

Percentile Risk-Constrained Budget Pacing for Guaranteed Display Advertising in Online Optimization

Liang Dai¹, Kejie Lyu¹, Chengcheng Zhang¹, Guangming Zhao², Zhonglin Zu¹, Liang Wang², Bo Zheng²

¹Alibaba Group, Hangzhou, China

²Alibaba Group, Beijing, China

{dailiang.dl, lvkejie.lkj, xuelun.zcc, lambert.zgm, zhonglin.zu1, liangbo.wl, bozheng}@alibaba-inc.com

Abstract

Guaranteed display (GD) advertising is a critical component of advertising since it provides publishers with stable revenue and enables advertisers to target specific audiences with guaranteed impressions. However, smooth pacing control for online ad delivery presents a challenge due to significant budget disparities, user arrival distribution drift, and dynamic change between supply and demand. This paper presents robust risk-constrained pacing (RCPacing) that utilizes Lagrangian dual multipliers to fine-tune probabilistic throttling through monotonic mapping functions within the percentile space of impression performance distribution. RCPacing combines distribution drift resilience and compatibility with guaranteed allocation mechanism, enabling us to provide near-optimal online services. We also show that RCPacing achieves $O(\sqrt{T})$ dynamic regret where T is the length of the horizon. RCPacing's effectiveness is validated through offline evaluations and online A/B testing conducted on Taobao brand advertising platform.

Introduction

According to a report by the Internet Advertising Bureau, online display advertising generated a remarkable revenue of \$63.5 billion in 2022, demonstrating a substantial year-over-year increase of 12.0% (IAB 2023). Ad exposures in display markets are sold through both guaranteed and non-guaranteed (like real-time bidding or RTB) selling channels. Within the guaranteed display (GD) selling channel, an advertiser (demand side) and publisher (supply side) negotiate a fixed price (cost-per-mille or CPM) for ad placement, including details such as when, where, and how the ad campaigns will be displayed. These contractual arrangements guarantee the delivery of a specified number of impressions that meet specific targeting criteria during a specified period.

In addition to contractual agreements, advertisers usually expect their ad campaigns to be delivered smoothly and steadily during the purchased period for various reasons, including making campaign performance as good as possible, reaching a wider audience, increasing the display ratio of the target audience, and maintaining stable online viewership for live streaming events. However, smooth and robust

pacing control for hundreds or thousands of GD advertisements on a brand advertising platform that deals with billions of daily requests is a challenging task. To summarize, the main challenges are as follows:

- Significant differences among campaigns: the guaranteed daily impressions range from thousands to millions, the targeted audience size also vary greatly. Moreover, different campaigns have different optimization goals, such as click-through rate (CTR) or conversion rate (CVR).
- Drastic changes in traffic environment: these changes include significant fluctuations in overall traffic, dynamic shifts in the distribution of user arrival over time, and the impact of other campaigns going online or offline.

The existing smooth pacing techniques have primarily focused on RTB ads (Nuara et al. 2022; Liu et al. 2020), which is incompatible with GD allocation. Although some research has considered the smoothness or representativeness in online optimal allocation of GD ads, it is often not optimized and evaluated as a separate key metric. In this paper, we consider smooth pacing for GD ads from the perspective of a publisher. Our contributions can be summarized as follows:

- We introduce a novel framework called RCPacing, which employs Lagrangian dual multipliers to adjust probabilistic throttling based on monotonic functions within the percentile space, allowing us to effectively manage risk and ensure optimal ad delivery performance.
- We also show that RCPacing attains regret of order $O(\sqrt{T})$ when the length of the horizon T and the initial number of resources are scaled proportionally.
- As there exists a tradeoff between smooth and optimal allocation in online matching problems, RCPacing offers flexible control over this balance.
- We implement RCPacing in our online display advertising system and conduct extensive online/offline experimental evaluations. The results demonstrate that RCPacing is highly effective in improving both the performance and smoothness of online delivery for GD campaigns.

Related Work

In the past few years, the allocation of GD advertising has received significant attention from researchers (Wu et al. 2021; Wang et al. 2022). It is typically modeled as an online

matching problem, intending to achieve the maximum match between impressions and contracts (Chen et al. 2012). While the primary objective is to provide each advertiser with a predetermined number of display opportunities, it is also necessary to consider the smoothness of budget consumption. Some researchers include a fixed representative term in objectives (Fang et al. 2019; Dai et al. 2023; Bharadwaj et al. 2012), which aims to minimize the deviation between the allocation probability and its corresponding supply-demand ratio for each contract. However, the representative term is fixed without consideration of dynamic adjustment.

Another research direction is to achieve budget pacing through feedback control, which can be further categorized into bid modification (Mehta et al. 2007; Zhou et al. 2021) and probabilistic throttling (Agarwal et al. 2014; Xu et al. 2015; Lee, Jalali, and Dasdan 2013). Bid modification influences the budget spending of an ad by adjusting its bidding price. Mehta et al. (Mehta et al. 2007) modify the bid by multiplying it with a value that reflects the proportion of unused budget, and the impression would be allocated to the ad with the highest modified bid. Balseiro et al. (Balseiro, Lu, and Mirrokni 2020) and Zhou et al. (Zhou et al. 2021) utilize a dual multiplier to form a virtual bid, which is consistently updated based on the variance between the actual and expected budget consumption. These methods adhere to the same principle of decreasing an ad's bid when the budget is being spent too rapidly. However, both the dramatic change in the bid win-rate curve and bid landscape make it challenging to control the budget through bid modification.

