Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Online Advertisement Allocation Under Customer Choices and Algorithmic Fairness

Xiaolong Li, Ying Rong, Renyu Zhang, Huan Zheng

informs
https://pubsonline.informs.org/journal/mnsc

MANAGEMENT SCIENCE
Vol. 71, No. 1, January 2025, pp. 825–843
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

Online Advertisement Allocation Under Customer Choices and Algorithmic Fairness

Xiaolong Li,[a] Ying Rong,[b,*] Renyu Zhang,[c,*] Huan Zheng[b,*]

[a] Institute of Operations Research and Analytics, National University of Singapore, Singapore 117602; [b] Antai College of Economics and Management, Data-Driven Management Decision-Making Lab, Shanghai Jiao Tong University, Shanghai 200030, China; [c] CUHK Business School, The Chinese University of Hong Kong, Hong Kong, China

*Corresponding authors

Contact: oralxi@nus.edu.sg, https://orcid.org/0009-0006-7570-3554 (XL); yrong@sjtu.edu.cn, https://orcid.org/0000-0003-3360-1868 (YR); philipzhang@cuhk.edu.hk, https://orcid.org/0000-0003-0284-164X (RZ); zhenghuan@sjtu.edu.cn, https://orcid.org/0000-0003-4803-3787 (HZ)

Received: September 24, 2022
Revised: April 26, 2023; August 27, 2023; September 19, 2023
Accepted: September 24, 2023
Published Online in Articles in Advance: April 24, 2024

https://doi.org/10.1287/mnsc.2021.04091

Abstract. Advertising is a crucial revenue source for e-commerce platforms and a vital online marketing tool for their sellers. In this paper, we explore dynamic ad allocation with limited slots upon each customer's arrival for an e-commerce platform, where customers follow a choice model when clicking the ads. Motivated by the recent advocacy for the algorithmic fairness of online ad delivery, we adjust the value from advertising by a general fairness metric evaluated with the click-throughs of different ads and customer types. The original online ad-allocation problem is intractable, so we propose a novel stochastic program framework (called *two-stage target-debt*) that first decides the click-through targets and then devises an ad-allocation policy to satisfy these targets in the second stage. We show the asymptotic equivalence between the original problem, the relaxed click-through target optimization, and the fluid-approximation (Fluid) convex program. We also design a debt-weighted offer-set algorithm and demonstrate that, as long as the problem size scales to infinity, this algorithm is (asymptotically) optimal under the optimal first-stage click-through target. Compared with the Fluid heuristic and its resolving variants, our approach has better scalability and can deplete the ad budgets more smoothly throughout the horizon, which is highly desirable for the online advertising business in practice. Finally, our proposed model and algorithm help substantially improve the fairness of ad allocation for an online e-commerce platform without significantly compromising efficiency.

History: Accepted by Jeannette Song, operations management.
Funding: Y. Rong is supported by the National Natural Science Foundation of China [Grants 72025201, 72331006, and 72221001]. R. Zhang is grateful for the financial support from the Hong Kong Research Grants Council General Research Fund [Grants 14502722 and 14504123] and the National Natural Science Foundation of China [Grants 72293560 and 72293565]. H. Zheng is supported by the National Natural Science Foundation of China [Grants 72231003, 72325003, and 72221001].
Supplemental Material: The online appendix and data files are available at https://doi.org/10.1287/mnsc.2021.04091.

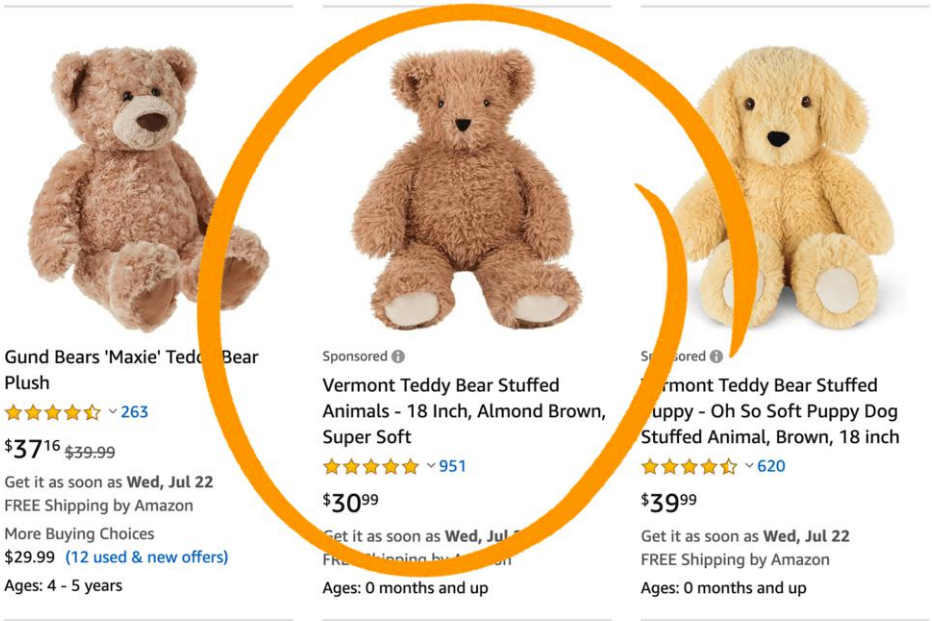
Keywords: online advertising platform • assortment optimization • algorithmic fairness • online convex optimization • mean reverting

1. Introduction

The past 10 years have witnessed the rapid growth of internet technology and smartphone penetration, both of which have driven online advertising to become an unprecedentedly enormous trillion-dollar industry[1] that has an enormous impact on the entire economy.

One important online advertising format is e-commerce advertising, which is designed to drive "top-of-tunnel" traffic to convert into product sales. For instance, Amazon Advertising provides "sponsored products"[2] where advertisers pay Amazon to promote their products by listing the ads both within shopping results and on product pages (see Figure 1). The sponsored-product ads use the cost-per-click (CPC) mechanism, under which advertisers pay a fee to the platform when customers click their ads. Advertisers choose the campaign budgets and how much to bid per click. Amazon also allows advertisers to set the keywords and products so that the ad can be more efficiently matched with customer queries. Alternatively, advertisers can select automatic targeting to allow Amazon to match their ads to relevant search terms. This advertising service is an important source of revenue for Amazon; it contributed $14.1 billion (5.02%) to its *annual net sales* in 2019.[3] As another example, Facebook launched its Dynamic Ads service to promote

Figure 1. (Color online) An Example of Sponsored Products on Amazon



advertisers' products to people who expressed an interest in relevant keywords or similar products.[4] Dynamic Ads will automatically choose products from the catalog provided by advertisers and display them to customers.

Such large-scale e-commerce advertising platforms generally run thousands of advertising campaigns for different advertisers simultaneously. Each campaign is usually associated with (1) a budget that the advertiser wishes to spend as much as possible of during the campaign horizon, and (2) a bid per CPC that dictates how much ad budget should be deducted upon each user click. The advertising platform dynamically allocates its ad spaces (i.e., customer impressions) to the ads whose campaigns are active. As discussed earlier, advertisers may require the platform to target their advertising campaign and ads to specific customer segments (specified by such features as location, age, and browsing, searching, and purchasing histories).

It is also not uncommon for advertisers and, thus, the platform to set click-through requirements for the ads (i.e., the minimal number of click-throughs during the entire ad campaign). For instance, Microsoft provides a Partner Incentive Cooperative Marketing Fund (Co-op) to subsidize its partners in whose website the number of click-throughs for Microsoft's ad is above 250 during the promotion events (Microsoft 2020). In addition, the number of click-throughs for an ad substantially affects the long-term retention of the advertiser, which prompts the platform to devise the ad-allocation policy to secure a certain number of click-throughs for each ad (see, e.g., Ye et al. 2023).

To efficiently allocate its ad spaces, an online advertising platform faces the central operations problem of dynamically selecting a set of ads, which we refer to as an offer-set, displayed to each arriving customer in order to generate the highest total value throughout the planning horizon, subject to advertising budgets and click-through requirements for different ads.

Solving this problem presents a twofold challenge. First, under customer-choice behavior, when an offer-set is displayed, the platform has to carefully balance the notorious trade-off in assortment optimization between expanding the offer-set to enlarge the market share and keeping it small enough to reduce cannibalization between different ads. Second, the click-through requirements, budget constraints, and advertisers' targeting rules altogether raise the difficulty of even searching for a feasible (but not necessarily optimal) ad allocation. In addition, we emphasize that the commonly used fluid-approximation (Fluid) approaches also face substantial computational challenges because the number of decision variables increases exponentially with the cardinality of the offer-set.

In addition to optimizing total advertising value subject to budget and click-through requirement constraints, the online platform also needs to address the fairness and discrimination concerns of its advertising/machine learning algorithms. It is well documented in the literature and in practice that a common source of algorithmic discrimination or bias in online advertising is that advertisers can target (or exclude) particular groups of users for their ads (e.g., Speicher et al. 2018, Dave 2021). The particular groups could be those classes harmed from counterfactual disparities, or legally protected characteristics, such as race and gender (e.g., Nilforoshan et al. 2022). Simple controls are insufficient to counter this issue because the platform's underlying ad-allocation algorithm for optimizing certain business objectives, such as advertising revenue or advertisers' return on investment, may automatically skew ad delivery to certain user groups (see, e.g., Lambrecht and Tucker 2019, Imana et al. 2021).

The main goal of this paper is to explore the e-commerce ad allocation of an advertising platform under customer choices and concerns for algorithmic fairness. Motivated by online e-commerce advertising practices, we seek to address the following key research question:

Taking into account algorithmic fairness, how should a platform dynamically personalize the ad offer-sets of each customer impression to maximize the fairness-adjusted value (FV) from advertising throughout a planning horizon in the presence of budget constraints and click-through requirements for different ads?

1.1. Main Contributions

In this paper, we present a general stochastic program model to study the complex dynamic ad-allocation problem. Our key contribution is proposing a novel *two-stage target-debt* (TTD) framework that yields a simple, computationally efficient, and asymptotically optimal policy for addressing this otherwise intractable online ad-allocation problem for an e-commerce platform. This TTD framework carefully combines three ideas: click-through target optimization, compact reformulation, and a debt-weighted offer-set algorithm.

Click-Through Target Optimization. In the first stage, we approximate the original problem as an (auxiliary) click-through target optimization in which the platform decides the click-through goal (targets) for each ad-customer pair to maximize the expected FV. Such a reduction to a deterministic convex program is not only tractable but also provides a new upper bound for the original (intractable) stochastic program for ad-allocation optimization.

Compact Reformulation. We characterize the necessary and sufficient condition under which the click-through targets are feasible. The characterization is surprisingly simple for most commonly used choice models such as multinomial logit (MNL), independent, and generalized attraction choices. In these cases, we can efficiently solve the click-through target optimization using the first-stage convex program because the scale of our reformulation increases linearly (rather than exponentially) with the number of ads.

Debt-Weighted Offer-Set Algorithm. In the second stage, an ad offer-set is displayed to each user upon arrival in order to satisfy the optimal click-through targets set in the first stage. We propose a simple and effective algorithm, referred to as the *debt-weighted offer-set* (DWO) policy. This policy dynamically assigns a "debt" to each click-through target that measures the difference between the realized total click-throughs and the endogenous target set in the first-stage convex program. Then, a standard offer-set/assortment optimization problem is solved to maximize a debt-weighted value function upon the arrival of each customer.

We prove that our TTD framework yields an asymptotic optimal policy (i.e., the DWO algorithm initialized with the optimal click-through targets) for the online ad-allocation problem. Consistent with most existing debt-weighted algorithms (e.g., Zhong et al. 2017, Jiang et al. 2023), our DWO algorithm satisfies the feasibility and approachability of click-through targets. Our refined analysis also provides the optimality guarantee of the TTD framework. Leveraging the *exact approachability* of the DWO algorithm to *exactly* meet the (feasible) target set in the first stage, we establish new intrinsic connections and asymptotic equivalence of the original ad-allocation problem, the first-stage convex program, and an auxiliary Fluid convex program, implying that the theoretical upper bound of FV characterized by the first-stage convex program can be achieved. If the first-stage click-through target vector is only feasible, but not optimal, the associated DWO policy will not incur any additional optimality loss on top of that from the suboptimal target in the asymptotic regime.

The TTD framework we propose in this paper is computationally scalable if customer choices follow commonly adopted models such as MNL, independent, and generalized attraction choices. Through numerical experiments, we demonstrate that our algorithm outperforms the commonly adopted Fluid-based algorithms (e.g., Liu and Van Ryzin 2008, Jasin and Kumar 2012, Bumpensanti and Wang 2020, Balseiro et al. 2023) in terms of performance, robustness, and computational efficiency for most problem instances. Our numerical experiments also demonstrate that the proposed policy depletes the budgets much more smoothly than the benchmarks over the entire planning horizon. This highlights the practical applicability of our approach because smooth budget depletion is a desirable property for the real-world online advertising business. In addition, our approach helps substantially improve the algorithmic fairness of ad allocation for an online e-commerce platform without compromising its efficiency much, achieving high FV with low variance. Hence, our computationally light TTD framework well handles the notorious bias and discrimination issues in online ad allocation.

In summary, the key takeaway from this paper is that the proposed TTD framework, which induces a two-stage stochastic program reformulation combining click-through target optimization and the debt-weighted off-set algorithm, can efficiently address the ad-allocation problem for e-commerce platforms to improve the FV from advertising. Our approach is simple, efficient, and scalable, with a provable optimality guarantee and strong numerical performance.

The rest of this paper is organized as follows. We review the related literature in Section 2. We introduce the model in Section 3, and we propose the two-stage stochastic program reformulation in Section 4. We study the optimal ad-allocation policy in Section 5, and we present the numerical studies in Section 6. Section 7 concludes. All proofs are relegated to the online appendices. Throughout this paper, we use **boldface** to represent vectors and matrices. For notational conciseness, we do not differentiate a random variable and its realization whenever there is no ambiguity.

