Interference by Feedback Loops: Challenges in the Counterfactual Interleaving Design (Extended Abstract)

ZHIHUA ZHU, LIANG ZHENG, ZHENG CAI, Tencent Technology (Shenzhen) Co., Ltd., China
NIAN SI, Booth School of Business, University of Chicago, U.S.A

1 INTERFERENCE CHALLENGES IN SELLER-SIDE A/B TESTS AND THE COUNTERFACTUAL INTERLEAVING DESIGN

Two-sided platforms have increasingly integrated into our daily routines. We utilize shopping platforms like Amazon and Taobao to purchase and sell goods. Video-sharing platforms such as TikTok and Kuaishou allow us to view and upload content. Moreover, platforms like Booking.com and Airbnb have simplified the process of renting out or booking accommodations. A common workflow across these platforms involves users (typically buyers) initiating a request (session), which the platform then processes to match them with a ranked list of sellers or providers.

To ensure optimal user experience, these platforms routinely conduct experiments (A/B tests) before implementing new features. While user-sided experiments are predominant, seller-sided experiments are also necessary when user-sided experiments are either infeasible or inappropriate. For instance, certain interventions, such as revamping a seller's user interface, can only be applied to sellers. Furthermore, when the objective is to measure metrics like seller retention, seller-sided experiments are indispensable.

In seller-sided experiments, interference often presents more intensely. To illustrate this challenge, consider an advertisement recommendation system in a video-sharing platform. Imagine we're evaluating a new algorithm designed to boost new ads, a "cold start" strategy. Within a seller-sided experiment, let's say we boost 50% of new ads, which consists of the treatment group, leaving the other half untouched. Due to this boost, ads within the treatment group naturally achieve a higher ranking. Yet, if we were to boost all new ads, the ones in our initial treatment group would actually descend in rank because of the increased volume of videos receiving the same boost. As a result, the data from the experiment could overstate the true impact.

To address this issue, Ha-Thuc et al. [2020] and Nandy et al. [2021] propose a counterfactual interleaving design and Wang and Ba [2023] enhance the design with a novel tie-breaking rule to guarantee consistency and monotonicity. In this approach, a subset of Ads is randomly divided into control ads and treatment ads. At the same time, during the ranking phase, both the control strategy and treatment strategy are applied to rank all ads. These two ranking strategies can be referred to as Ranking C and Ranking T, respectively. The results of these two rankings are then merged to produce a final order, M. The merging process uses the order of control ads in Ranking C as their order in M, while the order of treatment ads in Ranking T determines their order in M. If there's a position conflict, it's resolved randomly: one ad retains its spot, while the other is shifted down a slot. Consequently, the placement of ads in the treatment group in the final ranking approximates their rank when all ads are sorted using the treatment strategy. Similarly, the placement of control group ads is nearly identical to their rank under the control strategy. We plot this procedure in Figure 1.

Following its introduction, this methodology was extensively implemented across major online platforms, such as Facebook, TikTok, and Kuaishou. Although the method appeared promising initially, our analysis revealed substantial interference, particularly in settings with feedback loops.

Corresponding author: Nian Si. Email: niansi@chicagobooth.edu.

