Practical Budget Pacing Algorithms and Simulation Test Bed for eBay Marketplace Sponsored Search

Phuong Ha Nguyen, Djordje Gligorijevic, Arnab Borah, Gajanan Adalinge and Abraham Bagherjeiran
eBay Inc.
San Jose, CA, USA
{hpnguyen,dgligorijevic,aborah,gadalinge,abagherjeiran}@ebay.com

ABSTRACT

Sponsored search program in online marketplaces is routinely offered to sellers allowing them the opportunity to enhance the visibility and performance of their items. Optimizing for different goals such are clicks or conversions, advertisers can run campaigns provided a budget constraint. However, without any control on the spend, campaigns with smaller budgets can run too briefly, failing to reach regions of high quality traffic. Moreover, the same lack of spending control can benefit a few dominating sellers, inducing a lack of competition that negatively affects the majority of sellers, the marketplaces' revenue, and users experience. Budget pacing technique is a common tool used to control spend of ads campaigns that can tackle aforementioned challenges in a principled manner providing benefits to sellers, the online marketplace and users. In this paper we propose a simulation test bed based on real traffic of sponsored search program at eBay for accurate and safe evaluation of different budget pacing algorithms. We study several simple budget pacing algorithms, characterizing their effect under complex environmental constraints. As an important contribution, we describe an efficient test bed for offline simulations and propose a new simple and efficient budget pacing algorithm based on the campaigns' remaining budget which can achieve improvements in many business metrics compared to the production benchmark.

CCS CONCEPTS

• **Information systems** → **Sponsored search advertising**; *Computational advertising*; • **Applied computing** → *Electronic commerce*.

KEYWORDS

online advertising, sponsored search, budget pacing

ACM Reference Format:
Phuong Ha Nguyen, Djordje Gligorijevic, Arnab Borah, Gajanan Adalinge and Abraham Bagherjeiran. 2018. Practical Budget Pacing Algorithms and Simulation Test Bed for eBay Marketplace Sponsored Search. In *Woodstock '18*. ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Many online marketplaces offer advertising programs to their sellers, with the most notable one being sponsored search. Such advertising program allows sellers to bid for better ranking positions or for preferential slots for their items. This in turn enhances performance of sponsored items through an optimized exposure or user engagement, depending on seller's strategies, with the most common target historically being pay-per-click (PPC) [3]. In order to run sponsored search campaigns, sellers need to define daily budgets, organize items into groups for which they specify targeting, and target and/or maximum bids. Allocating small budgets, or high maximum bids can cause campaigns to deplete their budget too fast and stop early. Moreover, the spending pattern of the campaigns' money is an important signal to advertisers that builds trust and partnership [7], notably, smoothness and/or repeating patterns of budget depletion may be key indicators for advertisers [9].

Thus, advertising platforms may offer tools such as budget pacing [5, 10] to sellers, whose purpose is to control the pace and manage how the campaigns' budget is being depleted throughout the day. Most common implementation of budget pacing optimizes for uniform spending throughout the day [1], but other strategies such as spending in high traffic or high response rate regions [7], as well as dayparting (where sellers define time slots where they want their budgets to be spent) are also common.

The expected risks of not having a budget pacing module may span: 1) an unhealthy competition as higher bid ads may not have enough budget left later in the day, resulting in lower clearing prices and user engagement, and 2) advertisers would miss out on good ad opportunities later in the day. On the other hand, introducing a budget pacing module fosters: 1) healthy ad competition that results in higher yield for the ad platform, 2) advertisers' ROAS as more ads can be shown where users are more likely to engage and 3) user retention is higher as they would be served more valuable ads. Budget pacing can have a profound effect on the entire advertising program. Under the right conditions, introduction of budget pacing can bring benefits to all sides.

To tackle the aforementioned challenges in a safe manner, supply and demand behavior as well as all constraints of the system have to be known and understood. Designers of budget pacing systems ought to understand the conditions and environment needed for successful implementation of existing algorithms. For instance, in competitive ad environments with limited supply, advertisers can allocate large budgets without concerns of premature depletion.