2. Literature Review

This paper proposes a general framework and an efficient algorithm to study optimal online ad allocation for an e-commerce platform under algorithmic fairness concerns. Our paper is primarily related to four streams of research in the literature: (1) ad allocation for online advertising platforms, (2) algorithmic discrimination/bias in online allocation, (3) resource allocation with individualized service-level constraints, and (4) (dynamic) personalized assortment optimization. Papers in the literature generally focus on one or more of the four topics given, whereas our work contributes to all four streams of literature jointly.

Ad allocation is a central challenge for online advertising platforms. For example, Nakamura and Abe (2005) propose an ad-targeting approach based on linear programming that achieves high click-through rates by optimizing ad-display probabilities. For maximizing the reach of customers and minimizing the variance of the outcome simultaneously in targeted advertising, Turner (2012) formulates an ad-planning problem with a quadratic objective to spread ads across all targeted customer types. Balseiro et al. (2014) formalize an ad-exchange problem as a multiobjective stochastic control problem considering both the revenue from exchange and click-through rates, and they derive an efficient policy for online ad allocation with uncertainty. For dealing with uncertainty, Shen et al. (2021a) propose an integrated planning model with a distributionally robust chance-constrained program in online ad allocation. Hojjat et al. (2017) consider a new contract to allow advertisers to specify the number of unique individuals who should see their ad and the minimum number of times each individual should be exposed. Shen et al. (2021b) deal with customers' ad-clicking behavior by an arbitrary point-inflated Poisson regression model, and they solve a mixed-integer nonlinear program (non-LP) for optimal ad allocation. We refer interested readers to Choi et al. (2020) for a comprehensive review of this literature. The key modeling difference of our paper from this literature is that using choice models, we clearly model the click-through behaviors of a customer in the presence of multiple ads.

As mentioned earlier, algorithmic discrimination/bias in online allocation has received increased scrutiny in recent literature. To mitigate the algorithmic discrimination/bias in online advertising, Lejeune and Turner (2019) derive a Gini index–based metric to measure how well dispersed the impressions are allocated across audience segments, and they formulate an optimization problem to maximize the spread of impressions across targeted audience segments while minimizing demand shortfalls. Balseiro et al. (2021) use a nonlinear regularizer as a fairness measure, and they design an online resource-allocation algorithm to maximize the weighted objective of efficiency and fairness subject to the resource constraints. Bateni et al. (2022) adopt a weighted proportional fairness metric under the setting that a platform dynamically allocates to budgeted buyers a collection of goods that arrive to the platform online. Ma et al. (2020) consider an online matching problem with concerns of agent-group fairness by defining two different service-level objectives as the metrics for long-run and short-run fairness. In this paper, we offer a new slant on mitigating algorithmic discrimination/bias in online advertising by considering different fairness metrics in the literature and *disparate impact*, which identifies unintentional bias of an algorithm (see, e.g., Feldman et al. 2015).

The resource-allocation problem of meeting service-target constraints in the face of uncertain demand has been extensively studied in the inventory literature (see, e.g., Alptekinoğlu et al. 2013). Leveraging Blackwell's approachability theorem, Zhong et al. (2017) characterize the optimal safety-stock level with individual type-II service-level constraints. Lyu et al. (2019) and Lyu et al. (2022), respectively, extend both the approach and the results to the context of type-I service-level constraints and process flexibility. Utilizing a semi-infinite linear program formulation, Jiang et al. (2023) generalize and unify models in this literature and propose a simple randomized rationing policy to meet general service-level constraints, including type-I and type-II constraints, and beyond. We contribute to this strand of the literature by generalizing the concept of service-level constraints to incorporate customer choice uncertainty and ad allocation through assortment planning.

We also propose a debt-weighted offer-set algorithm and demonstrate its optimality for meeting the endogenous service targets and for generating the total payoff for the platform. Our debt-weighted offer-set policy and other existing debt-weighted policies are closely related to the well-known max-weight policy proposed in the queueing literature (see, e.g, Stolyar 2004), which is also commonly used in the resource-allocation and capacity-planning literature. Dai and Lin (2005) propose a max-weight policy for dynamically allocating service capacities in a stochastic processing network. By applying a max-weight policy with good performance, Shi et al. (2019) analyze the design of sparse flexibility structures in a multiperiod make-to-order production system. Xu and Zhong (2020) formulate a generalized version of max-weight policy to study the impact of information constraints and memories on dynamic resource allocation. In this paper, we adopt target-based weight updating, which utilizes both the targets optimized in the first-stage program and the realized click-throughs—rather than just the current states, as shown in this literature—to adjust the weights in each period.

Over the past 10 years, online e-commerce platforms have typically provided numerous products for customers to choose from (Feldman et al. 2022). Manufacturing firms have also expanded their product lines because of business trends (e.g., fast fashion (Caro et al. 2014)) or technology revolution (e.g., 3D printing (Dong et al. 2022)). The ever-expanding product pool makes personalized assortments more attractive. Therefore, personalized-assortment optimization has also been receiving increased attention in the literature.

Leveraging the competitive-ratio framework, Golrezaei et al. (2014) propose inventory-balancing algorithms that guarantee the worst-case revenue performance without any forecast of the customer-type distribution. Bernstein et al. (2019) combine dynamic assortment planning, demand learning, and customer-type clustering in a Bayesian framework, and they propose a prescriptive assortment-personalization approach for online retailing. Meanwhile, Kallus and Udell (2020) consider a dynamic assortment-personalization problem in high dimensions as a discrete contextual bandit problem. Chen et al. (2023) propose an inventory-protection algorithm with a bounded competitive ratio for a new checkout-recommender system. Gallego et al. (2015b) study a general personalized resource-allocation model with customer choices, and they introduce algorithms for solving a choice-based linear program. Considering the uncertainty in estimating the MNL choice model, Cheung and Simchi-Levi (2017) propose a Thompson sampling–based policy to estimate the latent parameters by offering a personalized assortment. We contribute to this literature by proposing a new, two-stage stochastic program framework (i.e., TTD) to study the ad allocation problem. Moreover, we design a novel DWO policy that proves to be asymptotically optimal and generates values with lower variance than the Fluid benchmarks commonly adopted in the literature.

3. Model

3.1. Model Setup

The Platform and Its Customers. We consider an e-commerce platform such as Amazon or Facebook Marketplace that matches its user traffic with product advertisements. Our model, however, can be straightforwardly applied to the setting of product recommendation. Throughout the planning horizon, there are T customer impressions (also called users or viewers) arriving at the platform sequentially. Therefore, we say customer t arrives in time t. Without loss of generality, we assume T is deterministic and known to the platform. Customers are segmented into m types based on their demographic information (e.g., age, gender, location) and behavior on the platform (e.g., average spending per year, product preferences, average time spent on the platform per year). We denote $\mathcal{M} := \{1, 2, \ldots, m\}$ as the set of all customer segments. For each customer t, type $j(t)$ is *i.i.d.* and follows a discrete distribution on $\mathcal{M}$, with $\mathbb{P}(j(t) = j) = p^j > 0$, where $j \in \mathcal{M}$ and $\sum_{j\in\mathcal{M}} p^j = 1$.

Advertisements. At the beginning of the horizon, advertisers launch a set of ad campaigns, which we denote as $\mathcal{N} := \{1, 2, \ldots, n\}$. For each ad campaign $i \in \mathcal{N}$, its advertiser sets $B_i > 0$ as the total budget and $b_i > 0$ as the bid price, which is a proxy for the ex-post price per click paid by the advertiser to the platform, with the exact auction setting abstracted away from our model.[5] Specifically, B_i is the maximum advertising fee the advertiser will pay the platform throughout the ad campaign's life, and the budget will be depleted by b_i upon each click by a customer. That is, the platform adopts the CPC mechanism, which is commonly used in online advertising. Furthermore, the ads are placed in specific slots exclusively allocated to advertising.

Offer-Sets. Upon the arrival of user t, the platform observes its type $j(t)$ and decides a (possibly randomized) set of ads/sponsored products displayed to this user (which we call an *offer-set*), denoted by $S(t) \in \mathfrak{S}^{j(t)} \subset 2^{\mathcal{N}}$, where $\mathfrak{S}^j$ is the set of all feasible offer-sets to type-j customers, and $2^{\mathcal{N}}$ is the power set of $\mathcal{N}$. Throughout our analysis, we make the following assumption on the structure of $\mathfrak{S}^j$.

Assumption 1. *If $S \in \mathfrak{S}^j$, then for any subset $S' \subset S$, we also have $S' \in \mathfrak{S}^j$.*

Assumption 1 implies that $\emptyset \in \mathfrak{S}^j$; that is, the platform may not display any ad to a customer of type j. We may impose additional structural constraints on $\mathfrak{S}^j$.

Of particular importance is the cardinality constraint (i.e., the total number of ads displayed to customers cannot exceed K; see, e.g., Rusmevichientong et al. 2010, Wang 2012, Sumida et al. 2021), which is prevalent in the online advertising setting where the platform cannot allocate more ads to a customer than the number of available ad-impression slots. We will demonstrate how to handle this cardinality constraint in our theoretical and numerical analyses.

Click-Throughs. For user t, if an offer-set $S(t)$ is displayed (see Figure 1), the user may or may not click some ads in the set $S(t)$. We denote $y_i^j(t)$ as the indicator of one click received by ad i from a type-j customer in time t. Therefore, $y_i^j(t) = 1$ only if $j(t) = j$, $i \in S(t)$, and customer t clicks ad i. Otherwise, $y_i^j(t) = 0$. We denote the click-through matrix in time t as $\boldsymbol{y}(t) := (y_i^j(t) : i \in \mathcal{N}, j \in \mathcal{M})$. We do not specify any structure of the customer click-through behavior. For each customer type j, each offer-set S, and each ad $i \in S$, we denote the expected click-throughs of ad i in an offer-set S from a type-j customer as $\phi_i^j(S) := \mathbb{E}_{\mathsf{D}_y}[y_i^j(t) | j(t) = j, S(t) = S]$, where D_y is the click-through distribution. We denote $\mathsf{D}_{(j,y)}$ as the joint customer type and click-through distribution.

We assume customers exhibit (conditionally) independent and stationary click-through behaviors. Specifically, conditioned on the realized offer-sets $\{S(\tau) : 1 \leq \tau \leq T\}$, the click-throughs, $\boldsymbol{y}(t)$'s, are independent across time t. Furthermore, conditioned on the same offer-set, $\boldsymbol{y}(t)$'s are identically distributed with respect to the time t, that is, for any set of click-through outcomes $\mathcal{Y}$, any realized customer type j, any offer-set S, and any $t \neq \tau$, $\mathbb{P}[\boldsymbol{y}(t) \in \mathcal{Y} | j(t) = j, S(t) = S] = \mathbb{P}[\boldsymbol{y}(\tau) \in \mathcal{Y} | j(\tau) = j, S(\tau) = S]$. By stationarity, the function $\phi_i^j(\cdot)$ is independent of t. If $i \notin S$, $\phi_i^j(S) = 0$ by definition.

Ad Targeting and Click-Through Requirement. From the advertisers' perspective, they target ads to the relevant customer segments based on their demographic information, past behavioral patterns, and potential interests (see, e.g., Choi et al. 2020). Furthermore, consistent with advertising practices (e.g., Microsoft 2020), the advertiser may require that ad i receives at least $\eta_i^{\mathcal{C}}$ click-throughs throughout the planning horizon for targeting a set of customer types $\mathcal{C} \in \mathfrak{K}_i$, where $\mathfrak{K}_i \subset 2^{\mathcal{M}}$ denote the set of all customer segment subsets $\mathcal{C}$ on which the advertiser sets a positive click-through requirement $\eta_i^{\mathcal{C}} > 0$ of ad i. Mathematically, the targeting and click-through requirement of ad i can be formalized as $\sum_{t=1}^T \sum_{j \in \mathcal{C}} y_i^j(t) \geq \eta_i^{\mathcal{C}}$ for any $i \in \mathcal{N}$ and $\mathcal{C} \in \mathfrak{K}_i$. In practice, $\mathfrak{K}_i$ often only contains either $\mathcal{M}$ (i.e., the requirement of total click-throughs) or some nonoverlapping subsets of $\mathcal{M}$. For example, Pampers may additionally ask the platform to target its diaper ads to new parents and babysitters. Note that because of the randomness in the customer types and choices, the targeting and click-through requirements given may not be satisfied with probability one. Hence, we model the click-through requirements in the expected sense, that is, $\mathbb{E}[\sum_{t=1}^T \sum_{j \in \mathcal{C}} y_i^j(t)] \geq \eta_i^{\mathcal{C}}$, which is a common modeling approach in the literature on resource allocation to meet service target constraints (e.g., Zhong et al. 2017, Jiang et al. 2023). We will also show in Theorem 2 that our proposed policy can indeed satisfy the click-through requirements almost surely in the asymptotic regime where the problem size scales to infinity. In Online Appendix B, we illustrate how to incorporate the click-through requirements as soft constraints, that is, by adding a penalty term $-(\eta_i^{\mathcal{C}} - \sum_{t=1}^T \sum_{j \in \mathcal{C}} y_i^j(t))^+$ into the objective function.

Moreover, large-scale online advertising platforms (e.g., Facebook and Google) have recently tightened their controls to prevent advertisers from excluding some user segments in their target in order to reduce lawsuits and regulatory probes into discrimination. Thus, in most cases, the total number of click-through requirement constraints $|\mathfrak{K}_i|$ is at most *linear* (instead of exponential) in the total number of customer segments m, making our model and solution approach scalable. In many scenarios, the advertising contract specifies the minimal click-through requirement. For instance, Microsoft (as an advertiser) requires its partners (i.e., the advertising platforms where Microsoft runs its advertising campaigns) to earn at least 250 click-throughs during one ad campaign to be qualified to receive the support through its co-op.