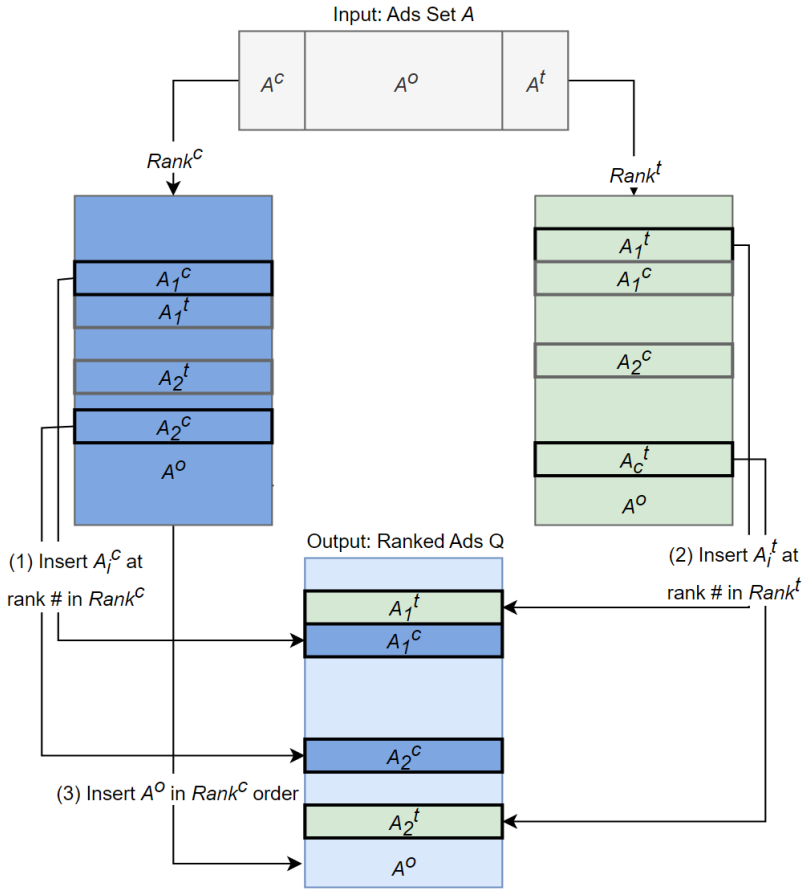

Fig. 1. Counterfactual interleave design

The rest of the paper is organized as follows: In Section 2, we explain interference induced by feedback loops and review the relevant literature on this issue. We then analyze a specific counterfactual interleaving design under the interference of feedback loops in 3. Finally, we substantiate our findings through real-world A/B test results presented in Section 4.

2 INTERFERENCE INDUCED BY FEEDBACK LOOPS

In contemporary recommendation systems, rankings in subsequent sessions can be influenced by the rankings of earlier sessions, primarily due to feedback loops. For instance, user responses gathered from initial sessions can be integrated into the training data for prediction models, thereby altering prediction outcomes for future sessions. Consider ad recommendations as another illustration. Due to campaign budget limitations and budget control mechanisms, an increased ad exposure in initial sessions can lead to fewer recommendations in subsequent sessions, and vice versa.

The presence of feedback loops in recommendation systems has been well-documented in the literature for a long time. Problems associated with the escalation of homogeneity and popularity biases due to feedback loops have been scrutinized by scholars, including Chaney et al. [2018], Mansoury et al. [2020], and Krauth et al. [2022]. Pan et al. [2021] delve into the topic of user

feedback loops and strategies for mitigating their impact. Moreover, Yang et al. [2023] and Khenissi [2022] have drawn attention to the potential fairness issues arising from these feedback loops. Furthermore, Jadidinejad et al. [2020] consider how these feedback loops affect underlying models.

Recent research has seen a growing focus on experimentation and A/B tests in the presence of feedback loops. Holtz et al. [2023] considers the challenge of training algorithms on shared data, a phenomenon they term "Symbiosis Bias." Meanwhile, Si [2023] introduces a weighted training approach to mitigate this bias effectively. In the domain of search ranking systems, Musgrave et al. [2023] advocates the use of query-randomized experiments as a means to alleviate feature spillover effects. When it comes to evaluating ranking algorithms, Goli et al. [2023] presents a bias-correction technique leveraging past A/B tests. Furthermore, for testing bandit learning algorithms, Guo et al. [2023] proposes a two-stage experimental design to estimate both the lower and upper bounds of treatment effects.

3 COUNTERFACTUAL INTERLEAVING DESIGN UNDER INTERFERENCE OF FEEDBACK LOOPS

When the treatment and control algorithms differ and produce different user-interaction dynamics, the counterfactual interleaving design would be unreliable and significantly biased by the presence of feedback loops.

Consider the testing of various pacing algorithms as a typical example. Such algorithms are prevalent in two-sided platforms, especially when seller-sided constraints are present. These can manifest as regulating the selling rate on online shopping platforms, modulating the exposure frequency in ad recommendations, or adjusting the exploration pace in cold start algorithms. Typically, a pacing algorithm will decelerate if the current consumption (exposure) is already large. Consider a scenario where the treatment algorithms employ a more rapid pacing speed. This implies that at any given time, the consumption of inventories under the treatment algorithms exceeds that of the control algorithms. In the counterfactual interleaving design, this would mean that treatment sellers consume more than their control counterparts on average. When the treatment algorithm is employed across all items to derive Ranking T, treatment items, on average, tend to land lower in rank compared to their positions in the global treatment setting. This is primarily due to the damping effect inherent in pacing algorithms. As a result, the treatment ranking realized in the experiments may not accurately reflect the ranking in the global treatment regime.