Simple budget pacing techniques have proven to be very effective at tackling the control over the spending [9, 11, 12], and as they are typically straightforward to analyze, understand and efficiently implement in real-world systems [1]. Moreover, an important appeal of simple budget pacing techniques stems from the ability to easily conduct deep system triage when needed.

Finally, testing the efficiency of budget pacing techniques can be very expensive and risky if they are exposed to real-word traffic, as the sponsored search is highly nonlinear program in terms of interactions of its components. To that end we have developed a budget pacing test bed which acts as our domain analysis framework where a range of budget pacing techniques can be evaluated quickly and safely, and hyperparameter fine-tuning can be done.

Summarizing the key contributions of this work with respect to above discussed questions and challenges to build practical working budget pacing system in a is listed as follows:

- A detailed construction framework of test bed of online advertisement is presented.
- Based on the real traffic of sponsored search program at eBay, a test bed is implemented, characterized and evaluated.
- Given the constraints of environment, we explain the importance of greedy budget pacing strategies as an important design approach and based on it, we evaluate practical budget pacing algorithms and propose reasonable extensions which allow us to achieve key business goals.
- These budget pacing algorithms are implemented in our test bed and then their performance are analyzed in detail and evaluated with our proposed metrics.

2 BACKGROUND AND RELATED WORK

In this study we propose a general sponsored search test bed of high integrity, which makes generic assumptions that are common in the domain, and requires a distribution of users' search requests as its main input. Given the main application being eBay's sponsored search program, we describe it first in this section. Following, we discuss related work in budget pacing techniques that were considered for experimentation in the designed test bed.

2.1 Background on sponsored search

Each time a user makes a new search request, sponsored search program would retrieve eligible and relevant items to participate in an internal ads auction for a chance to be displayed to the user in ad slots. At this stage, other advertisements may join the ads auction creating an additional, external, competition which creates an important aspect to consider when designing controllers for ads campaigns. The score with which all sponsored items are ranked in an ad auction is called ad expected value v and it is calculated for an item i, given seller provided bid b_i and response probability (in our case click-through-rate pCTR) p_i as $v_i = b_i \times p_i$. We simulate item allocation with a multi-slot generalized second-price auction for allocating items across the search result page [4]. In case of user response (click), the sellers' campaign will be charged clearing price c_i based on second-price auction mechanism using the ad expected value of the first lower $(r+1)$ ranked item j:

$$c_i^r = \frac{v_i^{r+1}}{p_i^r} = \frac{b_j^{r+1} \times p_j^{r+1}}{p_i^r}, \tag{1}$$

logically capped so that $c_i^r \leq b_j^r$. Ad campaigns would run throughout the day given their budgets B_i's, deplete the budget after each user response, and if the budget is completely depleted stop whenever that happens.

Interesting observation is that no pacing may be a good solution when the ads competition is high enough as it always guarantees that for a search query the highest and second highest ranking sponsored items among not-yet-depleted budget campaigns are always present in the auction, thus maximizing the revenue.

2.2 Related work on budget pacing methodologies

Many different budget pacing solutions have been proposed in the past ranging from solutions based on bipartite graph allocation methodologies [6, 10], based on business metrics [7], control theory [8] or more empirical solutions [1, 9, 12]. Moreover, all of these solutions can be separated into two general approached: hard budget pacing or throttling (preventing an ad to be allocated) and soft budget pacing (altering bid value for an ad used for the allocation). In practice, throttling is a more appealing approach for young ad platforms as its effect is clearly distinct which better supports system triage and moreover, it has been shown that advertisers tend to prefer throttling approaches over bid altering approaches in sponsored search [7]. In this study we completely focus on throttling-based budget pacing approaches, noting that approaches discussed could easily be implemented as bid altering approaches as well.