Advertising Value and Fairness. The total value of online advertising generated throughout the planning horizon depends on matching the n ads with T customers. Specifically, each click of ad i by a type-j customer generates a value of $r_i^j \geq 0$, which is allowed to be both ad and customer type specific. The interpretation of r_i^j, which can be quite general, includes the following scenarios as special cases. For the case where the value is the total advertising revenue of the platform, $r_i^j = b_i$ for each $i \in \mathcal{N}$. For the case where the value is the total advertising return of the advertisers (see, e.g., Hao et al. 2020), r_i^j is interpreted as the value of one click-through for ad i by a type-j customer to its advertiser. Therefore, the total value of online advertising is given by $\sum_{t=1}^T \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} r_i^j y_i^j(t)$.

A salient feature of our model is that, in addition to total advertising value, the platform may also be concerned about the *fairness* of the system. For example, the recent advocacy on machine learning/algorithmic fairness postulates that customers who are considered minorities should have sufficient click-throughs/conversions in a recommender/advertising system; otherwise, their needs cannot be well taken care of because of data sparsity (see, e.g., Lambrecht and Tucker 2019). In a similar vein, advertisers generally prefer receiving impressions that are evenly spread across their targeted customer types (see, e.g., Lejeune and Turner 2019). To account for such algorithmic fairness, we introduce a general fairness metric $F(\cdot): \mathbb{R}^{nm} \longmapsto \mathbb{R}$, which is a function of the per-customer-impression click-through matrix $\overline{\boldsymbol{y}} := (\overline{y}_i^j : i \in \mathcal{N}, j \in \mathcal{M})$, where $\overline{y}_i^j := \frac{1}{T}\sum_{t=1}^T y_i^j(t)$ is the per-customer-impression click-through of ad i by type-j customers. Throughout our analysis, we make the following assumption on the concavity of $F(\cdot)$.

Assumption 2. *The fairness metric $F(\cdot): \mathbb{R}^{nm} \longmapsto \mathbb{R}$ is a concave function.*

We remark that the concavity of $F(\cdot)$ is a mild assumption, which can be satisfied by most of the commonly adopted fairness metrics in the literature, as detailed in Online Appendix C. For example, $F(\cdot)$ can be modeled as *allocative fairness* measures, for example, max-min fairness (see, e.g., Young 1995, Kumar and Kleinberg 2000, and Bertsimas et al. 2012), Gini mean difference (GMD) fairness (see, e.g., Atkinson 1970) to keep the advertisers from exclusively targeting a small subset of user groups, *disparate impact* measures (see, e.g., Rubin 1978, Feldman et al. 2015) to reduce algorithmic discrimination, and so on (see, e.g., Balseiro et al. 2021). Therefore, we measure the fairness in a *per-customer-impression* sense and evaluate the *per-customer-impression* FV from advertising as

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} r_i^j y_i^j(t) + \lambda F(\overline{\boldsymbol{y}})$$

where $\lambda \geq 0$ parameterizes the trade-off between efficiency and fairness. The smaller (respectively, larger) the λ, the higher weight the platform puts on efficiency (respectively, fairness). In the extreme case where $\lambda \longrightarrow 0$ (respectively, $\lambda \longrightarrow +\infty$), the system is purely efficiency driven (respectively, fairness driven).

3.2. Stochastic Program Formulation

We consider the (randomized) policies Π. First, we define the realized history until time t as $\mathcal{H}_{t-1} := \{(j(\tau), S(\tau), \boldsymbol{y}(\tau)) : 1 \leq \tau \leq t-1\}$. By convention, $\mathcal{H}_0 = \emptyset$. In time t, a policy $\pi \in \Pi$ maps the realized customer type $j(t)$ and the realized history $\mathcal{H}_{t-1}$ to a distribution on all feasible offer-sets to a type-$j(t)$ customer, $\mathfrak{S}^{j(t)}$, that is, $S(t) = \pi(j(t), \mathcal{H}_{t-1})$. Note that deterministic policies are special cases of Π, which map $(j(t), \mathcal{H}_{t-1})$ to a deterministic offer-set in $\mathfrak{S}^{j(t)}$. Sometimes, it is useful to spell out the dependence of the click-through outcomes in time t, $\boldsymbol{y}(t)$, on the history $\mathcal{H}_{t-1}$ and the policy π. We use $y_i^j(t|\pi)$ as the number of click-throughs for ad i by a type-j customer in time t, given that the history is $\mathcal{H}_{t-1}$ and the offer-set displayed to a type-$j(t)$ customer is $S(t) = \pi(j(t), \mathcal{H}_{t-1})$. Likewise, we define $\overline{y}_i^j(\pi) := \frac{1}{T}\sum_{t=1}^T y_i^j(t|\pi)$ as the per-customer click-throughs of ad i by type-j customers in the entire horizon under policy π. We denote $\overline{\boldsymbol{y}}(\pi) := (\overline{y}_i^j(\pi) : i \in \mathcal{N}, j \in \mathcal{M})$ as the per-customer-impression click-through matrix under policy π.

Of particular importance are the (randomized) *static* policies, the set of which we denote as $\Pi_{\mathsf{static}} \subset \Pi$. Specifically, if $\pi \in \Pi_{\mathsf{static}}$, the offer-set displayed in time t, $S(t) = \pi(j(t), \mathcal{H}_{t-1})$ is independent of (i) time t and (ii) the history $\mathcal{H}_{t-1}$, solely conditioned on the realized customer type $j(t)$. Hence, for $\pi \in \Pi_{\mathsf{static}}$, we can drop the history $\mathcal{H}_{t-1}$ to write $\pi(j(t), \mathcal{H}_{t-1})$ as $\pi(j(t))$. The word "static" also refers to that the distribution of $S(t)$ is stationary with respect to time t for each customer type j. Therefore, we denote π_{static} as a static policy to distinguish it from an arbitrary policy $\pi \in \Pi$.

We are now ready to formulate the platform's ad-allocation problem as a multiperiod stochastic program. Specifically, the platform seeks to optimize the total expected FV of online advertising throughout the planning horizon

$$\max_{\pi\in\Pi}\ \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} r_i^j y_i^j(t|\pi) + \lambda F(\overline{\boldsymbol{y}}(\pi))\right]$$

$$\text{s.t.}\ \frac{1}{T}\sum_{t=1}^{T}\sum_{j\in\mathcal{M}} b_i y_i^j(t|\pi) \leq \frac{B_i}{T}, \quad \textit{almost surely} \text{ for each } i \in \mathcal{N},$$

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\sum_{j\in\mathcal{C}} y_i^j(t|\pi)\right] \geq \frac{\eta_i^{\mathcal{C}}}{T}, \quad \text{for each } i \in \mathcal{N} \text{ and } \mathcal{C} \in \mathfrak{K}_i \qquad (\mathcal{OP})$$

where the first term in the objective is the total per-customer-impression value from advertising, which we call the *efficiency* of policy π denoted by $\mathcal{E}(\pi) := \mathbb{E}[\frac{1}{T}\sum_{t=1}^T\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} r_i^j y_i^j(t|\pi)]$, the second term is the *fairness* of policy π denoted by $\lambda \cdot \mathcal{F}(\pi) := \lambda \cdot \mathbb{E}[F(\overline{\boldsymbol{y}}(\pi))]$, and all the expectations, including the following ones, are taken with respect to π and $\mathsf{D}_{(j,y)}$ unless otherwise stated. Hence, the total FV under policy π is given by $\mathcal{V}(\pi) := \mathcal{E}(\pi) + \lambda \cdot \mathcal{F}(\pi)$. We also remark that the first constraint of $(\mathcal{OP})$ refers to the budget constraint of each ad, and the second refers to the click-through requirement for each ad with respect to different sets of customer types in the expected sense. We denote the optimal FV of $(\mathcal{OP})$ as $\mathcal{V}^* = \limsup_{\pi\in\Pi}\mathcal{V}(\pi)$ and the optimal policy (if it exists) as $\pi^* = \arg\max_{\pi\in\Pi}\mathcal{V}(\pi)$.

Roadmap to Solve $(\mathcal{OP})$. For the rest of this paper, our goal is to design a two-stage ad-allocation framework (i.e., TTD) to find a policy π that achieves the optimal FV, $\mathcal{V}^*$, while satisfying the budget- and click-through-requirement constraints. As detailed here, our TTD framework to solve ($\mathcal{OP}$) can be decomposed into the following two stages:

- **First-stage click-through target optimization,** where we solve a deterministic (but *nonequivalent*) convex program to obtain the optimal click-through target of the ads and customer types (Section 4).
- **Second-stage offer-set allocation,** where we adaptively decide the offer-set displayed to each customer based on how far away the click-throughs are from the optimal targets obtained in the first stage by a debt-weighted offer-set policy (Section 5.1).

We demonstrate that the TTD framework is asymptotically optimal (Section 5.2) and enjoys impressive performance in the nonasymptotic regime compared with the commonly adopted benchmarks in the existing literature (Section 6).

4. Reformulation and Feasibility Conditions

In this section, we propose a novel reformulation of the original intractable ad-allocation program ($\mathcal{OP}$) to maximize the expected FV as a much simpler, two-stage convex optimization. The core of our reformulation is to introduce the auxiliary click-through targets of the ads by different customer types and then to design an online debt-based ad-allocation algorithm to achieve these targets.

4.1. Problem Reformulation with Click-Through Targets

To solve the dynamic ad-allocation problem ($\mathcal{OP}$), a commonly adopted approach in the literature is to consider a fluid approximation of this problem and solve the Fluid problem by linear programming (choice-based linear programming (CDLP); see, for example, Liu and Van Ryzin 2008; in our case, the Fluid problem is a convex program). The Fluid-based formulation of ($\mathcal{OP}$) is provided by ($\mathcal{OP}_{\mathsf{Fluid}}$) in Section 5.3 as an auxiliary problem to demonstrate the optimality of our proposed algorithm. With the cardinality constraint on the feasible offer-sets, one difficulty using (the analogs of) CDLP is that the number of variables (i.e., the probability of each offer-set for all customer types) quickly explodes as the number of advertisements increases, even when the choice model is as simple as the independent choice or MNL model.

To tackle the aforementioned challenges of the standard Fluid approach, we develop a novel reformulation of ($\mathcal{OP}$) that transforms the original problem as a two-stage convex optimization by introducing click-through targets associated with each ad-customer pair as auxiliary decision variables. Such reformulation also proves useful in designing our asymptotically optimal ad-allocation algorithm. Specifically, we define $\boldsymbol{\alpha} := (\alpha_i^j, i \in \mathcal{N}, j \in \mathcal{M}) \in \mathbb{R}_+^{nm}$, where α_i^j refers to the (virtual) target for the per-customer-impression number of click-throughs for ad i by type-j customers. Therefore, the platform operationalizes its ad-allocation algorithm such that the total number of click-throughs for ad i by type-j customers exceeds $T\alpha_i^j$. We define the concave per-customer-impression FV associated with click-through target vector $\boldsymbol{\alpha}$ as

$$\mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha}) := \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} r_i^j \alpha_i^j + \lambda F(\boldsymbol{\alpha}), \tag{1}$$

where the first term captures efficiency, the second captures fairness with respect to the click-through target vector $\boldsymbol{\alpha}$, and the subscript "CT" stands for *click-through target.* We transform ($\mathcal{OP}$) into the following (nonequivalent) optimization problem:

$$\begin{aligned} \max_{\pi\in\Pi,\, \boldsymbol{\alpha}\geq \mathbf{0}} \quad & \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha}) \\ \text{s.t.} \quad & \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} y_i^j(t|\pi)\right] \geq \alpha_i^j, \ \text{for each } i\in\mathcal{N}, j\in\mathcal{M}, \\ & b_i \sum_{j\in\mathcal{M}} \alpha_i^j \leq \frac{B_i}{T}, \ \text{for each } i \in \mathcal{N}, \\ & \sum_{j\in\mathcal{C}} \alpha_i^j \geq \frac{\eta_i^{\mathcal{C}}}{T}, \ \text{for each } i\in\mathcal{N} \text{ and } \mathcal{C}\in\mathfrak{K}_i \end{aligned} \tag{2}$$

Comparing (2) with ($\mathcal{OP}$) reveals that our reformulation relaxes the sample path–based objective function and constraints in the original problem with their counterparts characterized by the per-customer-impression click-through target vector $\boldsymbol{\alpha}$. To ensure that the reformulation is close enough to the original problem and that the click-through targets are achievable, we introduce an additional constraint: the expected click-throughs per customer should meet the click-through targets, as specified by the first constraint of (2), that is, $\mathbb{E}[\frac{1}{T}\sum_{t=1}^{T} y_i^j(t|\pi)] \geq \alpha_i^j$. It is still challenging to characterize when this constraint can be satisfied. Therefore, we further modify (2) by replacing this constraint with one for the expected number of click-throughs under static policies Π_{static}. Specifically, we replace the first constraint of (2) with

$$\mathbb{E}[y_i^j(t|\pi_{\mathsf{static}})] \geq \alpha_i^j, \ \text{for each } i\in\mathcal{N}, j\in\mathcal{M}. \tag{3}$$

One should note that the expected Click-Through Target Constraint (3) is independent of time t. Hence, we reformulate (2) by further modifying the click-through target constraint as follows:

$$\max_{\pi\in\Pi_{\mathsf{static}},\,\boldsymbol{\alpha}\geq 0} \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})$$
$$\text{s.t. } \mathbb{E}[y_i^j(t|\pi)] \geq \alpha_i^j, \text{ for each } i\in\mathcal{N}, j\in\mathcal{M},$$
$$b_i\sum_{j\in\mathcal{M}}\alpha_i^j \leq \frac{B_i}{T}, \text{ for each } i\in\mathcal{N},$$
$$\sum_{j\in\mathcal{C}}\alpha_i^j \geq \frac{\eta_i^{\mathcal{C}}}{T}, \text{ for each } i\in\mathcal{N} \text{ and } \mathcal{C}\in\mathfrak{K}_i. \qquad (2\mathcal{SSP})$$

It is clear from ($2\mathcal{SSP}$) that the original problem is reformulated as a *two-stage stochastic program*. In the first stage, the platform selects the click-through targets $\boldsymbol{\alpha}$ to maximize a variant of the FV, $\mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})$; in the second stage, it selects a static policy π_{static} to meet $\boldsymbol{\alpha}$. Compared with (2), ($2\mathcal{SSP}$) has a more stringent Constraint (3) given that $\Pi_{\mathsf{static}}\subset\Pi$. In other words, ($2\mathcal{SSP}$) provides a lower bound of (2). However, we demonstrate their equivalence in the following lemma.