Let's delve deeper into this issue using a basic model. Imagine we have N sellers in total. We use $\mathcal{T}$ and C to denote the treatment and control groups respectively: $\mathcal{T} \cap C = \emptyset$ and $\mathcal{T} \cup C = \{1, 2, \ldots, N\}$. Let $s_i(t)$ denote the state of the i-th sellers at time t, for $i = 1, 2, \ldots, N$, capturing past user interactions. In the previous pacing example, $s_i(t)$ stands for the consumption of the i-th seller at time t. We consider a conventional ranking-by-score mechanism. Suppose that treatment and control algorithms use two different ranking score functions $r^T(s_i(t-), i)$ and $r^C(s_i(t-), i)$, which may depend on the current states differently. To emphasize the effects on the state dependence on treatment/control algorithms, we use $s_i^T(t)$ and $s_i^C(t)$ to denote the state process under treatment and control algorithms, respectively. Then, a systematic difference between $s_i^T(t)$ and $s_i^C(t)$ means the combined ranking scores $\left\{r^T\left(s_i^T(t-), i\right), i \in \mathcal{T}; r^T\left(s_j^C(t-), j\right), j \in C\right\}$ would deviate from those in the global treatment regime, represented as $\left\{r^T\left(s_i^T(t-), i\right), i = 1, 2, \ldots, N\right\}$, which further results in different rankings in the experiment and the global treatment scenario.

4 EMPIRICAL RESULTS BASED ON REAL-WORLD A/B TESTS

We partnered with the advertising recommendation team at Tencent, a world-class content-sharing platform. In the advertising recommendations, it is commonly observed the estimated scores

overestimate the real effect, due to perhaps maximization bias, especially for those ads with low impressions [Fan et al., 2022]. To address this bias, Tencent implemented a strategy: at any given time t, if the cumulative realized value (clicks, conversions, etc.) up to that moment surpasses the cumulative estimated value, the current estimated scores are adjusted downward. Conversely, if the overall realized value falls short of the cumulative estimated value, the scores are incremented.

In mathematical terms, let's denote $e(t, i)$ as the raw estimated score for the i-th ad in relation to a request at time t. Additionally, $s_i(\cdot)$ is the state process that captures the cumulative overestimation (or underestimation) for the i-th ad up until time t. Given these, the ranking score can be defined as $r(s_i(t-), i) = \lambda_i(s_i(t-))e(t, i)$, for $i = 1, \ldots, N$, where $\lambda_i(s_i(t-))$ represents the adjustment factor. At the beginning of a day, λ is initialized at 1. Further, $\lambda_i(s)$ decreases as s increases, and $\lambda_i(s_i(t)) < 1$ signifies overestimation ($s_i(t) > 1$), while $\lambda_i(s_i(t)) > 1$ indicates underestimation ($s_i(t) < 1$).

Our empirical observation reveals a challenge of this adjusting mechanism: the values of λs tend to fluctuate significantly due to the inherent randomness in realized values. Such volatility can negatively impact the overall performance. To address this, we devised a new strategy to constrain the variability of λ, effectively reducing its swing or "effective range."

In our experiments, we'll contrast this new strategy with the original approach using the counterfactual interleaving design. More precisely, we'll be comparing the treatment adjustment method $\lambda^T(\cdot)$ with the control adjustment method $\lambda^C(\cdot)$, where we allocate 10% control ads and 10% treatment ads. Specifically, the platform ranks and charges based on the scores

$$r^T(s_i(t-), i) = \lambda_i^T(s_i(t-))e(t, i), \text{ and } r^C(s_i(t-), i) = \lambda_i^C(s_i(t-))e(t, i), \text{ for } i = 1, 2, \ldots, N,$$

in the treatment and control groups, respectively, and $|\lambda_i^T(s) - 1| < |\lambda_i^C(s) - 1|$ for any $s \in \mathbb{R}_+$.

Despite simulations and A/A tests consistently indicating the superiority of the treatment strategy over the control strategy, the counterfactual interleaving design suggests otherwise:

Table 1. The experimental results using the counterfactual interleaving design

Advertising cost (consumption)		Views		Gross merchandise value (GMV)	
Estimator	Confidence Interval	Estimator	Confidence Interval	Estimator	Confidence Interval
-23%	[-34%,-12%]	-27%	[-38%,-15%]	-21%	[-34%,-9%]

We attribute this discrepancy to interference arising from feedback loops. Given the systematic overestimation and our specific treatment strategy, it's anticipated that $\lambda_i^T(s_i(t-))$ would typically exceed $\lambda_i^C(s_i(t-))$. Consequently, $s_i^T(t-) > s_i^C(t-)$ in general. Due to the inverse monotonic behavior of $\lambda(\cdot)$, this means that $\lambda_i^T(s_i^T(t-)) < \lambda_i^T(s_i^C(t-))$ and $\lambda_i^C(s_i^T(t-)) < \lambda_i^C(s_i^C(t-))$, causing that the treatment ads frequently rank lower, while control ads often rank higher in the experiment. To bolster our rationale, we visualize the average λ^T values under the treatment strategy, $\bar{\lambda}^T$s, for the control ads, treatment ads, and other ads over time in Figure 2. The figure elucidates that while the $\bar{\lambda}^T$s are nearly identical across the three groups initially, the $\bar{\lambda}^T$s in the treatment group noticeably diminishes towards the day's end.

