. (2024); Ivanov et al. (2022); Zeng et al. (2024b;a) have explored parametrizing the mechanism using neural networks and learning them end-to-end from data. While one sacrifices strong theoretical guarantees associated with the so-obtained mechanisms, this approach achieves favorable empirical trade-offs between the competing objectives. It would be of interest to extend this approach to the sequential problem, which can be formulated as a Markov decision process, leveraging reinforcement learning.

References

Lin An, Andrew A Li, Benjamin Moseley, and Gabriel Visotsky. Best of many in both worlds: Online resource allocation with predictions under unknown arrival model. *arXiv preprint arXiv:2402.13530*, 2024.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Dual mirror descent for online allocation problems. In *International Conference on Machine Learning*, pp. 613–628. PMLR, 2020.

Santiago Balseiro, Yuan Deng, Jieming Mao, Vahab Mirrokni, and Song Zuo. Robust auction design in the auto-bidding world. *Advances in Neural Information Processing Systems*, 34:17777–17788, 2021a.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Regularized online allocation problems: Fairness and beyond. In *International Conference on Machine Learning*, pp. 630–639. PMLR, 2021b.

Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. Online resource allocation under horizon uncertainty. In *Abstract Proceedings of the 2023 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 63–64, 2023.

Ali Belgacem. Dynamic resource allocation in cloud computing: analysis and taxonomies. *Computing*, 104(3):681–710, 2022.

Hui Cao and Simin Huang. Principles of scarce medical resource allocation in natural disaster relief: a simulation approach. *Medical Decision Making*, 32(3):470–476, 2012.

Richard Cole, Vasilis Gkatzelis, and Gagan Goel. Mechanism design for fair division: allocating divisible items without payments. In *Proceedings of the fourteenth ACM conference on Electronic commerce*, pp. 251–268, 2013.

Yuan Deng, Jieming Mao, Vahab Mirrokni, and Song Zuo. Towards efficient auctions in an auto-bidding world. In *Proceedings of the Web Conference 2021*, pp. 3965–3973, 2021.

Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David C Parkes, and Sai Srivatsa Ravindranath. Optimal auctions through deep learning: Advances in differentiable economics. *Journal of the ACM*, 71(1):1–53, 2024.

Michael R Ehmann, Elizabeth K Zink, Amanda B Levin, Jose I Suarez, Harolyn ME Belcher, Elizabeth L Daugherty Biddison, Danielle J Doberman, Karen D'Souza, Derek M Fine, Brian T Garibaldi, et al. Operational recommendations for scarce resource allocation in a public health crisis. *Chest*, 159(3):1076–1083, 2021.

Ali Forootani, Raffaele Iervolino, Massimo Tipaldi, and Joshua Neilson. Approximate dynamic programming for stochastic resource allocation problems. *IEEE/CAA Journal of Automatica Sinica*, 7(4):975–990, 2020.

J Álvaro Gómez-Pantoja, M Angélica Salazar-Aguilar, and José Luis González-Velarde. The food bank resource allocation problem. *Top*, 29:266–286, 2021.

Wanying Guo, Nawab Muhammad Faseeh Qureshi, Isma Farah Siddiqui, and Dong Ryeol Shin. Cooperative communication resource allocation strategies for 5g and beyond networks: A review of architecture, challenges and opportunities. *Journal of King Saud University-Computer and Information Sciences*, 34(10):8054–8078, 2022.

Jason D Hartline and Tim Roughgarden. Optimal mechanism design and money burning. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 75–84, 2008.

Parisa Hassanzadeh, Eleonora Kreacic, Sihan Zeng, Yuchen Xiao, and Sumitra Ganesh. Sequential fair resource allocation under a markov decision process framework. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pp. 673–680, 2023.

Dmitry Ivanov, Iskander Safiulin, Igor Filippov, and Ksenia Balabaeva. Optimal-er auctions through attention. *Advances in Neural Information Processing Systems*, 35:34734–34747, 2022.

Rohit Konda, Rahul Chandan, David Grimsman, and Jason R Marden. Optimal utility design of greedy algorithms in resource allocation games. *IEEE Transactions on Automatic Control*, 2024.

Luofeng Liao, Yuan Gao, and Christian Kroer. Nonstationary dual averaging and online fair allocation. *Advances in Neural Information Processing Systems*, 35:37159–37172, 2022.

Vahab S Mirrokni, Shayan Oveis Gharan, and Morteza Zadimoghaddam. Simultaneous approximations for adversarial and stochastic online budgeted allocation. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pp. 1690–1701. SIAM, 2012.

Mathieu Molina, Nicolas Gast, Patrick Loiseau, and Vianney Perchet. Trading-off price for data quality to achieve fair online allocation. *Advances in Neural Information Processing Systems*, 36:40096–40139, 2023.

Warren B Powell and Huseyin Topaloglu. Approximate dynamic programming for large-scale resource allocation problems. In *Models, Methods, and Applications for Innovative Decision Making*, pp. 123–147. Informs, 2006.

Sean R Sinclair, Gauri Jain, Siddhartha Banerjee, and Christina Lee Yu. Sequential fair allocation of limited resources under stochastic demands. *arXiv preprint arXiv:2011.14382*, 2020.

Sean R Sinclair, Siddhartha Banerjee, and Christina Lee Yu. Sequential fair allocation: Achieving the optimal envy-efficiency tradeoff curve. *ACM SIGMETRICS Performance Evaluation Review*, 50(1):95–96, 2022.

Ruoyu Su, Dengyin Zhang, Ramachandran Venkatesan, Zijun Gong, Cheng Li, Fei Ding, Fan Jiang, and Ziyang Zhu. Resource allocation for network slicing in 5g telecommunication networks: A survey of principles and models. *IEEE Network*, 33(6):172–179, 2019.

Xiaoqi Tan, Bo Sun, Alberto Leon-Garcia, Yuan Wu, and Danny HK Tsang. Mechanism design for online resource allocation: A unified approach. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 4(2):1–46, 2020.

V V Vinothina, R Sridaran, and Padmavathi Ganapathi. A survey on resource allocation strategies in cloud computing. *International Journal of Advanced Computer Science and Applications*, 3(6), 2012.

Toby Walsh. Online cake cutting. In *Algorithmic Decision Theory: Second International Conference, ADT 2011, Piscataway, NJ, USA, October 26-28, 2011. Proceedings 2*, pp. 292–305. Springer, 2011.

Qiaoyu Yang. *Optimal allocation algorithm for sequential resource allocation in the context of food banks operations*. PhD thesis, Cornell University, 2018.

Zongjun Yang, Luofeng Liao, Yuan Gao, and Christian Kroer. Online fair allocation with best-of-many-worlds guarantees. *arXiv preprint arXiv:2408.02403*, 2024.

Sihan Zeng, Sujay Bhatt, Alec Koppel, and Sumitra Ganesh. Regularized proportional fairness mechanism for resource allocation without money. *Transactions on Machine Learning Research*, 2024a. ISSN 2835-8856. URL https://openreview.net/forum?id=K85nJ60lDR.

Sihan Zeng, Sujay Bhatt, Eleonora Kreacic, Parisa Hassanzadeh, Alec Koppel, and Sumitra Ganesh. Learning payment-free resource allocation mechanisms. In *2024 Winter Simulation Conference (WSC)*, pp. 2667–2678. IEEE, 2024b.

Mohammad Zhalechian, Esmaeil Keyvanshokooh, Cong Shi, and Mark P Van Oyen. Online resource allocation with personalized learning. *Operations Research*, 70(4):2138–2161, 2022.