Lemma 1. *A click-through target vector* $\boldsymbol{\alpha}$ *is feasible to* ($2\mathcal{SSP}$) *if and only if it is feasible to* (2). *Furthermore, any optimal vector* $\boldsymbol{\alpha}$ *of* ($2\mathcal{SSP}$) *is also optimal for* (2), *and vice versa.*

Lemma 1 suggests that to find the optimal click-through targets of (2), it suffices to solve ($2\mathcal{SSP}$). Indeed, we show in Section 5 that there is an algorithm based on the solution to (2), that is, the DWO policy, and a randomized static policy for ($2\mathcal{SSP}$), that is, the Fluid policy, achieving the optimal FV of ($\mathcal{OP}$) in the asymptotic regime where the problem size scales to infinity. This helps us establish the asymptotic equivalence of these two reformulations and the original ad-allocation problem.

We call a click-through target vector $\boldsymbol{\alpha}$ *single-period feasible* if there exists a static policy $\pi_{\mathsf{static}}\in\Pi_{\mathsf{static}}$ such that (3) holds. The single-period feasibility condition for a click-through target vector $\boldsymbol{\alpha}$ is central to the design and analysis of our algorithm to solving both ($2\mathcal{SSP}$) and, eventually, ($\mathcal{OP}$). The rest of this section will be devoted to characterizing the necessary and sufficient condition for an $\boldsymbol{\alpha}$ to be single-period feasible.

Sometimes it is more convenient to rewrite this expected click-through target condition (3) as a periodic-review infinite-horizon sample average-feasibility condition, which will prove useful to establish the optimal dynamic ad-allocation policy, that is, to find a (randomized) policy π such that

$$\liminf_{T\uparrow+\infty}\frac{1}{T}\sum_{t=1}^{T} y_i^j(t|\pi) \geq \alpha_i^j, \text{ for each } i\in\mathcal{N}, j\in\mathcal{M}. \qquad (4)$$

Note that a similar periodic-review reformulation of service-level constraints has also been adopted in the literature on resource allocation (e.g., Zhong et al. 2017, Jiang et al. 2023).

4.2. Necessary and Sufficient Condition for Single-Period Feasibility

To obtain the optimal click-through targets that solve ($2\mathcal{SSP}$), we first characterize the necessary and sufficient condition under which the first-stage click-through target vector $\boldsymbol{\alpha}$ is single-period feasible; that is, (3) holds. We consider the following formulation with a constant objective function:

$$\max_{\pi\in\Pi_{\mathsf{static}}} 0$$
$$\text{s.t. } \mathbb{E}[y_i^j(t|\pi)] \geq \alpha_i^j, \text{ for each } i\in\mathcal{N}, j\in\mathcal{M} \qquad (5)$$

Note that because of stationarity, (5) is regardless of time t. We now characterize when (5) has a feasible solution. We first reformulate (5) as a linear program. Note that the set of deterministic static policies Π_{d} are all the mappings that take a type-j customer to an offer-set in $\mathfrak{S}^j$, which is finite with cardinality $|\Pi_{\mathsf{d}}| = \prod_{j\in\mathcal{M}}|\mathfrak{S}^j|$. Hence, a randomized static policy π_{static} is defined by a probability measure $\mu(\cdot)$ on the finite set Π_{d}, which is essentially a probability simplex in the space $\mathbb{R}^{|\Pi_{\mathsf{d}}|}$.

Under a deterministic static policy $\pi\in\Pi_{\mathsf{d}}$, if a type-j customer arrives, the platform displays an offer-set $S=\pi(j)$ (because of stationarity, we drop the time index t). Thus, the average per-customer-impression number of click-throughs for ad i by type-j customers is given by $p^j\phi_i^j(\pi(j))$. Therefore, (5) can be reformulated as the following LP, the solution to which we denote as $\mu^*(\cdot)$:

$$\max_{\mu(\cdot)\geq 0} 0$$
$$\text{s.t.} \sum_{\pi\in\Pi_{\mathsf{d}}}\mu(\pi)p^j\phi_i^j(\pi(j)) \geq \alpha_i^j, \text{ for each } i\in\mathcal{N} \text{ and } j\in\mathcal{M}$$
$$\sum_{\pi\in\Pi_{\mathsf{d}}}\mu(\pi) = 1. \qquad (6)$$

Taking the dual of the LP (6), we obtain that

$$\min_{\theta_0,\theta_i^j\geq 0}\left\{\theta_0 - \sum_{i\in\mathcal{N},j\in\mathcal{M}}\alpha_i^j\theta_i^j\right\}$$
$$\text{s.t.} \sum_{i\in\mathcal{N},j\in\mathcal{M}} p^j\phi_i^j(\pi(j))\theta_i^j - \theta_0 \leq 0, \text{ for all } \pi\in\Pi_{\mathsf{d}}. \qquad (7)$$

Note that, in (7), $\theta_i^j\geq 0$ is the shadow price for the click-through target α_i^j, whereas θ_0 is the dual variable for the normalization constraint $\sum_{\pi\in\Pi_{\mathsf{d}}}\mu(\pi)=1$. We also define $\boldsymbol{\theta} := (\theta_i^j : i\in\mathcal{N}, j\in\mathcal{M})$.

We note that the objective function of the primal formulation (6) is a constant zero, and there exists a feasible solution $\theta_0 = 0$ and $\theta_i^j = 0$ (for all i and j) to the dual formulation (7) with an objective value equal to zero. By strong duality, (6) has a feasible solution if and only if the optimal objective function value of (7) is nonnegative. By (7), the minimal objective function value of (7) can be obtained at the smallest feasible θ_0, that is, $\max_{\pi\in\Pi_\mathsf{d}}\sum_{i\in\mathcal{N},j\in\mathcal{M}} p^j\phi_i^j(\pi(j))\theta_i^j$ based on the first set of constraints in (7). Combining the aforementioned two observations, (6) is feasible if and only if

$$\min_{\theta_i^j\geq 0}\left\{\max_{\pi\in\Pi_\mathsf{d}}\sum_{i\in\mathcal{N},j\in\mathcal{M}} p^j\phi_i^j(\pi(j))\theta_i^j - \sum_{i\in\mathcal{N},j\in\mathcal{M}}\alpha_i^j\theta_i^j\right\}\geq 0,$$

which is equivalent to

$$\max_{\pi\in\Pi_\mathsf{d}}\sum_{i\in\mathcal{N},j\in\mathcal{M}} p^j\phi_i^j(\pi(j))\theta_i^j \geq \sum_{i\in\mathcal{N},j\in\mathcal{M}}\alpha_i^j\theta_i^j \quad \text{for all } \theta_i^j\geq 0,\ i\in\mathcal{N}, j\in\mathcal{M}. \tag{8}$$

The following theorem summarizes the argument, and it establishes the necessary and sufficient condition for the click-through target vector $\boldsymbol{\alpha}$.

Theorem 1. (Necessary and Sufficient Condition). *A click-through target vector $\boldsymbol{\alpha}$ is single-period feasible; that is, there exists a static policy π_{static} such that* (3) *holds if and only if* (8) *holds.*

Indeed, when $\boldsymbol{\alpha}$ satisfies (8), an optimal dual-vector $\boldsymbol{\theta}^*$ that solves (7) helps characterize the set of *deterministic* static policies over which a primal policy $\mu^*(\cdot)$ (feasible to (6)) randomizes. Specifically, strong duality and the complementary slackness condition imply that for a deterministic policy $\pi\in\Pi_\mathsf{d}$ to have a positive weight in a feasible primal policy $\mu^*(\cdot)$, that is, $\mu^*(\pi)>0$, it must hold that the first constraint of Dual Problem (7) is binding for π, that is,

$$\pi(j)\in\arg\max_{S\in\mathfrak{S}^j}\sum_{i\in S}p^j\theta_i^j\phi_i^j(S) = \arg\max_{S\in\mathfrak{S}^j} p^j\sum_{i\in S}\theta_i^j\phi_i^j(S) = \arg\max_{S\in\mathfrak{S}^j}\sum_{i\in S}\theta_i^j\phi_i^j(S).$$

Note that the left-hand side of Inequality (8) can be viewed as a personalized offer-set optimization problem. Specifically, for each customer type j, we seek to provide an offer-set that maximizes the total revenue from this customer type with the customer's per-click revenue of ad i set to θ_i^j; that is,

$$S^*(\boldsymbol{\theta}|j) = \arg\max_{S\in\mathfrak{S}^j}\sum_{i\in S}\theta_i^j\phi_i^j(S) \tag{9}$$

For tied solutions, an arbitrary offer-set that solves (9) with the smallest cardinality is displayed to the type-j customer so that $S^*(\boldsymbol{\theta}|j)$ is uniquely determined for a given $\boldsymbol{\theta}$. Given a vector $\boldsymbol{\theta}$, we denote the deterministic policy generated by (9) as $\pi_{\boldsymbol{\theta}}$ (hence, $\pi_{\boldsymbol{\theta}}(j) = S^*(\boldsymbol{\theta}|j)$).

We define $g(\boldsymbol{\theta}) := \sum_{i\in\mathcal{N},j\in\mathcal{M}} p^j\theta_i^j\phi_i^j(S^*(\boldsymbol{\theta}|j))$, which is the left-hand side of (8). Hence, we obtain an equivalent necessary and sufficient condition for the feasibility of click-through targets $\boldsymbol{\alpha}$:

$$h(\boldsymbol{\alpha})\geq 0$$
$$\text{where } h(\boldsymbol{\alpha}) := \min_{\boldsymbol{\theta}\geq\mathbf{0}}\left\{g(\boldsymbol{\theta}) - \sum_{i\in\mathcal{N},j\in\mathcal{M}}\alpha_i^j\theta_i^j : \theta_i^j\geq 0, \text{ for each } i\in\mathcal{N}, j\in\mathcal{M}\right\}. \tag{10}$$

Because $g(\boldsymbol{\theta})$ is the maximum of a family of linear functions, it is jointly convex in $\boldsymbol{\theta}$. Therefore, checking the feasibility of the two-stage stochastic program ($2\mathcal{SSP}$) is reduced to minimizing a convex function $g(\boldsymbol{\theta}) - \sum_{i\in\mathcal{N},j\in\mathcal{M}}\alpha_i^j\theta_i^j$ over the quadrant $\{\theta_i^j\geq 0 : i\in\mathcal{N}, j\in\mathcal{M}\}$. Hence, as long as Personalized Offer-Set Optimization Problem (9) is tractable (i.e., the customer click-throughs follow independent, MNL, nested-MNL, or generalized attraction choice models), one could numerically check the feasibility of the click-through targets $\boldsymbol{\alpha}$. By (10), $h(\boldsymbol{\alpha})$ is the minimum of a family of linear functions (in $\boldsymbol{\alpha}$), so it is jointly concave in $\boldsymbol{\alpha}$.

With the characterization of the necessary and sufficient Condition (10) for the feasibility of click-through targets $\boldsymbol{\alpha}$ in the second stage, we are now ready to reformulate ($2\mathcal{SSP}$) as the following single-stage (*deterministic*) convex program to obtain the optimal target vector:

$$\begin{aligned}
&\max_{\boldsymbol{\alpha}\geq\mathbf{0}} \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})\\
&\text{s.t. } h(\boldsymbol{\alpha})\geq 0,\\
&\quad b_i\sum_{j\in\mathcal{M}}\alpha_i^j \leq \frac{B_i}{T}, \text{ for each } i\in\mathcal{N},\\
&\quad \sum_{j\in\mathcal{C}}\alpha_i^j \geq \frac{\eta_i^{\mathcal{C}}}{T}, \text{ for each } i\in\mathcal{N} \text{ and } \mathcal{C}\in\mathfrak{K}_i.
\end{aligned} \tag{$\mathcal{OTP}$}$$

Of particular importance is a special case of ($\mathcal{OTP}$), where customer click-throughs follow the MNL choice model; that is, for any $i\in S$,

$$\phi_i^j(S) = \frac{v_i^j}{1+\sum_{i'\in S}v_{i'}^j}, \tag{11}$$

where $v_i^j>0$ is the attractiveness of ad i to type-j customers. We demonstrate in the following proposition that if customer click-throughs follow the MNL Choice Model (11) with the cardinality constraint for any offer-set displayed to a customer (i.e., $|S(t)| \le K$ for some K), Optimal Target Problem ($\mathcal{OTP}$) can be simplified to a compact convex program with a few linear constraints.[6]

Proposition 1. *If customer click-throughs follow the MNL choice model (11) and the set of all feasible offer-sets is $\mathfrak{S}^j = \{S \subset \mathcal{N} : |S| \le K\}$ for each $j \in \mathcal{M}$, the first-stage convex program ($\mathcal{OTP}$) can be simplified to the following one:*

$$\max_{\boldsymbol{\alpha} \ge \mathbf{0}} \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})$$

$$s.t. \sum_{i' \in \mathcal{N}} \alpha^j_{i'} + \frac{\alpha^j_i}{v^j_i} \le p^j, \text{ for each } i \in \mathcal{N}, j \in \mathcal{M},$$

$$\sum_{i \in \mathcal{N}} \alpha^j_i + \frac{1}{K} \sum_{i \in \mathcal{N}} \frac{\alpha^j_i}{v^j_i} \le p^j, \text{ for each } j \in \mathcal{M},$$

$$b_i \sum_{j \in \mathcal{M}} \alpha^j_i \le \frac{B_i}{T}, \text{ for each } i \in \mathcal{N},$$

$$\sum_{j \in \mathcal{C}} \alpha^j_i \ge \frac{\eta^{\mathcal{C}}_i}{T}, \text{ for each } i \in \mathcal{N} \text{ and } \mathcal{C} \in \mathfrak{K}_i. \qquad (\mathcal{OTP} - \mathcal{MNL})$$

Proposition 1 shows that the number of linear constraints for the convex program ($\mathcal{OTP} - \mathcal{MNL}$) is $\mathcal{O}(mn)$ (instead of exponential in m and n), which ensures its tractability.