On the other hand, probabilistic throttling methods decouple spending control from bid calculation, they directly impact the participating likelihood based on its budget consumption speed. Agarwal et al. (Agarwal et al. 2014) set a global pass-through rate (PTR), which is decreased when the budget consumption speed exceeds the expectation, and increased when the consumption speed falls below. Although this method demonstrates good budget control capability, it heavily relies on the accuracy of traffic forecasting.

To further consider performance optimization while achieving budget control, Xu et al. (Xu et al. 2015) group requests with similar response rates (e.g. CTR) together and share a PTR among them. When the PTRs need to be adjusted, the average response rate of each group determines the priority of that group. While effective in budget control, relying solely on PTR regulation is insufficient to ensure guaranteed display for GD allocation.

Preliminaries

Problem Formulation

We formulate the GD allocation problem as the following optimization problem:

$$\begin{aligned} \max_{x^{(t)} \in \mathcal{X}} \sum_{t=0}^{T-1} f_t(x^{(t)}) &= \sum_{t=0}^{T-1} v^{(t)\top} x^{(t)} \\ \text{s.t.} \sum_{t=0}^{T-1} x^{(t)} &\leq B \end{aligned} \quad (1)$$

where $x^{(t)} \in \mathcal{X} \subseteq \mathbb{R}^M$ is the one-hot decision vector at time $t \in [1, T]$, M is the total number of campaigns and the impression arrived at time t would be allocated to the j -th campaign if the j -th component of $x^{(t)}$ is 1, $v^{(t)} \in \mathbb{R}^M$ denotes the impression quality between the impression and the campaigns, $f_t(x^{(t)}) = v^{(t)\top} x^{(t)} \in \mathbb{R}$ is the revenue obtained at time t , $B \in \mathbb{R}^M$ is the positive budget vector which represents the campaign budgets.

Following Balseiro et al (Balseiro, Lu, and Mirrokni 2020), we define the offline dual problem as

$$\begin{aligned} \min_{\alpha \geq 0} D(\alpha) &= \sum_{i=1}^n p_i f_i^*(\alpha) + \alpha^\top \rho \\ &= \sum_{i=1}^n p_i \max_{x \in \mathcal{X}} \left\{ v^{(i)\top} x - \alpha^\top x \right\} + \alpha^\top \rho \end{aligned} \quad (2)$$

where $f_i^*(\alpha) := \max_{x \in \mathcal{X}} \{f_i(x) - \alpha^\top x\}$ is the conjugate function of $f_i(x)$ (restricted in \mathcal{X}), p_i is the probability that the i -th impression has a quality vector of $v^{(i)}$, n is the total number of impressions, $\rho = B/T$ is the average budget for each time period, α is the dual variable, and the j -th element of α (denoted as α_j) reflects the additional revenue generated by allowing one unit of resources to be added to the j -th campaign's budget.

Dual Mirror Descent Algorithm

Our method is built upon the Dual Mirror Descent (DMD) algorithm (Balseiro, Lu, and Mirrokni 2020), which addresses the general online allocation problem with budget constraint. At time t , DMD filters out campaigns that have exhausted their budget and assigns the request to the campaign that offers the highest premium among the remaining campaigns (equation 3). The dual variable is then updated according to the online mirror descent (equation 6). More details about DMD is given in Algorithm 1.

Motivation

Assumptions

In this paper, we adopt the following assumptions:

- **The small bids assumption.** Each impression has only one slot available for displaying ads, which is significantly lower than the demand for campaigns and the supply of publishers (Mehta 2012).
- **The known IID assumption.** The known Independent and Identically Distributed (IID) assumption implies that impressions arrive online according to a known probability distribution with repetition (Huang and Shu 2021), which is a realistic assumption in our problem.

Motivation

Upon receiving a user's online request, the ad engine retrieves GD campaigns with recall rate (RR) that meet the targeting criteria and employs real-time prediction models such as deep neural networks (DNN) to estimate performance scores for each campaign (Zhou et al. 2019; Gao et al. 2021). The decision-maker then determines whether

Algorithm 1: Dual Mirror Descent Algorithm

Require: Time period T , remaining resources $B^{(0)} = T\rho$, reference function $h(\cdot) : \mathbb{R}^M \rightarrow \mathbb{R}$, and step-size η .

- 1: $\alpha^{(0)} = 0$
- 2: **for** $t = 0$ **to** $T - 1$ **do**
- 3: Receive $v^{(t)} \sim \mathcal{P}$
- 4: Make the decision $\tilde{x}^{(t)}$ and update the remaining resources $B^{(t+1)}$, where

$$\tilde{x}_j^{(t)} = \begin{cases} 1, & \text{if } j = \arg \max_{B_j^{(t)} \geq 1} \{v_j^{(t)} - \alpha_j^{(t)}\} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

$$B^{(t+1)} = B^{(t)} - x^{(t)} \quad (4)$$

- 5: Obtain a stochastic sub-gradient of $D(\alpha^{(t)})$

$$\tilde{g}^{(t)} := -\tilde{x}^{(t)} + \rho \quad (5)$$

- 6: Update the dual variable by mirror descent

$$\alpha^{(t+1)} = \arg \min_{\alpha \geq 0} \left\langle \tilde{g}^{(t)}, \alpha \right\rangle + \frac{1}{\eta} V_h(\alpha, \alpha^{(t)})$$

where $V_h(\alpha, \alpha^{(t)}) = h(\alpha) - h(\alpha^{(t)}) - \left\langle \nabla h(\alpha), \alpha - \alpha^{(t)} \right\rangle$ (6)

7: **end for**

and which campaign to display based on the scores. The detailed processing flow is illustrated in figure 1. The campaign j passes the pacing control module with the pass-through rate (PTR) before the calculation of the “price premium”. It will only win out if it has the highest positive price compared to all other campaigns. The positive ratio of price and the win-out ratio are referred to as the participation ratio (PR) and win rate (WR). Without loss of generality, we use CTR as performance score in the following paragraph. The cost for campaign j can be denoted as:

$$\mathbb{E}[Cost_j] = \mathbb{E}[RR_j] \sum PTR_j PR_j WR_j \quad (7)$$

It is worth noting that the primary risk GD campaigns is over-spend of the budget because it is irreversible once the budget is over-spent. Apart from PTR, the delivery of GD campaign is primarily determined by PR and WR. Let’s consider two GD campaigns Ad1 and Ad2, with identical impression budgets, performance distributions, and similar competitive environments, but with different supply amounts ($Ad1 > Ad2$).