REFERENCES

Allison JB Chaney, Brandon M Stewart, and Barbara E Engelhardt. 2018. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *Proceedings of the 12th ACM conference on recommender systems*. 224–232.

Yewen Fan, Nian Si, and Kun Zhang. 2022. Calibration Matters: Tackling Maximization Bias in Large-scale Advertising Recommendation Systems. *arXiv preprint arXiv:2205.09809* (2022).

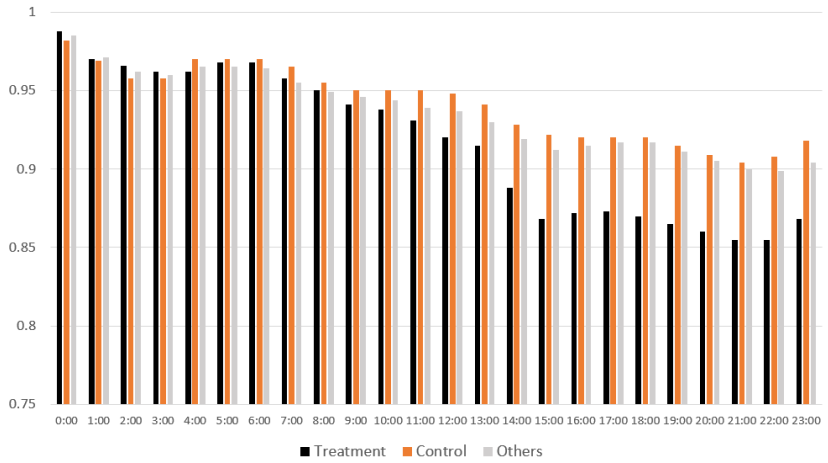

Fig. 2. Average λ^T values, $\bar{\lambda}^T$s, for the control ads, treatment ads, and other ads over time in a day

Ali Goli, Anja Lambrecht, and Hema Yoganarasimhan. 2023. A bias correction approach for interference in ranking experiments. *Marketing Science* (2023).

Hongbo Guo, Ruben Naeff, Alex Nikulkov, and Zheqing Zhu. 2023. Evaluating Online Bandit Exploration In Large-Scale Recommender System. In *KDD-23 Workshop on Multi-Armed Bandits and Reinforcement Learning: Advancing Decision Making in E-Commerce and Beyond.*

Viet Ha-Thuc, Avishek Dutta, Ren Mao, Matthew Wood, and Yunli Liu. 2020. A counterfactual framework for seller-side a/b testing on marketplaces. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 2288–2296.

David Holtz, Jennifer Brennan, and Jean Pouget-Abadie. 2023. A Study of" Symbiosis Bias" in A/B Tests of Recommendation Algorithms. *arXiv preprint arXiv:2309.07107* (2023).

Amir H Jadidinejad, Craig Macdonald, and Iadh Ounis. 2020. Using exploration to alleviate closed loop effects in recommender systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 2025–2028.

Sami Khenissi. 2022. Modeling and debiasing feedback loops in collaborative filtering recommender systems. (2022).

Karl Krauth, Yixin Wang, and Michael I Jordan. 2022. Breaking feedback loops in recommender systems with causal inference. *arXiv preprint arXiv:2207.01616* (2022).

Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. 2020. Feedback loop and bias amplification in recommender systems. In *Proceedings of the 29th ACM international conference on information & knowledge management.* 2145–2148.

Paul Musgrave, Cuize Han, and Parth Gupta. 2023. Measuring service-level learning effects in search via query-randomized experiments. (2023).

Preetam Nandy, Divya Venugopalan, Chun Lo, and Shaunak Chatterjee. 2021. A/b testing for recommender systems in a two-sided marketplace. *Advances in Neural Information Processing Systems* 34 (2021), 6466–6477.

Weishen Pan, Sen Cui, Hongyi Wen, Kun Chen, Changshui Zhang, and Fei Wang. 2021. Correcting the user feedback-loop bias for recommendation systems. *arXiv preprint arXiv:2109.06037* (2021).

Nian Si. 2023. Tackling Interference Induced by Data Training Loops in A/B Tests: A Weighted Training Approach. *arXiv preprint arXiv:2310.17496* (2023).

Yan Wang and Shan Ba. 2023. Producer-Side Experiments Based on Counterfactual Interleaving Designs for Online Recommender Systems. *arXiv preprint arXiv:2310.16294* (2023).

Mengyue Yang, Jun Wang, and Jean-Francois Ton. 2023. Rectifying unfairness in recommendation feedback loop. In *Proceedings of the 46th international ACM SIGIR Conference on Research and Development in Information Retrieval.* 28–37.