Definition 1. We say that a vector $\boldsymbol{\alpha} \in \mathbb{R}^{nm}_{+}$ is *feasible* if it is a feasible solution to ($\mathcal{OTP}$).

By definition, if $\boldsymbol{\alpha}$ is *feasible*, then it is *single-period feasible*. Throughout this paper, we assume the *feasible* region is nonempty, so an optimal solution to ($\mathcal{OTP}$) exists, which we denote as $\boldsymbol{\alpha}^*$. Also, we denote $\mathcal{V}^*_{\mathsf{CT}} := \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha}^*)$ as the optimal objective function value of ($\mathcal{OTP}$). Thus, $\boldsymbol{\alpha}^*$ is the "optimal" click-through target vector for our reformulated ad-allocation problem. According to Theorem 1, $h(\boldsymbol{\alpha})$, defined by (10) as being nonnegative, provides a necessary and sufficient condition for the click-through targets, $\boldsymbol{\alpha}$, to be obtainable in the expected sense $\mathcal{V}^*_{\mathsf{CT}}$, which proves to be an upper bound of the optimal FV for the original problem, $\mathcal{V}^*$ (see Theorem 3). Convex Program Formulation ($\mathcal{OTP}$), therefore, characterizes the optimal click-through target vector $\boldsymbol{\alpha}^*$ and the associated optimal (relaxed) per-customer-impression FV in the expected sense. However, the following two critical questions remain to be addressed.

- **Achieving $\boldsymbol{\alpha}^*$:** How should we display the offer-sets upon the arrival of each customer to achieve the optimal click-through targets $\boldsymbol{\alpha}^*$?
- **Optimality of achieving $\boldsymbol{\alpha}^*$:** Will the offer-set display strategy achieving $\boldsymbol{\alpha}^*$ suffice to obtain the true (unrelaxed) optimal value of Original Problem ($\mathcal{OP}$), that is, $\mathcal{V}^*$?

5. Algorithms for Advertisement Allocation Optimization

In this section, we develop an offer-set allocation algorithm under the TTD framework to address Ad-Allocation Problem ($\mathcal{OP}$) based on the solution to the click-through target model, $\boldsymbol{\alpha}^*$. Specifically, we propose an adaptive offer-set policy that meets the optimal click-through targets $\boldsymbol{\alpha}^*$, and we demonstrate that our proposed algorithm is asymptotically optimal as the problem size scales to infinity. If only a compromised solution can be obtained for the optimal target problem ($\mathcal{OTP}$), the algorithm will achieve the same (asymptotic) optimality gap as that in the optimal target problem, suggesting the robustness of our approach.

5.1. Debt-Weighted Advertisement Allocation Policy

By our two-stage stochastic program (re)formulation of the ad-allocation problem, ($2\mathcal{SSP}$), once we solve the optimal click-through target vector $\boldsymbol{\alpha}^*$, the problem is reduced to devising a randomized offer-set algorithm to achieve $\boldsymbol{\alpha}^*$. To this end, one may solve Primal-Dual Problems (5) and (7) with $\boldsymbol{\alpha} = \boldsymbol{\alpha}^*$ to obtain a feasible randomized policy that achieves $\boldsymbol{\alpha}^*$. This approach, though intuitive, may be computationally prohibitive because Primal LP (6) has $\mathcal{O}(m2^n)$ decision variables and $\mathcal{O}(mn)$ constraints. Therefore, we resort to a data-driven algorithm to generate the random dual-vector $\boldsymbol{\theta}(t)$ upon the arrival of each customer t, based on which we adaptively customize the appropriate ad offer-set $S^*(\boldsymbol{\theta}(t)|j(t))$. Algorithm 1 presents our policy. We refer to the DWO policy (Algorithm 1) initialized with the click-through target vector $\boldsymbol{\alpha}$ as the DWO-$\boldsymbol{\alpha}$ policy, denoted by $\pi_{\mathsf{DWO}}(\boldsymbol{\alpha})$. Of particular importance is the DWO-$\boldsymbol{\alpha}^*$ policy, where the platform solves Optimal Target Problem ($\mathcal{OTP}$) *offline* to obtain the optimal click-through target vector $\boldsymbol{\alpha}^*$ and then implements $\pi_{\mathsf{DWO}}(\boldsymbol{\alpha}^*)$ *online* to adaptively display a personalized offer-set to each customer.

Algorithm 1 (Debt-Weighted Offer-Set Policy $\pi_{\mathsf{DWO}}(\boldsymbol{\alpha})$)

Initialize: The click-through target vector $\boldsymbol{\alpha}$ and the initial debts $d^j_i(1) \leftarrow 0$ for all $i \in \mathcal{N}$ and $j \in \mathcal{M}$.

For each customer $t = 1, 2, \cdots T$**:**

1: Observe the customer type $j(t)$.

2: Display the offer-set

$$S^*(\boldsymbol{d}(t)|j(t)) := \arg\max_{S \in \mathfrak{S}^j} \sum_{i \in S} (d^{j(t)}_i(t))^+ \phi^{j(t)}_i(S) \quad (12)$$

to customer t, where $\boldsymbol{d}(t) = (d^j_i(t) : i \in \mathcal{N}, j \in \mathcal{M})$ is the realized debt vector upon the arrival of customer t. For tied solutions, an arbitrary offer-set that solves (12) with the smallest cardinality is displayed to the type-j customer.

3: Observe the customer click-throughs $(y_i^j(t) : i \in \mathcal{N}, j \in \mathcal{M})$ and collect the advertising value $\sum_{i\in\mathcal{N}} r_i^{j(t)} y_i^{j(t)}$. Remove any offer-set containing ad i with $(\sum_j \sum_{\tau \le t} y_i^j(\tau)) b_i \ge B_i$ (i.e., the budget of ad i has already been depleted) from $\mathfrak{S}^j$ for all j hereafter.
4: Update the debt $d_i^j(t+1) \leftarrow d_i^j(t) + \alpha_i^j - y_i^j(t)$ for all $i \in \mathcal{N}$ and $j \in \mathcal{M}$.

A few remarks are in order with respect to Algorithm 1. First, the DWO policy displays the offer-set to each customer based on Offer-Set Optimization Problem (9). This standard personalized offer-set optimization problem is tractable so the offer-set $S^*(\boldsymbol{d}(t)|j(t))$ can be efficiently obtained for a broad class of choice models: independent, MNL, nested MNL, and generalized attraction. For example, Feldman et al. (2022) demonstrate that, for the MNL choice model, the single-period assortment optimization problem can be solved with the running time $\mathcal{O}(n^2)$ (where n is the number of products) and have successfully deployed their algorithm on one of Alibaba's large-scale online retailing platforms (Tmall).

Second, we call Algorithm 1 the DWO policy because the offer-set optimization is weighted by the "debt" of each customer-advertisement pair for customers $\{1, 2, \ldots, t-1\}$. Note that $(t-1)\alpha_i^j$ is the total click-through target of ad i by type-j customers until the start of time t, whereas $\sum_{\tau=1}^{t-1} y_i^j(\tau)$ is the total realized click-throughs by then. Therefore, $(d_i^j(t))^+ = \max((t-1)\alpha_i^j - \sum_{\tau=1}^{t-1} y_i^j(\tau), 0)$ is the total "debt" owed by the platform to the click-through target associated with ad i and customer type j when deciding the offer-set displayed to customer t. The debts $\boldsymbol{d}(t)$ only depend on $\mathcal{H}_{t-1}$ and are independent of any information revealed on or after time t. For a feasible click-through target vector $\boldsymbol{\alpha}$, we can also view the debt process $\{\boldsymbol{d}(t) : t \ge 1\}$ as a data-driven adaptive way to generate the random dual-vector $\boldsymbol{\theta}$, which prescribes a feasible randomized policy $\pi = \pi_{\boldsymbol{\theta}}$.

Finally, for tied solutions, an arbitrary offer-set that solves (12) with the smallest cardinality is displayed to the type-j customer so that $S^*(\boldsymbol{d}(t)|j(t))$ is uniquely defined. Hence, any ad i with $d_i^j(t) \le 0$ will not be offered to customer t with type j.

5.2. Asymptotic Analysis

In this subsection, we will establish that the DWO-$\boldsymbol{\alpha}^*$ policy can achieve the optimal FV for Original Ad-Allocation Problem ($\mathcal{OP}$) asymptotically. Before demonstrating the optimality of the DWO-$\boldsymbol{\alpha}^*$ policy, we first introduce the asymptotic regime where the problem size scales up to infinity. Specifically, we denote a family of ad-allocation problems with the budget for each ad i, $B_i(\gamma) := B_i\gamma$, the click-through requirement for ad i and customer-type set $\mathcal{C} \in \mathfrak{K}_i$, $\eta_i^{\mathcal{C}}(\gamma) = \eta_i^{\mathcal{C}}\gamma$, and the planning horizon length $T(\gamma) := T\gamma$, as $\mathcal{OP}(\gamma)$, where $\gamma > 0$ is a scaling parameter of problem size. Hence, Original Problem ($\mathcal{OP}$) is equivalent to $\mathcal{OP}(1)$. For the problem $\mathcal{OP}(\gamma)$ and a policy $\pi \in \Pi$, we denote $\mathcal{E}(\pi|\gamma)$ as the expected efficiency, $\mathcal{F}(\pi|\gamma)$ as the expected fairness, and $\mathcal{V}(\pi|\gamma) = \mathcal{E}(\pi|\gamma) + \lambda\mathcal{F}(\pi|\gamma)$ as the expected FV generated by π in $\mathcal{OP}(\gamma)$. Furthermore, $\mathcal{V}^*(\gamma) := \max_{\pi\in\Pi} \mathcal{V}(\pi|\gamma)$ denotes the optimal expected FV for $\mathcal{OP}(\gamma)$. Note that the market-size scaling factor γ does not affect the feasibility of a click-through target vector $\boldsymbol{\alpha}$, nor does it change Two-Stage Stochastic Program Reformulation ($2\mathcal{SSP}$) or Target Problem Reformulation ($\mathcal{OTP}$).

We first establish that for any *feasible* click-through target vector $\boldsymbol{\alpha}$, the DWO-$\boldsymbol{\alpha}$ policy exactly achieves $\boldsymbol{\alpha}$ in $\mathcal{OP}(\gamma)$ as the problem size γ scales to infinity.

Theorem 2. *If $\boldsymbol{\alpha}$ is feasible, that is, all constraints of ($\mathcal{OTP}$) are satisfied, then we have*

$$\lim_{\gamma\uparrow+\infty} \frac{1}{T(\gamma)} \sum_{t=1}^{T(\gamma)} y_i^j(t|\pi_{\mathsf{DWO}}(\boldsymbol{\alpha})) = \alpha_i^j \text{ almost surely for all } i \in \mathcal{N} \text{ and } j \in \mathcal{M}. \tag{13}$$

Theorem 2 is the central technical result of this paper—it is an important stepping stone on the way to proving the asymptotic optimality of the DWO-$\boldsymbol{\alpha}^*$ policy. Interestingly, as long as this policy is initiated with a *feasible* click-through target vector $\boldsymbol{\alpha}$, it will not only achieve click-through levels *at least as high as* these targets (i.e., (4)) but also *exactly approach* them (i.e., (13)). Adopting a coupling argument, Proof of Theorem 2 (see Online Appendix D for details) demonstrates that if the problem size γ scales up to infinity, the DWO-$\boldsymbol{\alpha}$ policy will *not* exhaust the budget of any ad and, thus, will secure the click-through targets $\boldsymbol{\alpha}$. Therefore, under our proposed DWO policy, the click-through requirements $\{\eta_i^{\mathcal{C}}(\gamma) : i \in \mathcal{N}, \mathcal{C} \in \mathfrak{K}_i\}$ can be satisfied almost surely, instead of in expectation, in the asymptotic regime.

Based on Theorem 2, one may conjecture that if the click-through target vector $\boldsymbol{\alpha}$ is optimally chosen (i.e., as the solution to Optimal Target Problem ($\mathcal{OTP}$), $\boldsymbol{\alpha}^*$), the DWO-$\boldsymbol{\alpha}^*$ policy could achieve the optimal FV for the original ad-allocation problem, $\mathcal{V}^*$. The main result of this section is that the following theorem validates this conjecture in the asymptotic regime and quantifies the nonasymptotic optimality gap of the policy.