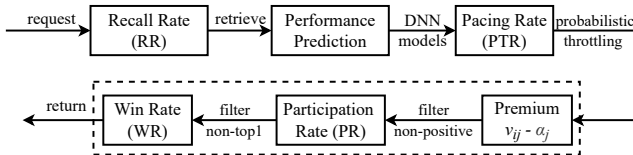


Figure 1: Online process for GD campaign j .

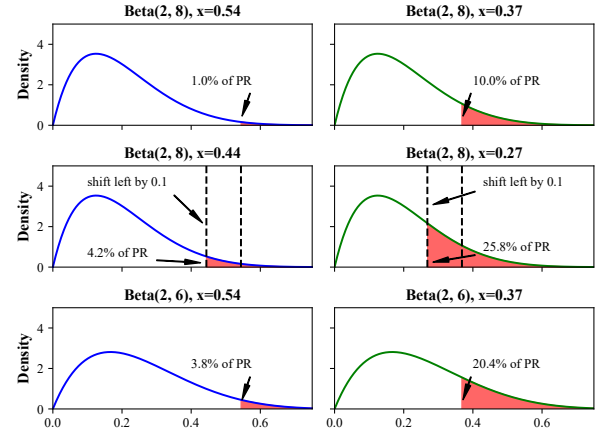


Figure 2: Different changes of PR when adjusting for dual variables or distribution drift with the same magnitude.

When the online allocation reaches a stable state, the dual variable of Ad1 is located at a higher percentile than Ad2 in performance distribution.

- **Risk analysis under stable conditions:** A higher percentile indicates a greater potential available traffic for Ad1, which makes it more vulnerable to over-spending. Moreover, Ad1 is more challenging to initialize dual variables because a higher percentile implies higher uncertainty especially before the start of delivery.
- **Risk analysis under dynamic conditions:** Higher percentile results in a smaller bid price, making the campaign more susceptible to over-acceleration if other campaigns suddenly go offline. Moreover, as shown in the figure 2, the PR of Ad1 generates greater fluctuations if the dual variable shifts the same distance, and is more sensitive to distribution drift in user arrival or switching of online prediction models, which can be deduced by the proof of Theorem 2 and Theorem 3 in appendix¹.

Based on the above risk analysis, RCPacing is designed to adjust the dual variables in the dual percentile space, while constraining dual variables within the low-risk region through the pacing module using probabilistic throttling method.

Risk-Constrained Pacing Algorithm

The factor dependency of RCPacing is illustrated in figure 3. The dual variables and PTRs are adjusted in the dual percentile space of performance distributions. These two factors jointly determine the final win-out of each request. Although RCPacing adjusts the dual variables in percentile space rather than dual space, the Theorem 1 in appendix shows that it attains regret of order $O(\sqrt{T})$ when the length of the horizon T and the initial number of resources are scaled proportionally.

Parametric Percentile Transformation

Forward Transformation RCPacing converts the CTR into the percentile space, which is called forward trans-

¹<https://arxiv.org/abs/2312.06174>.

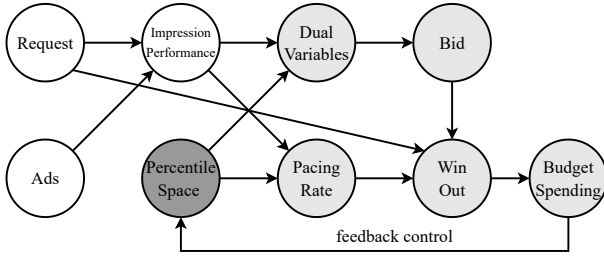


Figure 3: Factor dependency graph of RCPacing.

formation, to assess the non-smooth risk and standardize the range of dual variables. Specifically, the CTR is first subjected to statistical Box-Cox transformation to achieve a normal shape, after which it is converted into the percentile space using the normal cumulative distribution function $\Phi(x)$. The parameter λ_j^* of campaign j can be estimated from global or campaign's historical logs using the maximum-likelihood method (Sakia 1992):

$$\lambda_j^* = \underset{\lambda_j}{\operatorname{argmax}} MLE(\lambda_j, v_{ij}) \quad (8)$$

And Box-Cox transformation can be denoted as:

$$v_{ij}^{boxc} = \operatorname{BoxCox}(\lambda_j^*, v_{ij}) = \begin{cases} \frac{v_{ij}^{\lambda_j^*} - 1}{\lambda_j^*} & \text{if } \lambda_j^* \neq 0 \\ \ln(v_{ij}) & \text{if } \lambda_j^* = 0 \end{cases} \quad (9)$$

The mean μ_j and standard deviation σ_j can be estimated:

$$\mu_j = \mathbb{E}(v_{ij}^{boxc}), \sigma_j = \sqrt{\mathbb{E}[(v_{ij}^{boxc} - \mu_j)^2]} \quad (10)$$