Another important aspect that new ad program cope with is that most campaigns have small budgets or small number of expected clicks, often due to dominating competition in the environment. Therefore, uniform budget pacing (spending money equally during the day) is often preferred over other solutions such are traffic-based or performance-based budget pacing or dayparting [9, 11, 12].

Each approach makes slightly different assumptions on the ad opportunity traffic that makes it difficult to evaluate them without exposing them to the real traffic which is expensive and risky. To enable safe and efficient testing of budget pacing solutions we design and present a traffic-based sponsored search test bed below.

3 TEST BED

In this section, based on above described sponsored search implementation flow, we explain the implementation of the test bed based on the real traffic collected from logs. The test bed runs on real daily logs utilizing historical information of ad opportunities, thus allowing us to simulate budget pacing algorithms with high integrity. As such, the test bed is suitable for a different types of ad programs such as display/native platforms or small ad exchanges.

3.1 Sketch of implementation flow of test bed

Data Model. The budgets of sponsored search campaigns are typically reset once a day. We, thus, focus on simulating 1-day traffic by partitioning it into 1440 1-minute datasets where each dataset contains all search requests eligible for displaying sponsored items. This approach can be applied to any other budget reset strategy. Achieving a real time budget pacing signal estimation is the optimal solution, however, this can lead to significant computational burden on the real world system potentially leading to system instabilities, while near-real-time updates of 1-minute resolution provide a satisfactory trade-off between time complexity and modelling accuracy.

Targeting-based retrieval. In practice, retrieval based on targeting input is a complex and time consuming process involving sorting by relevance and other important values. To simulate the targeting-based retrieval task, for each search query, we build its own targeting set from logged recall sets.

The bid value. One item in a campaign may have different bid values depending on its targeting (i.e. keywords) strategy, where keywords may overlap. Therefore, to calculate *ad expected value*, we prioritize largest bid ad group similarly as production system.

The pCTR. In practice, sponsored items would have pCTR calculated depending on the context, however, during simulation this process can be very time consuming. In order to address the challenges of computing pCTR scores for less frequent items and sponsored items from other ad programs, we resort to choosing from item-query level, item-level, or slot-level pCTR, in the given order.

Simulating click behavior of users. We simulate the click behavior of users using click value generator (elaborated in Section 3.2) that is based on the concept of counterfactual modelling.

Evaluation metrics. In order to characterize the quality of the test bed we focus on smoothness of spend throughout the day, but also on other key performance metrics: the number of impressions, the number of clicks, CTR, total ad revenue and cost-per-click (CPC).

3.2 Probabilistic click value generation

Opposed to a naive random click generation strategy, we exploit the fact that the log of each search query contains the information about the $pCTR$ and click values C of sponsored items shown at a given slot. We declare this information as $pCTR_1$ and click value C_1, while the generator will use $pCTR_2$ from a newly generated ranking to produce click value C_2 in a manner described below.

Case 1. $pCTR_2 < pCTR_1$ and $C_1 = 1$.

- Generate a random number R in $[0, 1]$
- If $R < pCTR_2/pCTR_1$, then $C_2 = 1$ otherwise $C_2 = 0$.

Case 2. $pCTR_2 < pCTR_1$ and $C_1 = 0$. C_2 is assigned to 0.

Case 3. $pCTR_2 > pCTR_1$ and $C_1 = 1$. C_2 is assigned to 1.

Case 4. $pCTR_2 > pCTR_1$ and $C_1 = 0$.

- Generate a random number R in $[0, 1]$
- If $R < (1 - pCTR_2)/(1 - pCTR_1)$, then $C_2 = 0$. Else, $C_2 = 1$.

This provides a counterfactually generated user response under altered user response due to a treatment tested.

3.3 Quality of the test bed

In this section, we evaluate the quality of the test bed of sponsored search ads by computing the number of impressions, the number of clicks, CTR, spending and cost-per-click for the whole day when we run the simulator without any treatment algorithms, accounting for natural noisiness of the system and any external competition. Another run of the simulator, with a naive click value generation (based on estimated pCTR) is generated to show the advantages of the proposed solution. These results are then compared to historical data to gives us an idea about the quality of the test bed.