Theorem 3. *The* DWO-$\boldsymbol{\alpha}^*$ *policy is asymptotically optimal; that is,*

$$\lim_{\gamma\uparrow+\infty} \mathcal{V}(\pi_{\mathsf{DWO}}(\boldsymbol{\alpha}^*)|\gamma) = \lim_{\gamma\uparrow+\infty} \mathcal{V}^*(\gamma) = \mathcal{V}^*_{\mathsf{CT}}. \tag{14}$$

Furthermore, the optimal objective function value of First-Stage Click-Through Target Optimization ($\mathcal{OTP}$) is an upper bound for Original Problem ($\mathcal{OP}$) in the nonasymptotic regime; that is, for any $\gamma > 0$,

$$\mathcal{V}^*_{\mathsf{CT}} \ge \mathcal{V}^*(\gamma), \tag{15}$$

and the optimality gap the DWO-$\boldsymbol{\alpha}^*$ *policy is of order $\mathcal{O}(\gamma^{-\frac{1}{2}})$;*

that is, there exists a constant $\mathcal{C} > 0$, *such that for any* $\gamma > 0$,

$$\mathcal{V}^*(\gamma) - \mathcal{V}(\pi_{\mathsf{DWO}}(\boldsymbol{\alpha}^*)|\gamma) \le \frac{\mathcal{C}}{\sqrt{\gamma}}. \tag{16}$$

Theorem 3 proves that the DWO-$\boldsymbol{\alpha}^*$ policy generated by our TTD framework is asymptotically optimal when the ad budgets, the click-through requirements, and the time-horizon length all scale up to infinity at the same rate. In particular, the optimal expected FV of $\mathcal{OP}(\gamma)$ is identical to the optimal FV of ($\mathcal{OTP}$) asymptotically, and the former is upper bounded by the latter in the nonasymptotic regime. Such equivalence suggests that our reformulation in Section 4.1 is an asymptotically equivalent relaxation of the original problem. Moreover, we show that the DWO-$\boldsymbol{\alpha}^*$ policy can achieve the optimal expected FV at a convergence rate of $\mathcal{O}(\gamma^{-\frac{1}{2}})$. Proof of Theorem 3 relies on a delicate application of Theorem 2, which shows that the DWO-$\boldsymbol{\alpha}^*$ policy generated by our TTD framework achieves the optimal FV of the click-through target optimization (i.e., $\lim_{\gamma\uparrow+\infty} \mathcal{V}(\pi_{\mathsf{DWO}}(\boldsymbol{\alpha}^*)|\gamma) = \mathcal{V}^*_{\mathsf{CT}}$). Another critical step to establish Theorem 3 is to find an asymptotically optimal static policy for the original problem $\mathcal{OP}(\gamma)$, which can be constructed with the auxiliary Fluid convex program ($\mathcal{OP}_{\mathsf{Fluid}}$), that is, Proposition 2. We relegate the detailed discussions to Section 5.3.

5.3. Discussions

Fluid Benchmark. To prove Theorem 3, one needs to construct an asymptotically optimal static policy for Original Problem ($\mathcal{OP}$). To find such a policy, we consider an auxiliary Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) and establish its intrinsic connections and, therefore, asymptotic equivalence to Original Problem ($\mathcal{OP}$). We also note that the convex program is a generalization of the standard CDLP approach (Liu and Van Ryzin 2008). Specifically, the auxiliary Fluid problem is defined as Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) as follows:

$$\max_{z\ge 0} \mathcal{V}_{\mathsf{Fluid}}(z) := \sum_{i\in\mathcal{N}, j\in\mathcal{M}, S\in\mathfrak{S}^j} r_i^j p^j \phi_i^j(S) z^j(S) + \lambda F(\boldsymbol{\zeta})$$

$$\text{s.t.} \sum_{j\in\mathcal{M}, S\in\mathfrak{S}^j} b_i p^j \phi_i^j(S) z^j(S) \le \frac{B_i}{T} \text{ for each } i \in \mathcal{N}$$

$$\sum_{j\in\mathcal{C}, S\in\mathfrak{S}^j} p^j \phi_i^j(S) z^j(S) \ge \frac{\eta_i^{\mathcal{C}}}{T} \text{ for each } i \in \mathcal{N} \text{ and } \mathcal{C} \in \mathfrak{K}_i$$

$$\sum_{S\in\mathfrak{S}^j} z^j(S) \le 1 \text{ for each } j \in \mathcal{M}$$

$$\boldsymbol{\zeta} \in \mathbb{R}^{nm}, \text{ with } \zeta_i^j = \sum_{S\in\mathfrak{S}^j} p^j z^j(S) \phi_i^j(S) \tag{$\mathcal{OP}_{\mathsf{Fluid}}$}$$

We denote the solution to ($\mathcal{OP}_{\mathsf{Fluid}}$) as z^* and the associated optimal objective function value as $\mathcal{V}^*_{\mathsf{Fluid}} = \mathcal{V}_{\mathsf{Fluid}}(z^*)$. It is self-evident from the formulation of ($\mathcal{OP}_{\mathsf{Fluid}}$) that $z^j(S)$ is the probability of displaying offer-set S to a type-j customer upon arrival, whereas $\zeta_i^j = \sum_{S\in\mathfrak{S}^j} p^j z^j(S) \phi_i^j(S)$ is the expected per-customer-impression click-throughs of ad i by type-j customers. A vector $z = (z^j(S) : j \in \mathcal{M}, S \in \mathfrak{S}^j)$ feasible for ($\mathcal{OP}_{\mathsf{Fluid}}$) naturally induces a randomized static policy for Original Problem ($\mathcal{OP}$), which displays offer-set $S \in \mathfrak{S}^j$ to a customer of type $j \in \mathcal{M}$ with probability $z^j(S)$. Whenever at least one ad runs out of budget, the policy offers nothing to each arriving customer afterward. We refer to this policy as the Fluid-z policy, denoted by $\pi_{\mathsf{Fluid}}(z)$. Note that policy $\pi_{\mathsf{Fluid}}(z)$ does not fully utilize the remaining budgets of nondepleted ads. To strengthen the performance of this policy, we will slightly adjust its implementation in our numerical experiments. The detailed discussions of the adjustment are deferred to Section 6.

Indeed, there are intrinsic connections between Optimal Target Problem ($\mathcal{OTP}$) and Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$). We can always construct a feasible (respectively, optimal) click-through target vector in ($\mathcal{OTP}$) from any feasible (respectively, optimal) offer-set assignment probabilities in ($\mathcal{OP}_{\mathsf{Fluid}}$). For any z feasible to ($\mathcal{OP}_{\mathsf{Fluid}}$), we define $\hat{\boldsymbol{\alpha}}(z) \in \mathbb{R}^{nm}$, where $\hat{\alpha}_i^j(z) = \sum_{S\in\mathfrak{S}^j} p^j z^j(S) \phi_i^j(S)$.

Lemma 2. *Assume that* z *is feasible for* ($\mathcal{OP}_{\mathsf{Fluid}}$). *We have that* $\hat{\boldsymbol{\alpha}}(z)$ *is first-stage feasible and can be achieved by policy* $\pi_{\mathsf{Fluid}}(z)$; *that is,* $\mathbb{E}[y_i^j(t|\pi_{\mathsf{Fluid}}(z))] = \hat{\alpha}_i^j(z)$. *Furthermore,* $\mathcal{V}_{\mathsf{CT}}(\hat{\boldsymbol{\alpha}}(z)) = \mathcal{V}_{\mathsf{Fluid}}(z)$. *In particular,* $\hat{\boldsymbol{\alpha}}(z^*)$ *is an optimal solution to* ($\mathcal{OTP}$) *with* $\mathbb{E}[y_i^j(t|\pi_{\mathsf{Fluid}}(z^*))] = \hat{\alpha}_i^j(z^*)$.

We are now ready to demonstrate that as the problem size γ scales to infinity, both the original problem $\mathcal{OP}(\gamma)$ and Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) have the same optimal (expected) per-customer-impression FV, which is also identical to the one generated by the Fluid-z^* policy in $\mathcal{OP}(\gamma)$. Formally, the following proposition establishes these equivalences and implies the asymptotic optimality of a static policy $\pi_{\mathsf{Fluid}}(z^*)$.

Proposition 2. *The following inequalities hold:*

$$\lim_{\gamma\uparrow+\infty} \mathcal{V}^*(\gamma) \ge \lim_{\gamma\uparrow+\infty} \mathcal{V}(\pi_{\mathsf{Fluid}}(z^*)|\gamma) = \mathcal{V}^*_{\mathsf{Fluid}} \ge \lim_{\gamma\uparrow+\infty} \mathcal{V}^*(\gamma). \tag{17}$$

Therefore, all inequalities in (17) *hold as equalities.*

As a stepping stone to prove Theorem 3, Proposition 2 shows that the static Fluid-z^* policy generated by ($\mathcal{OP}_{\mathsf{Fluid}}$) is asymptotically optimal for the original problem $\mathcal{OP}(\gamma)$. Hence, we will also use this policy and its resolving variants as the benchmarks in our numerical experiments to evaluate our proposed DWO-$\boldsymbol{\alpha}^*$

policy in Section 6. Note that the equivalences in (17) generalize Proposition 1 of Liu and Van Ryzin (2008) to our setting with algorithmic fairness, personalized offer-sets, and click-through requirements.

Computational Efficiency. Our proposed DWO-$\boldsymbol{\alpha}^*$ policy generated by our TTD framework involves two steps. The first step solves Optimal Target Problem ($\mathcal{OTP}$) to obtain the optimal click-through target vector $\boldsymbol{\alpha}^*$ offline, and the second step implements the second-stage DWO display procedure online. We now discuss the computational efficiency of the two steps separately, starting from the second-stage online implementation.

Second-Stage Online Implementation. To implement Algorithm 1 online, given any click-through target vector $\boldsymbol{\alpha}$, the bottleneck is to solve Single-Period Offer-Set Optimization Problem (9) upon the arrival of each customer t. Standard results in the assortment-optimization literature suggest that if customers follow a wide range of commonly used choice models—such as the independent choice (Feldman et al. 2022), MNL (Rusmevichientong et al. 2010, Sumida et al. 2021), nested MNL (Davis et al. 2014), and generalized attraction (Luce 2012, Gallego et al. 2015a) choice models—the personalized Offer-Set Optimization (9) can be solved efficiently. Therefore, the second-stage online implementation of the DWO-$\boldsymbol{\alpha}$ is computationally efficient as long as the single-period offer-set optimization is tractable, which is generally the case for choice models commonly used in practice.

First-Stage Convex Program. Section 4.2 and Online Appendix F show that if the customer choices follow the MNL, independent, and generalized attraction choice models, Optimal Target Problem ($\mathcal{OTP}$) can be greatly simplified to a convex program with a few linear constraints, which can be solved efficiently in general. We emphasize that the MNL, independent, and generalized attraction choice models are all widely used in practice. If customers follow a general choice model, Lemma 2 implies that ($\mathcal{OTP}$) shares the same computational complexity as Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$). To see this, note that for any solution to ($\mathcal{OP}_{\mathsf{Fluid}}$), $\boldsymbol{z}^*$, we can construct a click-through target vector $\hat{\boldsymbol{\alpha}}(\boldsymbol{z}^*)$ that is feasible and optimal for ($\mathcal{OTP}$). Therefore, as long as auxiliary Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) is computationally tractable, we can efficiently obtain an optimal solution to ($\mathcal{OTP}$) as well. In general, the DWO-$\boldsymbol{\alpha}^*$ policy solves ($\mathcal{OTP}$) offline only once at the beginning of the planning horizon, which is generally tractable in most applications. Indeed, Table 1 in Section 6 shows that our DWO-$\boldsymbol{\alpha}^*$ policy is much more scalable than the Fluid-based benchmarks commonly adopted in the literature.

Table 1. Comparison of Average Solving Times

Policy	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
DWO	0.0054	0.0067	0.0095	0.0099	0.0064
Fluid	0.0080	0.0918	2.5454	60.1610	Out of memory

Note. Data are given in seconds unless marked otherwise.

If the auxiliary Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) is intractable, obtaining an *optimal* click-through target vector $\boldsymbol{\alpha}^*$ may be prohibitive. However, we can still identify a *feasible* click-through target vector $\boldsymbol{\alpha}$ by applying Theorem 1. Once a feasible $\boldsymbol{\alpha}$ is found, as long as Single-Period Offer-Set Optimization (9) is tractable, the second-stage online implementation of the DWO-$\boldsymbol{\alpha}$ policy should be tractable as well. Furthermore, the DWO-$\boldsymbol{\alpha}$ policy achieves the same asymptotic FV as $\mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})$, as shown in the following proposition.

Proposition 3. *If $\boldsymbol{\alpha}$ is first-stage feasible, that is, it is a feasible solution to ($\mathcal{OTP}$), then we have*

$$\lim_{\gamma\uparrow+\infty} \mathcal{V}(\pi_{\mathsf{DWO}}(\boldsymbol{\alpha})|\gamma) = \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha}). \tag{18}$$

The key implication from Proposition 3 is that in the *asymptotic regime* where the problem size γ scales to infinity, the second-stage online implementation of the DWO-$\boldsymbol{\alpha}$ policy will not incur any additional optimality loss on top of that from a feasible suboptimal click-through target vector $\boldsymbol{\alpha}$ for the first-stage target optimization; that is, $\lim_{\gamma\uparrow+\infty}\mathcal{V}^*(\gamma) - \lim_{\gamma\uparrow+\infty}\mathcal{V}(\pi_{\mathsf{DWO}}(\boldsymbol{\alpha})|\gamma) = \mathcal{V}^*_{\mathsf{CT}} - \mathcal{V}_{\mathsf{CT}}(\boldsymbol{\alpha})$.

Comparison with Existing Algorithms. It is useful to compare the DWO-$\boldsymbol{\alpha}^*$ policy with relevant algorithms in the existing literature.