To improve the robustness of drifts in the user arrival distribution, RCPacing skews the transformation towards the middle percentile region by a factor ϵ :

$$\bar{v}_{ij} = \Phi\left(\frac{\operatorname{BoxCox}(\lambda_j^*, v_{ij}) - \mu_j}{\sigma_j + \epsilon\sigma_j}\right), \text{ where } \epsilon \geq 0 \quad (11)$$

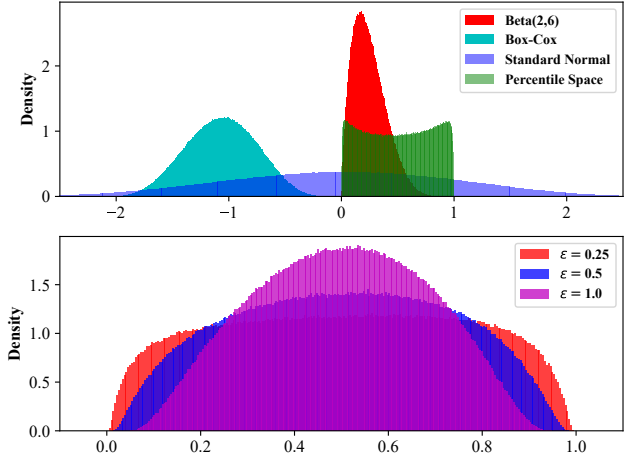
Backward Transformation RCPacing periodically updates the dual variables in the percentile space through feedback and then performs a backward transformation of the percentile variables $\bar{\alpha}_j$ into the original dual space α_j for online service. It guarantees that RCPacing approaches the optimal solution in the original space rather than percentile space. Here is the backward process:

$$\alpha_j = \operatorname{BoxCox}^{-1}(\lambda_j^*, \mu_j + \Phi^{-1}(\bar{\alpha}_j) * (\sigma_j + \epsilon\sigma_j)) \quad (12)$$

Pacing Rate Factor Decoupling

The pacing rate serves multiple functions in RCPacing, including constraining the percentile of dual variables within the safety region and addressing unexpected environmental changes and cold-start problem. RCPacing decouples the pacing rate into different factors to achieve optimal performance, for the campaign j retrieved in request i :

$$\overline{PTR}_{ij} = PTR_j^{base} \cdot fp(\bar{\alpha}_j) \cdot fv(\bar{\alpha}_j, \bar{v}_{ij}) \quad (13)$$

Figure 4: The transformation process from beta distribution to percentile uniform distribution and the different skewness of the distribution under different ϵ .

where PTR_j^{base} is the basic statistical PTR, $f(\cdot)$ and $fv(\cdot)$ are the fine-tune factors. Given a safe upper bound of percentile threshold P_{ub} (such as 90%), the expected PTR can be calculated based on its targeted audience TA_j without considering the competition from other campaigns:

$$PTR_j^{exp} = \frac{B_j}{(1.0 - P_{ub})TA_j} \quad (14)$$

The initial value of $\bar{\alpha}_j$ can be expressed as:

$$\bar{\alpha}_j^{(0)} = \begin{cases} P_{ub}, & \text{if } PTR_j^{exp} \leq 1 \\ 1 - (1 - P_{ub})PTR_j^{exp}, & \text{otherwise.} \end{cases} \quad (15)$$

Given the global hyper-parameter WR_{glb} (such as 0.2), the basic PTR considering the competition of WR can be expressed as:

$$PTR_j^{base} = \min\{1.0, PTR_j^{exp}/WR_{glb}\} \quad (16)$$

During the dynamic update in RCPacing, PTR_j should be gradually increased to enhance traffic supply if $\bar{\alpha}_j < P_{ub}$. Conversely, it should be quickly decayed to reduce the non-smooth risk. It is illustrated in equation 17 and figure 5:

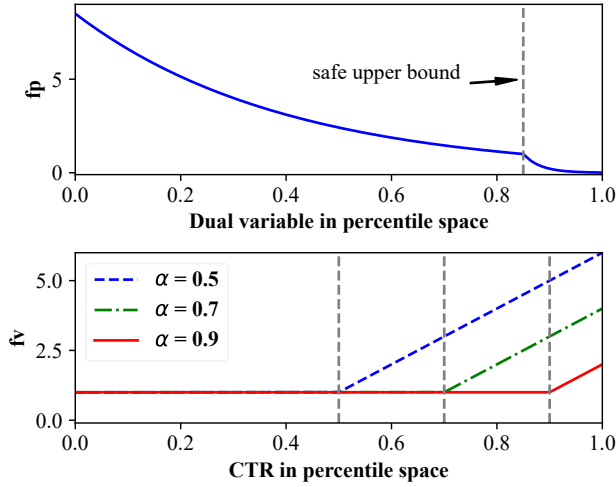
$$fp(\bar{\alpha}_j) = \begin{cases} 50^{(P_{ub} - \bar{\alpha}_j)/P_{ub}}, & \text{if } \bar{\alpha}_j \leq P_{ub} \\ 0.2^{(P_{ub} - \bar{\alpha}_j)/(P_{ub} - 1)}, & \text{otherwise.} \end{cases} \quad (17)$$

Taking inspiration from smart pacing, RCPacing assigns a higher PTR to traffic with higher performance scores. Instead of employing discrete layered pacing, RCPacing utilizes linear functions to achieve non-uniform pacing:

$$fv(\bar{\alpha}_j, \bar{v}_{ij}) = 10(\bar{v}_{ij} - \bar{\alpha}_j) + 1 \quad (18)$$

Emergence Control and Cold Start Problem

Despite RCPacing's adaptive adjustment of the PTR, it cannot completely mitigate the risks of non-smooth delivery caused by unpredictable factors, such as sharp increases in

Figure 5: Functions of fp and fv in percentile space.