The global evaluation of test bed is provided in Table 1. A key property of the simulator is that the overall behavior of simulated traffic should match the real traffic. While the gaps in the simulation results compared to the real traffic can come as artifacts from the approximations of the test bed described above, the provided results show that the key components of the simulator: probabilistic response generation brings a significant improvement across the key metrics. Moreover, the provided generated time series results of clicks, spend and clearing price (detrended to preserve sensitive information) follow the combination of plots of impressions and CTR, we present in Figures 1a, 1b, 1c, 1d and 1e.

	Total Impressions	Total Clicks	Total Spend	CTR	CPC
Naive	**8.5%**	-10.12%	-14.91%	-17.16%	-5.33%
New	13.5%	**-5.15%**	**-4.91%**	**-16.16%**	**+1.08%**

Table 1: Evaluation of the test bed according to changes in key metrics compared to the original data collected in logs when naive and the proposed click value generators are used.

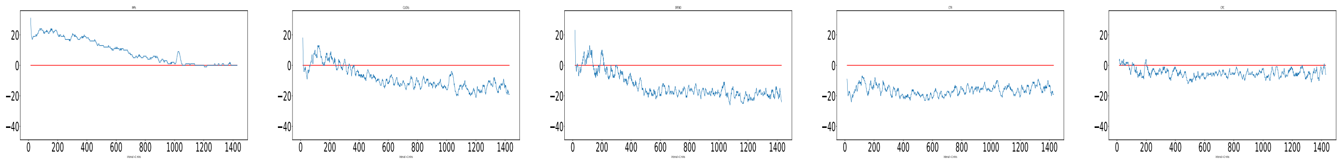
We noted that "no pacing" simulation, called reset time at 0-th interval (RST_0), has the same trend as the original traffic for all curves. Using these time series, we validated the test bed by calculating the relative difference between RST_0 and real traffic, shown in the aforementioned figures, and discovered that the difference is acceptably small. Most importantly, the simulated RST_0 is used as the baseline for all approaches tested, amortizing any discrepancy created due to the test bed artifacts.

3.4 Important notes on the simulations using the test bed

The proposed simulator is designed to test different use cases a sponsored search program can face such as having external competition, working in an environment with different budget reset strategies, using budget pacing or no pacing as control, and other. In the experiments we present in this study, we simulate scenario of applying budget pacing techniques on the system that previously did not control budget spend, while having a competition from an external advertising program. This particular scenario has a caveat of a potential reduction of total impressions. Moreover, throttling campaigns for some ad opportunities does not mean that there will be more opportunities in the future, so for simple budget pacing techniques greedy strategies may be preferable, which we discuss below. Finally, campaigns with small budgets may not benefit much from a duration improvement.

4 BUDGET PACING ALGORITHMS

A large volume of papers discuss about the budget pacing algorithms [1, 6, 7, 9–12] given business goals and constraints of competition resources (e.g., campaign budgets, bids), system resource (e.g., supporting complex computations), the information of environment (e.g., competition rate, winning rate, traffic curve, the performance of the system). While the majority of the approaches assume that campaign budgets are large, that there are only a few polices in the ad program, or that there is no competition, we developed a simulator that may account for all such scenarios and their variations, even including fine tailored policies for specific campaigns when needed. We discuss budget pacing algorithms suitable for implementation through a framework given in Algorithm 1 in the following context used for eBays sponsored search program.



(a) IMS(RST_0 vs org) **(b) CLKS(RST_0 vs org)** **(c) SPEND(RST_0 vs org)** **(d) CTR(RST_0 vs org)** **(e) CPC(RST_0 vs org)**

Figure 1: Measured difference between the simulated control annotated as RST_0 vs the real system as diff $= 100\% \times (RST_0 -$ real traffic$)/$real traffic.

