

A Comparative Study of Static and Adaptive Traffic Allocation in A/B Testing Systems

Abhishek Kumar Shrivastava¹, Dr. Punit Kumar Johari²

¹Student, Madhav Institute of Technology & Science, Gwalior, M.P, India

²Associate Professor, Madhav Institute of Technology & Science, Gwalior, M.P, India

Abstract - A/B testing is widely used for comparing feature variants based on user interaction data such as clicks and conversions. In most conventional systems, traffic is distributed equally among variants, which is referred to as static traffic allocation. While this approach is simple and easy to analyze, it does not utilize intermediate results during the experiment. Because of this, it may miss opportunities to adapt earlier. In this work, we propose a methodology using a custom A/B Feature Testing application to simulate and analyze both static and adaptive traffic allocation behaviors. The simulated experiment evaluates 100,000 hypothetical users across three variants to observe system efficiency and bias. Interestingly, the static method identifies Variant 1 as the winner, whereas the adaptive method favors Variant 2. This difference suggests a possible trade-off. Adaptive allocation may improve efficiency by reducing regret, but it also appears to be sensitive to early statistical fluctuations. This paper explores that balance in more detail.

Key Words: A/B Testing, Static Allocation, Adaptive Allocation, Epsilon-Greedy, Traffic Allocation, Experimentation Systems

1. INTRODUCTION

A/B testing has become a standard approach in modern product development, allowing teams to evaluate feature variations based on real user behavior. A systematic literature review indicates that most implementations still rely on traditional single A/B test setups [1]. In these setups, traffic is distributed equally between variants, and results are evaluated only after sufficient data has been collected. While this method is straightforward and reliable, it is also rigid. Once the experiment begins, allocation does not change, even if early observations suggest that some variants are performing significantly worse than others. This can lead to inefficiencies, particularly in terms of wasted traffic and delayed decision-making. The concept of regret highlights this issue. Traffic assigned to underperforming variants represents a lost opportunity, and this cost accumulates overtime [2]. Additionally, real-world data often deviates from ideal assumptions. Non-Gaussian distributions and intra-user correlations can affect the reliability of results [3], [4], and static allocation does not account for these variations. To address these concerns, adaptive allocation strategies have been proposed. These methods adjust traffic dynamically based on observed performance. However, this raises another question: can early adaptation lead to premature conclusions? This work does not attempt to declare one method superior. Instead, it investigates how both approaches behave under the same conditions. A custom simulation environment is used to compare static allocation with an epsilon-greedy adaptive strategy. The goal is to understand their practical differences and identify potential trade-offs.

2. RELATED WORK

The challenges associated with A/B testing have been widely studied, particularly in the context of statistical reliability. Traditional methods often struggle when data deviates from normality, leading to unreliable estimates [3]. Similarly, ratio metrics introduce additional complexity and can behave unpredictably under certain conditions [4]. Another important issue is the assumption of independent users. In many real-world systems, users interact with each other, leading to violations of the Stable Unit Treatment Value Assumption (SUTVA) [5], [6]. This can introduce bias into experimental results. Techniques such as cluster-based randomization and link-aware testing have been developed to address these concerns. Specialized domains also require tailored approaches. For example, auction-based systems involve strategic interactions that complicate traditional A/B testing methods [7]. In parallel, automated experimentation frameworks like SEABYTE aim to integrate A/B testing into continuous deployment pipelines [8]. Recent work has also explored simulation-based evaluation using AI-driven agents [9]. Despite these developments, the problem of traffic allocation during an experiment remains an active area of interest. This work focuses specifically on that aspect.