user traffic, significant distribution changes due to the switch of online real-time prediction models, offsets caused by updates of Box-Cox parameters, and modifications of budgets. Additionally, due to the absence of historical logs, there is also a risk of non-smoothness during the cold start phase. To address these risks, RCPacing incorporates an emergent PTR intervention module (ePTR) that can be activated in emergency situations. The final PTR can be denoted as:

$$PTR_{ij} = \min\{1, \overline{PTR}_{ij}\} \times ePTR_j \quad (19)$$

The motivation behind ePTR is to limit the consumption speed within a certain range when a campaign is over-accelerated while maintaining the gradient direction of dual variables. The ratio of the actual cost to the expected cost can represent the spending speed of campaign j during period t :

$$spd_j^{(t)} = \frac{Cost_j^{(t)}}{eCost_j^{(t)}} \quad (20)$$

RCPacing uses proportional control instead of gradient methods to quickly control the risks. Given a safe upper ratio 2.0, the update of ePTR is:

$$ePTR_j^{(t+1)} = \min\{1, ePTR_j^{(t)} * \min\{2, \frac{2}{spd_j^{(t)}}\}\} \quad (21)$$

An initial trial rate is usually set for each campaign at the start of delivery to reduce the risks of the cold start problem.

Adaptive Gradient Clipping

Stable online iterative update of dual variables is also a critical factor for smooth delivery. However, choosing inappropriate learning rates can result in significant fluctuations and may have a cascading effect on the overall competitive environment. A simple and direct method is to restrict the change range into $\hat{\alpha}$ by gradient clipping (Chen, Wu, and Hong 2020). Given the updated dual variables $\tilde{\alpha}_j^{(t+1)}$, gradient clipping can be denoted as:

$$\bar{\alpha}_j^{(t+1)} = \max\left\{\tilde{\alpha}_j^{(t)} - \hat{\alpha}, \min\left\{\tilde{\alpha}_j^{(t+1)}, \tilde{\alpha}_j^{(t)} + \hat{\alpha}\right\}\right\} \quad (22)$$

Suppose that $spd_j^{(t)} < 1.0$, which indicates that the campaign's spending is lower than expected. The feedback control method will decrease the value of $\alpha_j^{(t)}$ to $\alpha_j^{(t+1)}$, leading to an increase in the bid price. Assuming that the competition remains the same, which indicates that $WR_{ij}^{(t+1)} \geq WR_{ij}^{(t)}$, if $v_{ij}^{(t+1)} = v_{ij}^{(t)}$. Suppose the expected spending speed in the next period is equal to 1, it can be deduced that:

$$\begin{aligned} 1.0 &= \frac{Cost_j^{(t+1)}}{eCost_j^{(t+1)}} = \frac{Cost_j^{(t+1)}}{eCost_j^{(t)}} = spd_j^{(t)} \frac{Cost_j^{(t+1)}}{Cost_j^{(t)}} \\ &= spd_j^{(t)} \frac{\mathbb{E}[RR_j^{(t+1)}] \sum PTR_j^{(t+1)} PR_j^{(t+1)} WR_j^{(t+1)}}{\mathbb{E}[RR_j^{(t)}] \sum PTR_j^{(t)} PR_j^{(t)} WR_j^{(t)}} \\ &\geq spd_j^{(t)} \sum PTR_j^{(t+1)} PR_j^{(t+1)} / \sum PTR_j^{(t)} PR_j^{(t)} \\ &= spd_j^{(t)} \mathbb{E}[PTR_j^{(t+1)} PR_j^{(t+1)}] / \mathbb{E}[PTR_j^{(t)} PR_j^{(t)}] \end{aligned} \quad (23)$$

Without consideration the effect of $ePTR$, PTR and PR are determined and have a monotonic decreasing relationship with $\bar{\alpha}$. We can calculate the expectation using the importance sampling method in uniform percentile space:

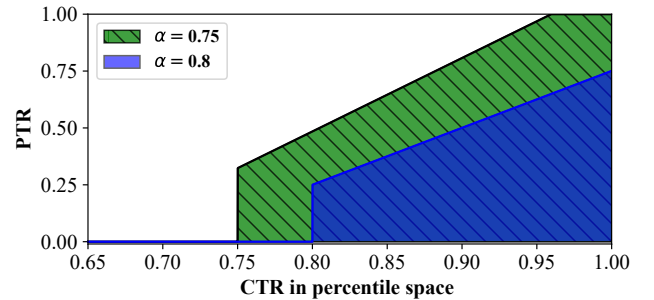
$$\begin{aligned} \psi_j(\bar{\alpha}_j^{(t)}) &= \mathbb{E}[PTR_j^{(t)} PR_j^{(t)}] = \int_0^1 PTR_j^{(t)}(\bar{\alpha}_j^{(t)}, x) \cdot PR_j(\bar{\alpha}_j^{(t)}, x) dx, \quad x \sim \text{uniform}(0, 1) \end{aligned} \quad (24)$$

The lower bound of $\alpha_j^{(t+1)}$ can be represented as:

$$\bar{\alpha}_j^{(t+1)} \geq \psi_j^{-1}(\psi_j(\bar{\alpha}_j^{(t)}) / spd_j^{(t)}) = \psi_j^{-1} \quad (25)$$

where $y = \psi_j^{-1}(\bar{\alpha}_j, x)$ can be approximated through an iterative process by solving the equation $\psi_j(\bar{\alpha}_j, y) = x$ based on the bisection method illustrated in figure 6. To include $spd_j^{(t)} \geq 1.0$, $\bar{\alpha}_j^{(t+1)}$ should satisfy the following conditions:

$$\bar{\alpha}_j^{(t+1)} = \begin{cases} \max\{\bar{\alpha}_j^{(t+1)}, \bar{\alpha}_j^{(t)} - \hat{\alpha}, \psi_j^{-1}\}, & \text{if } \bar{\alpha}_j^{(t)} \geq 0 \\ \min\{\bar{\alpha}_j^{(t+1)}, \bar{\alpha}_j^{(t)} + \hat{\alpha}, \psi_j^{-1}\}, & \text{otherwise.} \end{cases} \quad (26)$$

Figure 6: The areas of the color section represent the value of $\psi_j(\bar{\alpha}_j^{(t)})$ under different variables $\alpha_j^{(t)}$.