- Most campaigns have small budgets or get small number of clicks in practice due to high competition of the environment.
- The budgets are reset once per day – no pacing. The algorithms in line are called RST_x where x is the reset time considered and is equal to 0 or 750 in this study.
- First group of budget pacing techniques considers only the remaining budget of a campaign. We discuss this scenario in Section 4.2, and call approaches *Budget* in this paper.
- Second group of budget pacing techniques considers information about campaigns' remaining budgets and the remaining time. We discuss this scenario in Section 4.3, and call approaches $BudgetTime$ in this paper.
- Third group of budget pacing techniques considers information about campaigns' remaining budgets and predicted information about the remaining spending opportunity. The predicted information about the spending opportunity is calculated based on historical data. We discuss this scenario in Section 4.4, and call approaches *ClkOp* in this paper.

If getting as many clicks as possible is the majority's interest and this is one of program's high priorities, then it seems *the optimal pacing strategy is greedy one, i.e., try to let the campaigns join the auctions whenever they can.* The reason is as follows. If the campaigns do not know anything about the environment except the fact that the competition is very high and they want to collect as many clicks as possible, then the best strategy is just always join the auctions whenever they can.

4.1 Daily reset algorithm

Daily budget reset strategy is typically the first budget control strategy implemented in advertising programs, thus, in our study it serves as the main baseline. This option is referred to as RST_0. As we discussed in Section 3.4, this approach is a good option because all campaigns can guarantee maximum number of impressions. Moreover, daily reset algorithm also maximizes CPC (greedy revenue maximization) when second auction logic is used as no support is being throttled. The negative point of this approach is that it does not attempt to smoothen the budget depletion, but maximizes the competition at the time of reset, which, due to early budget depletion results in an unhealthy competition across the day.

Finally, depending on the traffic curve and user response curve of sponsored search program and its business goals, we may search for the optimal reset time. After doing grid search, reset at 750th minute (RST_{750}) provided good results in the number of clicks, total revenue, CTR and CPC.

Below, we discuss several solutions for throttling-based budget pacing where they would, for each ad call, generate the throttling signal that will provide a probability of item joining the ads auction.

4.2 Remaining budget based pacing algorithm

This algorithm only uses remaining budget to pace the spend. Motivated by the budget pacing algorithm proposed in [10], we provide a modification described in Algorithm 1 to apply it to the sponsored search program at eBay.

4.3 Remaining budget and time based pacing algorithm

This algorithm requires the information about the remaining budget and remaining time in day with the current pacing rate θ. We modify the vanilla budget pacing algorithm proposed in [7] as described in Algorithm 1, by enforcing a more greedy spending strategy.

In this approach, for a given campaign with initial budget B, the target total spend of the campaign up to the moment t is described by $\frac{B}{T} \times t$, $t = 0, \ldots, 1439$, where T is the duration the campaign wants to complete its budget. Therefore, if $ratio$ is larger than 1, then the campaign is overspending (spend more than what it should at time t), and should be slowed down by multiplying with the factor f_{over} which is smaller than 1. When θ is small enough (i.e., f_{bound}), then we do not want to reduce it any more by setting it to a small value 0.001. The main motivation is that we still adopt greedy strategy, the campaigns do not completely stop spending. Given our offline experimentation, we set $f_{over} = 0.8$ and $f_{bound} = 0.01$. When $ratio$ is smaller than 1, it implies the campaign is under-spending, the pacing rate $\theta = 1$, as we want it to join auctions immediately. Based on the simulated logs for that campaign, we can calculate the true spend of the campaign S_{sim}. The closer S_{target} and S_{sim}, the better pacing algorithm in terms of spreading the spend over the time.

4.4 Remaining click opportunity based pacing algorithm

This algorithm requires the information about the remaining budget and spend opportunities. It is a modification of the vanilla budget pacing algorithm proposed in [7] as described in Algorithm 1.