3. CONCLUSIONS

The proposed approach is based on a simulation framework that models user traffic and allocation strategies. Incoming users are assigned to different variants through a routing mechanism, and two allocation strategies are evaluated simultaneously: static allocation and adaptive allocation. Static allocation distributes traffic uniformly across all variants, whereas adaptive allocation follows an epsilon greedy strategy, where a portion of traffic is used for exploration and the remainder is directed toward the currently best-performing variant. User behavior is modeled probabilistically, with clicks and conversions generated using Bernoulli distributions to capture variability in user interactions. The backend simulation logic integrates several design choices to reflect real-world behavior while maintaining computational efficiency. User events are generated probabilistically, specifically modeled as Click Bernoulli(p click) and Conversion Bernoulli(p conversion). To balance responsiveness with stability, updates are not performed after every user; instead, users are processed in batches of 100, which helps reduce noise and ensure more stable updates. At each batch interval, the system recalculates performance metrics and adjusts the allocation strategy accordingly, while continuously tracking key metrics such as impressions, conversions, and cumulative regret. The simulation processes a total of 100,000 users across three variants (V1, V2, V3), and the workflow includes initialization, user routing, event simulation, metric updates, and final evaluation. The results of adaptive allocation for 10,000 users are presented in Table 1, while the corresponding static allocation results are shown in Table 2. For the larger-scale experiment with 100,000 users, adaptive allocation results are provided in Table 3, and static allocation results are shown in Table 4. The overall comparative trend in regret across these experiments is illustrated in Figure 1.

4. MATHEMATICAL FORMULATIONS

$$\begin{aligned}
 CR &= \text{Conversions/Users} \\
 CTR &= \text{Clicks/Users} \\
 \text{Regret} &= \sum (\text{Users assigned to non-optimal variants}) \\
 P(\text{explore}) &= \epsilon, \\
 P(\text{exploit}) &= 1 - \epsilon
 \end{aligned}$$

5. EXPERIMENTAL RESULTS

A. Experiment 1 (10000 users)

1) Adaptive Allocation

Table 1

Variant	Users	Allocation	Clicks	Conversions	CTR	CR
A	6344	63.44%	1563	628	24.64%	9.90%
B	526	5.26%	133	45	25.29%	8.56%
C	3130	31.30%	813	302	25.97%	9.65%

2) Static Allocation

Table 2

Variant	Users	Allocation	Clicks	Conversions	CTR	CR
A	3334	33.34%	833	323	24.98%	9.69%
B	3333	33.33%	847	324	25.41%	9.72%
C	3333	33.33%	829	325	24.87%	9.75%

B. Experiment 2 (100000 users)

1) Adaptive Allocation

Table 3

Variant	Users	Allocation	Clicks	Conversions	CTR	CR
A	26342	26.34%	6684	2662	25.37%	10.11%
B	70189	70.19%	17578	6996	25.04%	9.97%
C	3469	3.47%	885	336	25.51%	9.69%

2) Static Allocation

Table 4

Variant	Users	Allocation	Clicks	Conversions	CTR	CR
A	33327	33.33%	8395	3368	25.19%	10.11%
B	33348	33.35%	8372	3235	25.10%	9.70%
C	33325	33.32%	8348	3324	25.05%	9.99%

6. GRAPHICAL ANALYSIS

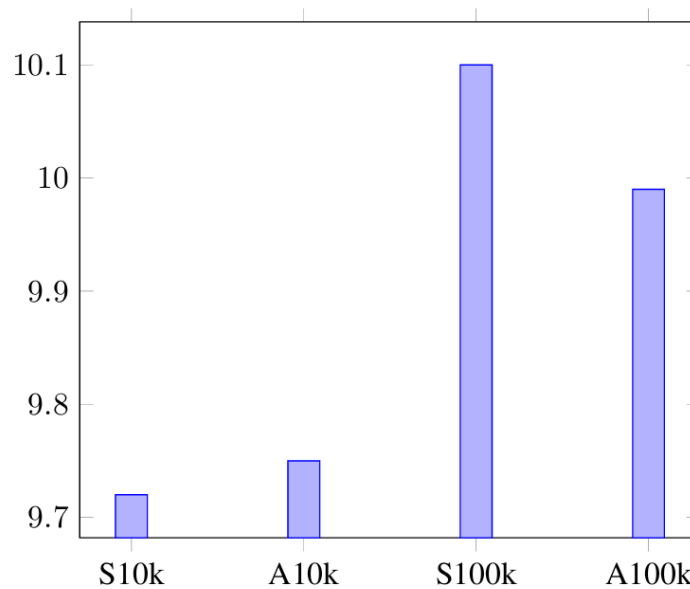


Figure 1

7. EXTENDED ANALYSIS

Adaptive allocation shows lower regret in smaller experiments but becomes unstable at scale. Static allocation, while inefficient early, provides more reliable outcomes over time.