Algorithm 2: RCPacing

Require: Budget of the campaigns B , safe upper bound P_{ub} , global win rate WR_{glb} , skew factor ϵ , step size η , static gradient clipping $\hat{\alpha}$, total time period T

- 1: Budget exhausted campaign set $\mathcal{G} = \emptyset$
- 2: Calculate PTR^{base} and $\bar{\alpha}^{(0)}$ with eq. 14 ~ 16
- 3: **for** $t = 0$ **to** $T - 1$ **do**
- 4: Estimate λ^* , μ , and σ from historical logs with eq. 8 ~ 10
- 5: Obtain $\alpha^{(t)}$ from $\bar{\alpha}^{(t)}$ by backward transformation in eq. 12
- 6: Receive $v^{(t)}$ from online requests
- 7: Obtain $\bar{v}^{(t)}$ from $v^{(t)}$ by forward transformation in eq. 11
- 8: Calculate $PTR^{(t)}$ with eq. 13 and eq. 17 ~ 19
- 9: $bid^{(t)} = v^{(t)} - \alpha^{(t)}$
- 10: Element-wise randomly set $bid_{ij}^{(t)} = 0$ with probability $1 - PTR_{ij}^{(t)}$ and set $bid_{ij}^{(t)} = 0$ if $j \in \mathcal{G}$
- 11: $j^* = \arg \max \{bid^{(t)}\}$
- 12: Make the decision $\tilde{x}^{(t)}$, where

$$\tilde{x}_{ij}^{(t)} = \begin{cases} 1, & \text{if } bid_{ij}^{(t)} > 0 \text{ and } j = j^* \\ 0, & \text{otherwise.} \end{cases} \quad (29)$$
- 13: $B = B - \sum_i \tilde{x}^{(t)}$
- 14: Add budget exhausted campaign to \mathcal{G}
- 15: Calculate $\tilde{\alpha}^{(t+1)}$ with eq. 28
- 16: Update $\bar{\alpha}^{(t+1)}$ by clipping $\tilde{\alpha}^{(t+1)}$ with eq. 26
- 17: Update $ePTR^{(t+1)}$ with eq. 21
- 18: **end for**

Bregman Divergence Selection

Algorithm 1 presents the basic decision process based on Bregman divergence with respect to a given convex reference function. It is obvious that if we use the squared loss function and the dual update becomes:

$$h(\alpha) = \alpha^2 \Rightarrow \tilde{\alpha}_j^{(t+1)} = \bar{\alpha}_j^{(t)} - \eta \tilde{g}_j^{(t)}, \forall j \quad (27)$$

However, due to the higher fluctuation of PR in the high percentile region with the same shift, the variation magnitude of dual variables should be smaller to minimize non-smooth risk. It means that as $\bar{\alpha}_j$ approaches 1.0, the η should become smaller. We propose a modified Itakura-Saito divergence (Banerjee et al. 2005) to achieve this objective:

$$h(\alpha) = -\ln(1.5 - \alpha) \Rightarrow \tilde{\alpha}_j^{(t+1)} = \bar{\alpha}_j^{(t)} - \frac{(1.5 - \bar{\alpha}_j^{(t)})^2}{1 - \eta \tilde{g}_j^{(t)}(1.5 - \bar{\alpha}_j^{(t)})} \eta \tilde{g}_j^{(t)}, \forall j \quad (28)$$

where $\eta \tilde{g}_j^{(t)}(1.5 - \bar{\alpha}_j^{(t)}) < 1$

The overall processing details of RCPacing are described in Algorithm 2.

parameter	value	description
ϵ	0.1	skew factor
η	0.2	step size
$\hat{\alpha}$	0.05	static gradient clipping
P_{ub}	90%	safe percentile upper bound
WR_{glb}	15%	global win rate

Table 1. Optimal values for the important hyper-parameters

Experimental Results

This section begins with an introduction to the evaluation metrics and the baseline methods, and compares RCPacing to the baselines through offline and online experiments.

Evaluation Metrics

- **Delivery rate** is defined as the ratio of allocated impressions to the total budgets of the advertisers:

$$delivery\ rate = \frac{\sum_t \sum_j \tilde{x}_j^{(t)}}{\sum_j B_j} \quad (30)$$

- **Unsmoothness index** (UI) measures the deviation between the actual and expected budget consumption:

$$unsmoothness = \frac{1}{M} \sum_{j=1}^M \sqrt{\frac{1}{T} \sum_{t=0}^{T-1} (\tilde{x}_j^{(t)} - \rho_j)^2} \quad (31)$$

- **Average CTR** reflects the quality of impressions, is calculated as the ratio of clicks to the total impressions:

$$CTR_{avg} = \frac{\sum_t \sum_j v_j^{(t)} \tilde{x}_j^{(t)}}{\sum_t \sum_j \tilde{x}_j^{(t)}} \quad (32)$$

Baseline Methods

We compare RCPacing with the following four methods: 1) **DMD** (Balseiro, Lu, and Mirrokni 2020) is a Lagrangian dual-based online allocation framework that maximizes revenue while adhering to resource constraints by adjusting their virtual bids. 2) **Smart Pacing** (Xu et al. 2015) is a control-based method proposed to achieve smooth delivery and optimal performance by probabilistic throttling. 3) **AUAF** (Cheng et al. 2022) is a dual-based method that optimizes delivery rate and impression quality with a fixed smoothness term. The dual variables are updated by feedback control algorithm to ensure fairness. 4) **PDOA** (Zhou et al. 2021) solves online matching in dynamic environments with experts and meta-algorithm. It achieves smoothness by bid modification.