In the vanilla algorithm proposed in [7], the authors train a model based on historical data of campaigns' spends to predict the possible remaining spend of each campaigns in case campaigns have no budget restrictions. The efficiency of the algorithm really depends on quality of the forecasting model. Training standard forecasting models such as autoregressive logistic regression or

Algorithm 1 Pacing algorithms for throttling spend

procedure PACING($methodName$)
  if $methodName == RST$ **then**
    $\theta = 1$
  end if
  if $methodName == Budget$ **then**
    Input: $budget_{rem}$ and $budget_{orig}$
    $x = budget_{rem}/budget_{orig}$
    $\theta = \Psi(x)/\Psi(1)$ where $\Psi(x) = 1 - e^{-x}$
  end if
  if $methodName == BudgetTime$ **then**
    Input: $budget_{rem}$,$budget_{orig}$,$time_{rem}$, $time_{orig}$
    Algo Parameters: f_{over}, f_{bound} and $bound$
    $rB = budget_{rem}/budget_{orig}$
    $rT = time_{rem}/time_{orig}$
    $ratio = rB/rT$
    if $ratio \geq 1$ **then**
      $\theta = \theta \times f_{over}$
    else
      $\theta = 1.0$
    end if
    if $\theta \leq f_{bound}$ **then**
      $\theta = 0.001$
    end if
  end if
  if $methodName == ClkOp$ **then**
    Input: $budget_{rem}$, $maxBid$ and $ClickOpp_{rem}$
    $\theta = budget_{rem}/(maxBid \times ClickOpp_{rem})$
  end if
  Generate a random number R in $[0, 1]$
  if $R \leq \theta$ **then**
    Do not throttle spend
  else
    Throttle spend
  end if
end procedure

ARIMA [2] for each campaign is very challenging as the data is very sparse for vast majority of campaigns, and results in high variance predictions that cause instability of budget pacing system when applied. Given these restrictions, we develop simple predictor using two simulations as follows. First, we run a simulation in which the campaigns have no budget restriction. In this case, we can collect information about the remaining spend opportunities (i.e., the number of clicks) campaigns can get. Next, we run a simulation in which the campaigns have budget restriction, this is equivalent to RST_0. We collect information about remaining spend opportunities (i.e., the number of clicks) of *all* campaigns which can get spent. Intuitively, we run two extreme cases and by combining the results of them, we can have a conservative prediction on remaining spend opportunities (i.e., the number of clicks).

5 EXPERIMENT AND ANALYSIS

In this section, we present the experiment results and provide detailed analysis of our proposed budget pacing algorithms.

	RST750	Budget	BudgetTime	ClkOp
PE	-35.40%	+3.10%	**-52.20%**	+1.70%
WPE	+32.30%	-10.40%	**-39.50%**	+2.20%
totalImps	-8.50%	-3.20%	-3.90%	**-2.30%**
totalClks	**+8.10%**	+0.70%	+5.60%	+3.90%
totalSpends	**+7.60%**	+0.60%	+4.10%	+3.80%
CTR	**+18.30%**	+4.00%	+9.90%	+6.40%
CPC	-0.50%	0.00%	-1.40%	0.00%

Table 2: Results obtained by different budget pacing algorithms compared to the RST_0 baseline measured by pacing error scores and business metric scores.

5.1 Experiment setup

As described above, the test bed consists of 1440 different 1-minute datasets. Each dataset consists of real search queries with the eligible ad slots. Before working with the t-th dataset, the information of the campaigns' initial and remaining budgets, and the remaining time are obtained for the t-th dataset. The budget pacing throttling threshold θ is calculated by algorithms described in Section 4 and used as a probability of participating in an auction for the t-th dataset. After the t-th dataset simulation, the remaining budgets of campaigns are updated based on the simulated spend.