Debt-weighted resource-allocation algorithms. As discussed earlier, the objective of existing debt-weighted algorithms (e.g., Zhong et al. 2017, Lyu et al. 2019, Jiang et al. 2023) is to allocate a centralized resource to satisfy some *feasible* and *exogenous* service-level constraints. The goal of the DWO-$\boldsymbol{\alpha}^*$ policy, however, is to maximize the FV of an online advertising system so that the click-through target vector, which is the counterpart of the service-level constraints in our setting, is *endogenized* in the first stage of the algorithm. Because we have such a different objective for our policy, we develop a two-stage reformulation to implement the algorithm and take a different path for its analysis, which relies on establishing the (asymptotic) equivalence of different formulations of the problem. Another key difference between our DWO-$\boldsymbol{\alpha}^*$ policy and other debt-weighted resource-allocation algorithms is that whereas those algorithms can freely control the allocation and consumption of the resources, our policy has to handle the additional complexity of customers' stochastic choice behaviors, which introduces another layer of challenge to controlling the debt process. Furthermore, to our best knowledge, we are also the first in the literature to study the dynamic assortment/offer-set optimization problem through the lens of a debt-weighted algorithm. Moreover, we show that our DWO-$\boldsymbol{\alpha}^*$ policy can work for general resource allocation and other ad-allocation problems, such as Ad Display and AdWords (see, e.g., Mehta 2013) in Online Appendix G.

Fluid *convex program heuristics.* In our setting with a nonlinear fairness term, the standard LP-based heuristics with or without resolving (e.g., Liu and Van Ryzin 2008, Jasin and Kumar 2012, Bumpensanti and Wang 2020, Balseiro et al. 2023) applied to linear rewards can be extended to similar heuristics based on Fluid convex programs (e.g., ($\mathcal{OP}_{\mathsf{Fluid}}$)). A core advantage of the DWO-$\boldsymbol{\alpha}^*$ policy over the family of Fluid heuristics is that, by the nature of the algorithm to assign a higher weight to an ad-customer pair with a larger debt, the click-through and, thus, the reward process will follow a mean-reverting pattern. Therefore, compared with the Fluid or Fluid-resolving heuristics, our DWO-$\boldsymbol{\alpha}^*$ algorithm can deplete the budget of each ad more smoothly throughout the horizon, which is highly desirable for the advertising business in practice.

For example, Google recommends a "standard" ad-delivery scheme for its advertisers, especially those with a low budget, to avoid exhausting their budgets early.[7] Under this standard delivery scheme, each advertisement can reach customers evenly throughout the day. Furthermore, with the cardinality constraint on the displayed offer-sets, one difficulty of using Fluid or its resolving variants is that the number of variables (i.e., the probability of each offer-set for all customer types) quickly explodes as the number of products increases, even when the choice model is restricted to MNL. The DWO-$\boldsymbol{\alpha}^*$ policy has a better scalability than those Fluid-based heuristics. These advantages of our policy are also confirmed by our numerical comparisons in Section 6.

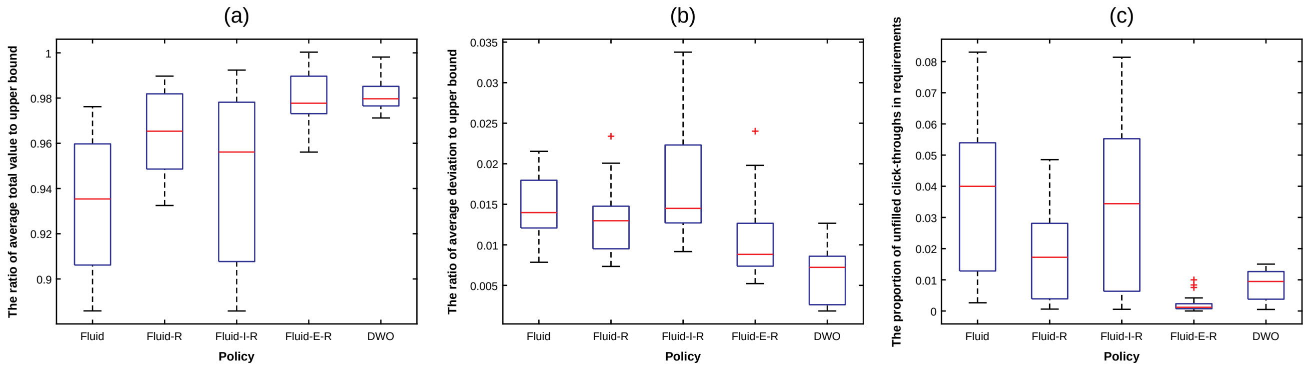
Inventory-balancing policy. Inventory balancing is another family of algorithms to address the personalized-assortment optimization problem with inventory constraints (see, e.g., Golrezaei et al. 2014). This policy uses the remaining inventory to reweight the value of each product. The inventory-balancing policy is difficult, if not impossible, to adapt to our setting because, on one hand, it is challenging for this policy to handle the click-through requirements $\{\eta_i^{\mathcal{C}} : i \in \mathcal{N}, \mathcal{C} \in \mathfrak{K}_i\}$, and on the other hand, it is hard to incorporate the nonlinear fairness metric into the inventory-balanced offer-set optimization problem upon the arrival of each customer. Our DWO-$\boldsymbol{\alpha}^*$ policy circumvents these two challenges under our two-stage framework within which the personalized offer-set optimization problem is reduced to a standard, single-period problem with a reward linear in the number of click-throughs (9). For completeness, in Online Appendix L, we numerically compare our DWO-$\boldsymbol{\alpha}^*$ policy with the inventory-balancing benchmark in the setting without click-through requirements (i.e., $\mathfrak{K}_i = \emptyset$ for all i) and fairness concerns (i.e., $\lambda = 0$). The numerical results demonstrate that our policy outperforms the inventory-balancing benchmark for all the problem instances examined when the demand-to-supply ratio is not too large. When the demand-to-supply ratio is large, our policy performs fairly well, achieving an average of more than 99% of the theoretical upper bound in all problem instances.

6. Numerical Experiments

In this section, we numerically evaluate our DWO-$\boldsymbol{\alpha}^*$ policy (simply DWO in this section) generated by our TTD framework for ad-allocation optimization, benchmarked against four Fluid-based heuristics. The first benchmark is the static policy induced by the optimal solution $\boldsymbol{z}^*$ to ($\mathcal{OP}_{\mathsf{Fluid}}$) (see also, Liu and Van Ryzin 2008), denoted as the *fluid-approximation policy* or the Fluid-$\boldsymbol{z}^*$ policy.[8] In each period, we randomly display offer-set S to a customer of type j with probability $z^{j*}(S)$, which is a solution to Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$). Once the budget of an ad is depleted, it is automatically deleted in any offer-set generated by the Fluid policy. In this case, all the remaining ads in the offer-set will continue to be displayed. This adjustment of the Fluid policy is to enhance budget utilization and, consequently, the performance of the policy. The second benchmark is a resolving variant of the Fluid policy, denoted as the Fluid *resolving policy* or the Fluid-R policy, which resolves the Fluid convex program at evenly spaced time epochs based on the remaining budgets and click-through requirements (see, e.g., Jasin and Kumar 2012). The third benchmark is a refined version of the Fluid-R policy, denoted as the Fluid *infrequent resolving policy* or the Fluid-I-R policy, under which the resolving time epochs are more carefully designed and are infrequent/sparse at the beginning of the ad campaign (see, e.g., Bumpensanti and Wang 2020). Finally, the fourth benchmark resolves the Fluid convex program at every period (see, e.g., Balseiro et al. 2023), denoted as the Fluid *every-period resolving policy* or the Fluid-E-R policy. See Online Appendix H for the implementation details of the Fluid-R, Fluid-I-R, and Fluid-E-R policies. The settings of our numerical studies and parameters varied for generating problem instances, concentration parameter (*CP*) and loading factor (*LF*), are introduced in Online Appendix I.[9]

We generate 30 sample paths for each problem instance to evaluate the following performance metrics of interest: (1) the ratio between the expected FV and its theoretical upper bound characterized by the solution to the first-stage convex program, $\mathcal{V}^*_{\mathsf{CT}}$, (2) the ratio between the standard deviation of FV and $\mathcal{V}^*_{\mathsf{CT}}$, and (3) the average proportion of unfilled click-through

Figure 2. (Color online) Comparison Between DWO-Based and Fluid-Based Policies



Notes. (a) Total fairness-adjusted value. (b) Standard deviation of FV. (c) Unfilled click-throughs.

requirements compared with $\eta_i^{\{j\}}$. We use the relative ratios (instead of the absolute values) to make the comparisons clear.

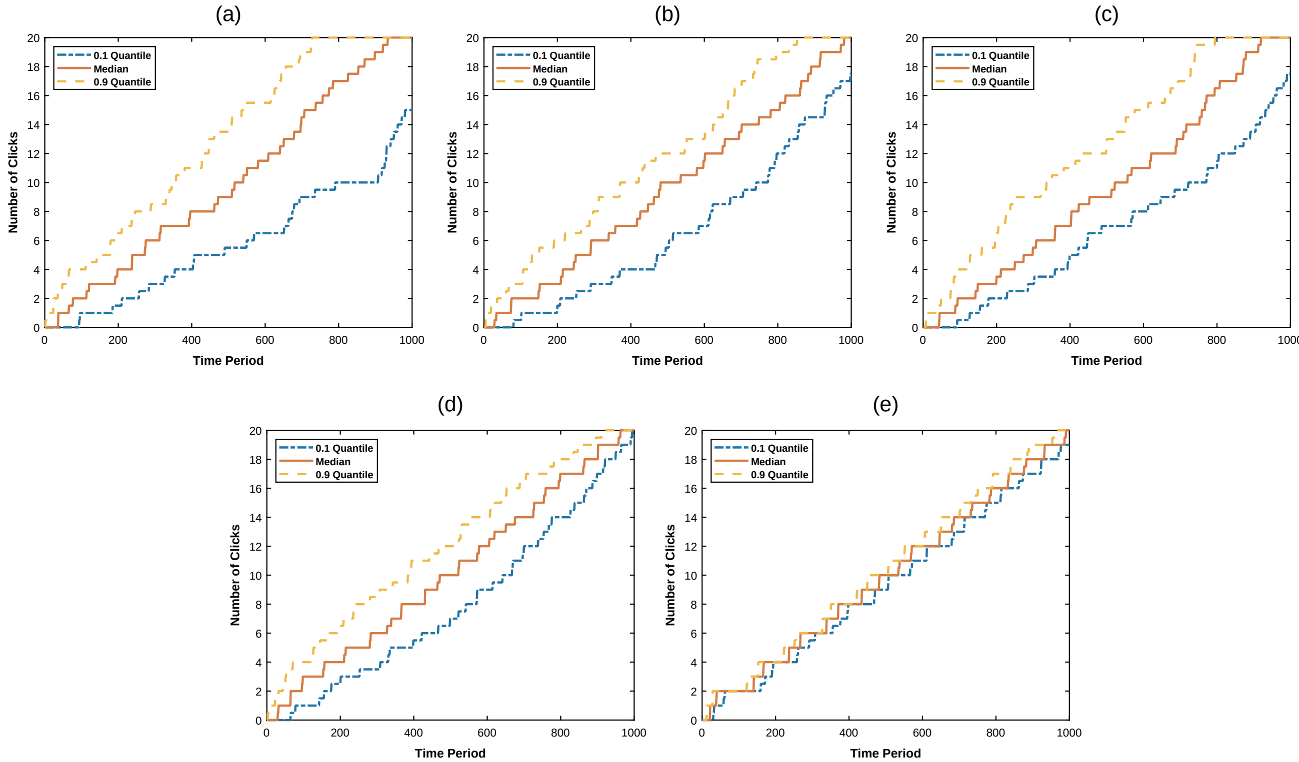
We report the numerical findings as box plots with respect to all problem instances in Figure 2, which clearly illustrates the advantages of our algorithm over the benchmarks in various dimensions. Figure 2(a) demonstrates that our DWO algorithm consistently outperforms all the Fluid-based benchmarks by delivering higher values in the total objective. Figure 2(b) shows that the variability of FV is much lower under our policy than it is under the benchmarks. Finally, Figure 2(c) shows that the DWO algorithm significantly reduces the proportion of unfilled click-through requirements compared with Fluid, Fluid-R, and Fluid-I-R policies but has more unfilled click-throughs than Fluid-E-R. In short, our proposed DWO algorithm not only generates higher FV than the Fluid-based benchmarks do but also reduces the variability of FV. Note that the resolving policies can achieve an optimality gap of order $\mathcal{O}(\gamma^{-1})$ under certain regularity conditions (see, e.g., Balseiro et al. 2023), whereas our DWO policy has a larger provable optimality gap of order $\mathcal{O}(\gamma^{-\frac{1}{2}})$. However, our DWO algorithm outperforms all the Fluid-based benchmarks in numerical results. We propose providing more in-depth analyses of the DWO policy in future research.

To understand why our DWO algorithm enjoys the great performance illustrated in Figure 2, we also plot the 0.1, 0.5 (i.e., median), and 0.9 quantiles of the click-through sample paths of an ad (see details in Online Appendix I) for the five approaches (under the problem instance $LF = 1$ and $CP = 100$) we studied in Figure 3. Our numerical experiments make clear that although all five policies deplete the ad's entire budget for more than 50% of sample paths, the variability of the click-through sample paths (equivalently, the budget-depleting process) through the entire time horizons under the Fluid, Fluid-R, Fluid-I-R, and Fluid-E-R algorithms are much higher than our DWO policy. Furthermore, the Fluid-based approaches all run out of budget long before the end of the ad campaign, whereas our DWO policy exhausts the budget only toward the very end.

We highlight that such smooth budget depletion of our proposed algorithm should be credited to their mean-reverting pattern driven by the fact that the offer-set displayed in each period is prescribed in accordance with the "debts" owed by the algorithm to the optimal click-through targets. In particular, the ads farther from (respectively, closer to) their optimal targets will receive higher (respectively, lower) weights when the algorithm is deciding which assortment to display upon the arrival of each user. Thus, such intertemporal pooling leads to the mean-reverting phenomenon of our proposed approach. Note that the Fluid-based policies also exhibit certain mean-reverting properties weaker than our DWO policy.[10] Smooth budget depletion is, in practice, a highly desirable property for advertisers that use online advertising platforms—Facebook has even built some API tools that help its clients pace their ad delivery and smooth their budget depletion.[11] Therefore, from a practical perspective, our DWO algorithm may, appealingly, help advertisers and advertising platforms achieve smoother budget depletion.