Offline Evaluation

Datasets We construct a large-scale industrial dataset² by collecting real-world ad-serving data from our display advertising system, which consists of 600K impressions and

²Dataset and the code for all methods are available in <https://github.com/danifree/RCPacing>.

Method	Unsmoothness	Delivery Rate (%)	CTR (%)
DMD	15.71 \pm 1.46	100.0 \pm 0.0	5.39 \pm 0.02
Smart	10.52 \pm 1.04	95.9 \pm 0.9	7.88 \pm 0.37
AUAF	12.95 \pm 1.29	100.0 \pm 0.0	5.14 \pm 0.01
PDOA	15.70 \pm 2.27	100.0 \pm 0.0	6.06 \pm 0.27
RCPacing	6.37 \pm 0.72	99.8 \pm 0.1	7.46 \pm 0.44

Table 2. Offline evaluation results

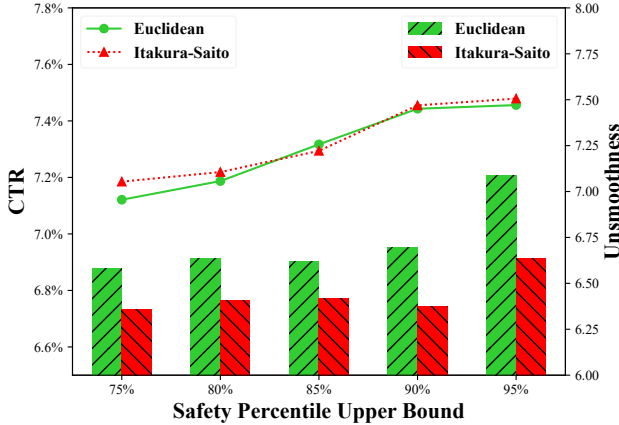


Figure 7: The ablative analysis.

300 GD ads. The impressions are evenly distributed across 50 time periods. The CTR values predicted by a DNN are reserved to measure the impression quality.

Implementation Details Table 1 provides a summary of the optimal values for the important hyper-parameters.

Evaluation Results In order to exclude the influence of accidental factors, we randomly scale the budget of GD ads by a factor ranging from 0.8 to 1.2, and calculate the mean and standard deviation across 50 rounds. As shown in Table 2, Smart Pacing achieves the highest average CTR, but its low delivery rate is inappropriate for GD allocation, which results in publishers being penalized for unsatisfied demand. RCPacing demonstrates a significant reduction in UI, with a 59.4% and 50.8% improvement compared to PDOA and AUAF, respectively. Furthermore, it delivers superior CTR performance, achieving a 23.1% and 45.1% increase compared to PDOA and AUAF.

Ablative Analysis We focus on UI and CTR since the delivery rates of the variants are very close to 100%.

- **The impact of percentile upper bound:** A higher safety percentile upper bound (P_{ub}) allows advertisers to filter low-quality impressions more effectively, but it also raises the risk of fluctuations. As demonstrated in figure 7, RCPacing has higher CTR when using a larger P_{ub} , but there is a 4.14% increase in unsmoothness when P_{ub} is changed from 90% to 95% (Itakura-Saito divergence).
- **The impact of different divergence:** As mentioned earlier, a modified Itakura-Saito divergence helps alleviate the issue of high fluctuations in the high percentile range.

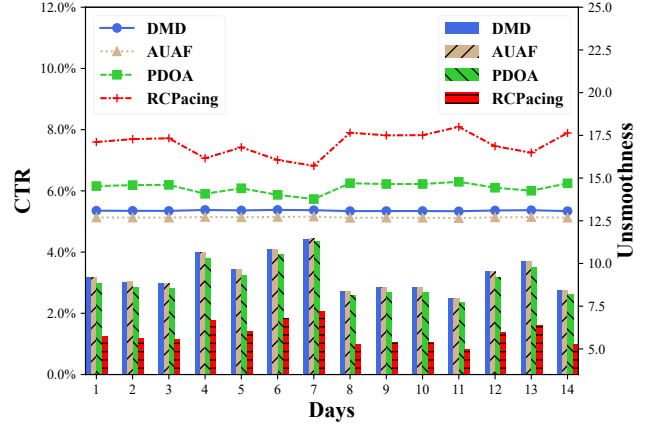


Figure 8: The online evaluation results.

Figure 7 illustrates that the proposed Itakura-Saito divergence provides better UI especial when P_{ub} is high (e.g., a 3.74% improvement in smoothness when P_{ub} equals 90%), while the average CTR is comparable to that of the Euclidean divergence.

Additional ablative analysis can be found in the appendix.

Online Evaluation

Implementation Details In order to evaluate the performance of RCPacing in an online environment, we conduct A/B testing on our Taobao brand advertising platform for a continuous period of two weeks. Since the delivery rate for Smart pacing is too low for GD allocation, we only compare our method with DMD, AUAF, and PDOA.

Evaluation Results As the delivery rates of all methods exceed 99.5%, we concentrate on the other two metrics in figure 8, RCPacing outperforms all the baselines. For example, compared with PDOA, our method achieves a 35.3% and 23.4% improvement in UI and CTR, respectively.