There are three groups of budget pacing strategies considered: 1) greedy (RST_0, RST_{750}), 2) based on budget and time, ($Budget$ and $BudgetTime$) and 3) based on forecasted opportunities ($ClkOp$). RST_0 is chosen as benchmark for comparison purpose.

5.2 Evaluation Metrics

To evaluate the impact of different budget pacing algorithms we measure key business metrics discussed in Section 3.1. In addition as key metrics we measure the system-level spend *pacing error (PE)* (measuring smoothness of spend over a day) metric defined as:

$$\text{PE} = (1/1440) \times \sum_{j=1}^{1440} \frac{|pS_j - pT_j|}{pT_j},$$

with pS_j and pT_j as the fraction of total spend and traffic in j-th interval. Moreover, we also measure campaign-weighted *Weighted Pacing Error (WPE)*:

$$WPE = \sum_{i=1}^{n} \left(\sum_{t=1}^{1440} \left(\left| CS_{i,t} - t \times \frac{S_i}{1440} \right| \right) \times \frac{S_i}{S} \right),$$

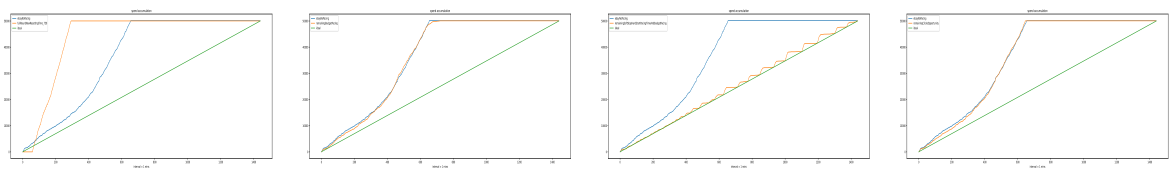
where $CS_{i,t}$, S_i and S are the accumulated spend of a campaign i up to t-interval, the targeted spend of campaign i up to t-interval, the total spend of campaign i and the total spend of the whole system in one day, respectively.

Values of all metrics are reported as changes relative to RST_0 strategy, and are presented in Table 2.

5.3 Analysis

RST_{750} vs RST_0. This pacing is not used to smooth out the spends of campaigns. Precisely, the campaigns join any auction as long as their budget is not empty.

The reset time is at 750-th minute gives us the best results in terms of number of clicks, revenue and CTR. This means the environment offers a very good spending time for all campaigns.



(a) $rank_1(RST_{750}$ vs $RST_0)$ (b) $rank_1(Budget$ vs $RST_0)$ (c) $rank_1(BudTime$ vs $RST_0)$ (d) $rank_1(ClkOp$ vs $RST_0)$

Figure 2: Comparison of Pacing Error Curves between 1) RST_{750} (orange) vs RST_0 (blue) vs Ideal (green) 2) $Budget$ (orange) vs RST_0 (blue) vs Ideal (green) 3) $ClkOp$ (orange) vs RST_0 (blue) vs Ideal (green) 4) $Bud(get)Time$ (orange) vs RST_0 (blue) vs Ideal (green)

Moreover, the campaigns have a full budget at 750-th minute and thus, we see a high peak of impressions and clicks at that moment. Given the fact that most CPC campaigns have a small amount of clicks, they quickly deplete their budgets after a few hours.

The result of RST_0 and RST_{750} shown in Table 2 confirms that if the environment has high competition and many CPC campaigns with small budgets, then greedy strategy seems to be a good one.

To understand the impact of RST_0 and RST_{750} on the spreading of the spend of CPC campaigns, we focus on top-1 campaigns which have highest number of clicks when working with RST_0 (benchmark). Figure 2a shows the pacing error curves of the top-1 campaign when working with RST_0 (blue lines) and RST_{750} (orange lines). The ideal spending curve (green line) is the benchmark. Clearly, the error curves of RST_{750} is much worse than RST_0 or the campaigns spend much quicker when they work with RST_{750}.