In addition to obtaining better performance in most of the cases we examine and much smoother depletion of ad budgets, our DWO policy is more scalable and efficient in both time and space complexities. We carried out our numerical studies by varying the offer-set size constraint K from two to five and generating 30 samples randomly for each K (under the problem instance $LF = 1$ and $CP = 100$) with other model primitives identical to those of the experiments in Section 6. We conducted the experiment by using Gurobi 10.0.0 within MATLAB R2022b on a 2.10 GHz Intel Core i7-1260P CPU with 32 GB of RAM. Table 1 shows that the average computation time of finding optimal click-through targets (i.e., solving Convex Program ($\mathcal{OTP}$)) is approximately 0.01 seconds regardless of the value of K, but solving Fluid Convex Program ($\mathcal{OP}_{\mathsf{Fluid}}$) is much

Figure 3. (Color online) The 0.1 Quantiles and Medians and 0.9 Quantiles in 30 Sample Paths over Time of Click-Numbers of an Ad with $LF = 1$, and $CP = 100$



Notes. (a) Fluid. (b) Fluid-R. (c) Fluid-I-R. (d) Fluid-E-R. (e) DWO.

more time-consuming (Fluid-R, Fluid-I-R, and Fluid-E-R are, of course, even slower). In addition, increasing K means exponentially more possible offer-sets for the Fluid, Fluid-R, Fluid-I-R, and Fluid-E-R policies, so much more memory and computational time are needed in this case. Table 1 shows that the case of $K = 5$ may even incur an "out of memory" error for the Fluid benchmark. In short, our algorithms enjoy higher scalability than the Fluid-based benchmarks. As a final remark, our numerical results also reveal that introducing the fairness term in the objective function could substantially reduce the algorithmic bias without compromising the advertising efficiency much (see Online Appendix K for details).

7. Conclusion

The allocation of customer traffic to different ads is a crucial operations decision for online e-commerce platforms to optimize their advertising business. The emerging advocacy for algorithmic fairness of online ad delivery has posed additional challenges for the design of ad-allocation policies. In this paper, we propose a TTD framework comprising a general model and an associated efficient algorithm to study optimal ad allocation under customer choices and algorithmic fairness. Although the original online ad-allocation problem is intractable, we develop an asymptotically equivalent two-stage stochastic program as a surrogate. Furthermore, we propose a simple but effective algorithm—the DWO-$\boldsymbol{\alpha}^*$ policy—which is provably optimal for achieving the maximum FV from advertising in the asymptotic regime. Furthermore, the proposed algorithm gives rise to the mean-reverting pattern of the budget consumption process and therefore achieves smoother budget depletion, which is highly desirable from a practical perspective. Our algorithm also helps substantially improve the fairness of ad allocation for a platform without compromising its efficiency much.

Acknowledgements

The authors thank Professor Jeannette Song, the associate editor, and reviewers for their constructive suggestions. The authors also thank Chung-Piaw Teo and Jiawei Zhang for their valuable feedback on this work. The authors are listed in alphabetical order, and the corresponding authors are Ying Rong, Renyu Zhang, and Huan Zheng.

Endnotes

[1] Interactive Advertising Bureau report. See https://www.iab.com/insights/internet-advertising-revenue-fy2019-q12020/.

[2] For an explanation, see https://advertising.amazon.com/solutions/products/sponsored-products.

[3] See Amazon's 2019 financial report at https://www.sec.gov/ix?doc=/Archives/edgar/data/1018724/000101872420000004/amzn-20191231x10k.htm.

[4] See https://www.facebook.com/business/help/397103717129942?id=1913105122334058.

[5] For ease of exposition, we consider the same bid price of advertiser i across all customer types (i.e., b_i). In practice, an advertiser may set different bids for different targeted groups. If we allow viewer-type-dependent bid price b_i^j, our analysis and results will not be affected.

[6] In Online Appendix E, we refer to additional insights on the feasible region of the click-through targets. In Online Appendix F, we show that if customer click-throughs follow the independent choice model (which is widely adopted in practice; see, e.g., Feldman et al. 2022) or the generalized attraction choice model (which is more general than MNL; see, e.g., Luce 2012, Gallego et al. 2015a), Optimal Target Problem ($\mathcal{OTP}$) can also be simplified to tractable convex programs.

[7] See https://support.google.com/google-ads/answer/2404248?hl=en.

[8] When there is no confusion in the context, we drop z^* and abbreviate it as the Fluid policy.

[9] See the GitHub repository at https://github.com/xlli878/Online_Advertisement_Allocation for the code of our simulations.

[10] In Online Appendix J, we regress the click-through on the per-period debt for all five algorithms. A high per-period debt of an ad-customer pair has a much stronger impact on the potential click-throughs under our DWO policy compared with the Fluid, Fluid-R, Fluid-I-R, and Fluid-E-R algorithms.

[11] See https://developers.facebook.com/docs/marketing-api/bidding/overview/pacing-and-scheduling.

References

Alptekinoğlu A, Banerjee A, Paul A, Jain N (2013) Inventory pooling to deliver differentiated service. *Manufacturing Service Oper. Management* 15(1):33–44.

Atkinson AB (1970) On the measurement of inequality. *J. Econom. Theory* 2(3):244–263.

Balseiro SR, Besbes O, Pizarro D (2023) Survey of dynamic resource-constrained reward collection problems: Unified model and analysis. *Oper. Res.*, ePub ahead of print May 9, https://doi.org/10.1287/opre.2023.2441.

Balseiro S, Lu H, Mirrokni V (2021) Regularized online allocation problems: Fairness and beyond. *Internat. Conf. Machine Learning* (PMLR, New York), 630–639.

Balseiro SR, Feldman J, Mirrokni V, Muthukrishnan S (2014) Yield optimization of display advertising with ad exchange. *Management Sci.* 60(12):2886–2907.

Bateni MH, Chen Y, Ciocan DF, Mirrokni V (2022) Fair resource allocation in a volatile marketplace. *Oper. Res.* 70(1) 288–308.

Bernstein F, Modaresi S, Sauré D (2019) A dynamic clustering approach to data-driven assortment personalization. *Management Sci.* 65(5):2095–2115.

Bertsimas D, Farias VF, Trichakis N (2012) On the efficiency-fairness trade-off. *Management Sci.* 58(12):2234–2250.

Bumpensanti P, Wang H (2020) A re-solving heuristic with uniformly bounded loss for network revenue management. *Management Sci.* 66(7):2993–3009.

Caro F, Martínez-de Albéniz V, Rusmevichientong P (2014) The assortment packing problem: Multiperiod assortment planning for short-lived products. *Management Sci.* 60(11):2701–2721.

Chen X, Ma W, Simchi-Levi D, Xin L (2023) Assortment planning for recommendations at checkout under inventory constraints. *Math. Oper. Res.* 49(1):297–325.

Cheung WC, Simchi-Levi D (2017) Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. Preprint, submitted November 27, https://dx.doi.org/10.2139/ssrn.3075658.

Choi H, Mela CF, Balseiro SR, Leary A (2020) Online display advertising markets: A literature review and future directions. *Inform. Systems Res.* 31(2):556–575.

Dai JG, Lin W (2005) Maximum pressure policies in stochastic processing networks. *Oper. Res.* 53(2):197–218.

Dave P (2021) Study flags gender bias in Facebook's ads tools. *Reuters* (April 9), https://www.reuters.com/technology/study-flags-gender-bias-facebooks-ads-tools-2021-04-09/.

Davis JM, Gallego G, Topaloglu H (2014) Assortment optimization under variants of the nested logit model. *Oper. Res.* 62(2):250–273.

Dong L, Shi D, Zhang F (2022) 3D printing and product assortment strategy. *Management Sci.* 68(8):5724–5744.

Feldman J, Zhang DJ, Liu X, Zhang N (2022) Customer choice models vs. machine learning: Finding optimal product displays on Alibaba. *Oper. Res.* 70(1):309–328.

Feldman M, Sorelle A, Friedler JM, Scheidegger C, Venkatasubramanian S (2015) Certifying and removing disparate impact. *Proc. 21th ACM SIGKDD Internat. Conf. Knowledge Discovery and Data Mining* (Association for Computing Machinery, New York), 259–268.

Gallego G, Ratliff R, Shebalov S (2015a) A general attraction model and sales-based linear program for network revenue management under customer choice. *Oper. Res.* 63(1):212–232.

Gallego G, Li A, Truong V-A, Wang X (2015b) Online resource allocation with customer choice. Preprint, submitted November 5, https://arxiv.org/abs/1511.01837.

Golrezaei N, Nazerzadeh H, Rusmevichientong P (2014) Real-time optimization of personalized assortments. *Management Sci.* 60(6):1532–1551.

Hao X, Peng Z, Ma Y, Wang G, Jin J, Hao J, Chen S, et al. (2020) Dynamic knapsack optimization toward efficient multi-channel sequential advertising. *Proc. 37th Internat. Conf. Machine Learning* (PMLR, New York), 4060–4070.

Hojjat A, Turner J, Cetintas S, Yang J (2017) A unified framework for the scheduling of guaranteed targeted display advertising under reach and frequency requirements. *Oper. Res.* 65(2):289–313.

Imana B, Korolova A, Heidemann J (2021) Auditing for discrimination in algorithms delivering job ads. *Proc. Web Conf. 2021* (Association for Computing Machinery, New York), 3767–3778.

Jasin S, Kumar S (2012) A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Math. Oper. Res.* 37(2):313–345.

Jiang J, Wang S, Zhang J (2023) Achieving high individual service levels without safety stock? Optimal rationing policy of pooled resources. *Oper. Res.* 71(1):358–377.

Kallus N, Udell M (2020) Dynamic assortment personalization in high dimensions. *Oper. Res.* 68(4):1020–1037.

Kumar A, Kleinberg J (2000) Fairness measures for resource allocation. *Proc. 41st Annual Sympos. Foundations Comput. Sci.* (IEEE, Piscataway, NJ), 75–85.

Lambrecht A, Tucker C (2019) Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of stem career ads. *Management Sci.* 65(7):2966–2981.

Lejeune MA, Turner J (2019) Planning online advertising using gini indices. *Oper. Res.* 67(5):1222–1245.

Liu Q, Van Ryzin G (2008) On the choice-based linear programming model for network revenue management. *Manufacturing Service Oper. Management* 10(2):288–310.

Luce RD (2012) *Individual Choice Behavior: A Theoretical Analysis* (Courier Corporation, Chelmsford, MA).

Lyu G, Chou MC, Teo C-P, Zheng Z, Zhong Y (2022) Stochastic knapsack revisited: The service level perspective. *Oper. Res.* 70(2):729–747.

Lyu G, Cheung W-C, Chou MC, Teo C-P, Zheng Z, Zhong Y (2019) Capacity allocation in flexible production networks: Theory and applications. *Management Sci.* 65(11):5091–5109.

Ma W, Xu P, Xu Y (2020) Group-level fairness maximization in online bipartite matching. Preprint, submitted November 27, https://arxiv.org/abs/2011.13908.

Mehta A (2013) Online matching and ad allocation. Foundations Trends *Theoret. Comput. Sci.* 8(4):265–368.

Microsoft (2020) *Partner Incentives Co-op Guidebook: Business Policies for FY21* (Microsoft Corporation, Redmond, WA).

Nakamura A, Abe N (2005) Improvements to the linear programming based scheduling of web advertisements. *Electronics Commerce Res.* 5(1):75–98.

Nilforoshan H, Gaebler JD, Shroff R, Goel S (2022) Causal conceptions of fairness and their consequences. *Internat. Conf. Machine Learning* (PMLR, New York), 16848–16887.

Rubin RB (1978) The uniform guidelines on employee selection procedures: Compromises and controversies. *Catholic Univ. Law Rev.* 28:605.

Rusmevichientong P, Shen ZJM, Shmoys DB (2010) Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Oper. Res.* 58(6):1666–1680.

Shen H, Li Y, Chen Y, Pan K (2021a) Integrated ad delivery planning for targeted display advertising. *Oper. Res.* 69(5):1409–1429.

Shen H, Li Y, Guan J, Tso GKF (2021b) A planning approach to revenue management for non-guaranteed targeted display advertising. *Production Oper. Management* 30(6):1583–1602.

Shi C, Wei Y, Zhong Y (2019) Process flexibility for multiperiod production systems. *Oper. Res.* 67(5):1300–1320.

Speicher T, Ali M, Venkatadri G, Ribeiro FN, Arvanitakis G, Benevenuto F, Gummadi KP, Loiseau P, Mislove A (2018) Potential for discrimination in online targeted advertising. *Conf. Fairness Accountability Transparency* (PMLR, New York), 5–19.

Stolyar AL (2004) Maxweight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic. *Ann. Appl. Probab.* 14(1):1–53.

Sumida M, Gallego G, Rusmevichientong P, Topaloglu H, Davis J (2021) Revenue-utility tradeoff in assortment optimization under the multinomial logit model with totally unimodular constraints. *Management Sci.* 67(5):2845–2869.

Turner J (2012) The planning of guaranteed targeted display advertising. *Oper. Res.* 60(1):18–33.

Wang R (2012) Capacitated assortment and price optimization under the multinomial logit model. *Oper. Res. Lett.* 40(6):492–497.

Xu K, Zhong Y (2020) Information and memory in dynamic resource allocation. *Oper. Res.* 68(6):1698–1715.

Ye Z, Zhang DJ, Zhang H, Zhang R, Chen X, Xu Z (2023) Cold start to improve market thickness on online advertising platforms: Data-driven algorithms and field experiments. *Management Sci.* 69(7):3838–3860.

Young HP (1995) *Equity: In Theory and Practice* (Princeton University Press, Princeton, NJ).

Zhong Y, Zheng Z, Chou MC, Teo C-P (2017) Resource pooling and allocation policies to deliver differentiated service. *Management Sci.* 64(4):1555–1573.