Conclusion

GD contracts are a crucial source of revenue for large publishers. This paper presents a robust percentile risk-constrained pacing framework designed from the perspective of a publisher. RCPacing achieves smooth and optimal allocation for GD campaigns by leveraging its compatibility with the guaranteed allocation mechanism. Our analysis presents the relationship between non-smooth risks and percentile of dual variables, and RCPacing is designed to constrain dual variables within the low-risk region. Adaptive gradient clipping and modified Bregman divergence techniques are also employed to achieve more stable update of dual variables. We also illustrate the trade-off and flexible control over smooth and optimal allocation in online matching. Our experimental evaluations on real-world A/B testing demonstrate that RCPacing outperforms other compared methods and it has been widely deployed in Taobao display advertising system.

References

- Agarwal, D.; Ghosh, S.; Wei, K.; and You, S. 2014. Budget pacing for targeted online advertisements at linkedin. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1613–1619.
- Balseiro, S.; Lu, H.; and Mirrokni, V. 2020. Dual mirror descent for online allocation problems. In *International Conference on Machine Learning*, 613–628. PMLR.
- Banerjee, A.; Merugu, S.; Dhillon, I. S.; Ghosh, J.; and Laferty, J. 2005. Clustering with Bregman divergences. *Journal of machine learning research*, 6(10).
- Bharadwaj, V.; Chen, P.; Ma, W.; Nagarajan, C.; Tomlin, J.; Vassilvitskii, S.; Vee, E.; and Yang, J. 2012. Shale: an efficient algorithm for allocation of guaranteed display advertising. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1195–1203.
- Chen, P.; Ma, W.; Mandalapu, S.; Nagarjan, C.; Shanmugasundaram, J.; Vassilvitskii, S.; Vee, E.; Yu, M.; and Zien, J. 2012. Ad serving using a compact allocation plan. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, 319–336.
- Chen, X.; Wu, S. Z.; and Hong, M. 2020. Understanding gradient clipping in private SGD: A geometric perspective. *Advances in Neural Information Processing Systems*, 33: 13773–13782.
- Cheng, X.; Liu, C.; Dai, L.; Zhang, P.; Fang, Z.; and Zu, Z. 2022. An Adaptive Unified Allocation Framework for Guaranteed Display Advertising. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, 132–140.
- Dai, L.; Zu, Z.; Wu, H.; Wang, L.; and Zheng, B. 2023. Fairness-aware Guaranteed Display Advertising Allocation under Traffic Cost Constraint. In *Proceedings of the ACM Web Conference 2023*, 3572–3580.
- Fang, Z.; Li, Y.; Liu, C.; Zhu, W.; Zheng, Y.; and Zhou, W. 2019. Large-scale personalized delivery for guaranteed display advertising with real-time pacing. In *2019 IEEE International Conference on Data Mining (ICDM)*, 190–199. IEEE.
- Gao, C.; Lei, W.; He, X.; de Rijke, M.; and Chua, T.-S. 2021. Advances and challenges in conversational recommender systems: A survey. *AI Open*, 2: 100–126.
- Huang, Z.; and Shu, X. 2021. Online stochastic matching, poisson arrivals, and the natural linear program. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 682–693.
- IAB. 2023. Internet Advertising Revenue Report. https://www.iab.com/wp-content/uploads/2023/04/IAB_PwC_Internet_Advertising_Revenue_Report_2022.pdf. Accessed: 2023-04.
- Lee, K. C.; Jalali, A.; and Dasdan, A. 2013. Real Time Bid Optimization with Smooth Budget Delivery in Online Advertising. *arXiv e-prints*.
- Liu, M.; Yue, W.; Qiu, L.; and Li, J. 2020. An effective budget management framework for real-time bidding in online advertising. *IEEE Access*, 8: 131107–131118.
- Mehta, A. 2012. Online Matching and Ad Allocation. *Foundations and trends in theoretical computer science*, (8-4).
- Mehta, A.; Saberi, A.; Vazirani, U.; and Vazirani, V. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5): 22–es.
- Nuara, A.; Trovò, F.; Gatti, N.; and Restelli, M. 2022. Online joint bid/daily budget optimization of internet advertising campaigns. *Artificial Intelligence*, 305: 103663.
- Sakia, R. M. 1992. The Box-Cox transformation technique: a review. *Journal of the Royal Statistical Society Series D: The Statistician*, 41(2): 169–178.
- Wang, X.; Tan, B.; Guo, Y.; Yang, T.; Huang, D.; Xu, L.; Freris, N. M.; Zhou, H.; and Li, X.-Y. 2022. CONFLUX: A Request-level Fusion Framework for Impression Allocation via Cascade Distillation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4070–4078.
- Wu, D.; Chen, C.; Chen, X.; Pan, J.; Yang, X.; Tan, Q.; Xu, J.; and Lee, K.-C. 2021. Impression Allocation and Policy Search in Display Advertising. In *2021 IEEE International Conference on Data Mining (ICDM)*, 749–756. IEEE.
- Xu, J.; Lee, K.-c.; Li, W.; Qi, H.; and Lu, Q. 2015. Smart pacing for effective online ad campaign optimization. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2217–2226.
- Zhou, G.; Mou, N.; Fan, Y.; Pi, Q.; Bian, W.; Zhou, C.; Zhu, X.; and Gai, K. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 5941–5948.
- Zhou, Y.-H.; Hu, P.; Liang, C.; Xu, H.; Huzhang, G.; Feng, Y.; Da, Q.; Wang, X.; and Zeng, A.-X. 2021. A Primal-Dual Online Algorithm for Online Matching Problem in Dynamic Environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 11160–11167.