Budget vs RST_0. This pacing method is based on the remaining budget information of campaigns. Given the state of CPC program, the impact of this algorithm on CPC program is very similar to RST_0, as seen through smallest differences in Table2.

If the campaigns have big budgets, then the ratio $\frac{budget_{rem}}{budget_{orig}}$ should slowly decrease. But the behavior of small budget campaigns does not much change too much, i.e., their budgets will still be quickly depleted. Combined effect, of the two types of campaigns will result in the lack of competition and a faster budget depletion of large campaigns, whose behavior will look similar to the one without any budget pacing applied. This is clearly seen in pacing error curve of the top campaign shown in Figure 2b.

BudgetTime vs RST_0. This pacing is based on the remaining budget and time information of campaigns. According to the Table 2 we can see that the algorithm has a strong impact on the CPC program, where in terms of business metrics, from the non-greedy spending approaches considered in this paper, the proposed $BudgetTime$ shows best results. The best pacing error results imply that $BudgetTime$ method variant can smooth out the spend throughout the day effectively. The error curves of top-1 campaign shown in Figure 2c show that the proposed algorithms work very well, i.e., the error curves of top-1 campaigns of the budget pacing algorithms nearly match the ideal ones.

ClkOp vs RST_0. This pacing is based on the remaining budget information of campaigns and estimated remaining spend opportunities of campaigns. In the situation where ads competition is not very large, this algorithm may appear very similar to RST_0 in terms of results (see Table 2 and Figure 2d). The reason for this is the bias of the predictions towards dominating campaigns that have had plenty of ad opportunities during the day, suffocating smaller campaigns for whom the prediction of ad opportunities would be underpredicted. The success of $ClkOp$ approaches, thus, heavily depends on the high volume of competition [7].

6 CONCLUSION

In this paper, we described a detailed construction framework of simple but efficient test bed for budget pacing algorithms. We study some simple yet efficient budget pacing algorithms for online advertising program with a competitive environment with plenty small budgets campaigns. Our theory analysis and experiments show that greedy strategy is quite a good option which led to an exploration of simple budget pacing algorithms given such environment. Finally, with our empirical experiments and characterization of the system, we proposed an approach we called $BudgetTime$ that achieved the best balance in terms of pacing error and business metrics impact.

REFERENCES

[1] Deepak Agarwal, Souvik Ghosh, Kai Wei, and Siyu You. 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.

[2] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.

[3] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review* 97, 1 (2007), 242–259.

[4] Benjamin Edelman and Michael Schwarz. 2010. Optimal auction design and equilibrium selection in sponsored search auctions. *American Economic Review* 100, 2 (2010), 597–602.

[5] Jon Feldman, Shanmugavelayutham Muthukrishnan, Martin Pal, and Cliff Stein. 2007. Budget optimization in search-based advertising auctions. In *Proceedings of the 8th ACM conference on Electronic commerce*. 40–49.

[6] Joaquin Fernandez-Tapia. 2015. Optimal budget-pacing for Real-time Bidding.

[7] Chinmay Karande, Aranyak Mehta, and Ramakrishnan Srikant. 2013. Optimizing budget constrained spend in search advertising. In *Proceedings of the sixth ACM international conference on Web search and data mining*. 697–706.

[8] Niklas Karlsson and Jianlong Zhang. 2013. Applications of feedback control in online advertising. In *2013 American control conference*. IEEE, 6008–6013.

[9] Kuang-Chih Lee, Ali Jalali, and Ali Dasdan. 2013. Real time bid optimization with smooth budget delivery in online advertising. In *Proceedings of the seventh international workshop on data mining for online advertising*. 1–9.

[10] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* 54, 5 (2007), 22–es.

[11] Weinan Zhang, Yifei Rong, Jun Wang, Tianchi Zhu, and Xiaofan Wang. 2016. Feedback control of real-time display advertising. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. 407–416.

[12] Weinan Zhang, Shuai Yuan, and Jun Wang. 2014. Optimal real-time bidding for display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1077–1086.