8. DISCUSSION

The results show a noticeable difference between the two approaches. Under static allocation, traffic remains evenly distributed, and Variant 1 achieves the highest conversion rate at 10.11%. The other variants follow closely behind. In contrast, adaptive allocation gradually shifts traffic toward better-performing variants. Over time, Variant 2 emerges as the top performer with a slightly higher conversion rate than Variant 1. While adaptive allocation reduces regret by minimizing exposure to weaker variants, it also appears to be sensitive to early performance signals. If these signals are influenced by randomness, the system may converge to a suboptimal choice.

There are additional considerations. Adaptive methods assume a relatively stable environment, but real-world conditions can change over time. User behavior may shift due to external factors, which can affect the validity of results. Furthermore, adaptive allocation does not ensure equal exposure across variants, which may raise concerns in certain applications. Looking more closely at the experimental results, especially across the two different scales (10,000 vs 100,000 users), a more nuanced pattern begins to emerge. In the smaller experiment, adaptive allocation shows clear advantages. It reduces regret significantly and slightly improves conversion rate. This suggests that in low-traffic or short-duration experiments, reacting quickly to early signals may provide practical benefits.

However, the larger experiment tells a different story. With 100,000 users, static allocation performs better in terms of both conversion rate and regret. This is important. It indicates that adaptive systems may struggle to recover once they commit too early. The bias introduced in the initial phase continues to influence allocation decisions throughout the experiment. Another subtle observation is how traffic gets distributed in adaptive systems. In Experiment 2, a large portion of traffic is allocated to Variant B, even though Variant A eventually turns out to be the best performer. This suggests that the system may have overreacted to early performance differences. Once that happens, exploration becomes limited, and correcting the decision becomes harder. From a theoretical standpoint, this behavior aligns with the known limitations of epsilon-greedy strategies. While they maintain a balance between exploration and exploitation, the balance is not always sufficient when early observations are noisy.

The exploitation component tends to dominate, especially as more data is accumulated. On the other hand, static allocation appears inefficient in the short term. It continues sending users to weaker variants. But over time, this uniform sampling provides a more reliable estimate of true performance. In that sense, static allocation acts as a safeguard against premature convergence. This leads to a clearer interpretation of the trade-off: Adaptive allocation prioritizes efficiency, but risks bias. Static allocation prioritizes reliability but sacrifices short-term gains. Neither approach fully dominates the other.

The choice depends heavily on the context of the experiment — traffic volume, duration, and tolerance for risk.

A hybrid approach naturally follows from this observation. Starting with static allocation allows the system to gather stable and unbiased data.

Once sufficient confidence is achieved, switching to adaptive allocation can improve efficiency without being overly sensitive to early noise. This balance between stability and responsiveness appears to be the key challenge in designing practical A/B testing systems.

9. CONCLUSION

This paper presents a comparative analysis of static and adaptive traffic allocation in A/B testing systems. Through controlled simulation experiments at different scales, the study highlights that the performance of these methods is not absolute but context dependent. Static allocation demonstrates strong reliability. By distributing traffic uniformly, it ensures unbiased data collection and stable estimates of variant performance. However, this comes at the cost of efficiency, as a significant portion of traffic may be allocated to underperforming variants. Adaptive allocation, on the other hand, improves efficiency by dynamically shifting traffic toward better-performing variants.

This leads to reduced regret, particularly in smaller experiments. However, the results also show that adaptive methods are sensitive to early-stage noise. Once a bias is introduced, the system may continue reinforcing it, making recovery difficult. The experimental results reinforce this trade-off. In smaller-scale experiments, adaptive allocation performs better. In larger-scale experiments, static allocation provides more consistent and reliable outcomes. This suggests that adaptive methods may not be scaled as effectively when early uncertainty plays a significant role. A key insight from this work is that adaptive allocation is not universally superior. While it offers practical advantages, it also introduces risks that must be carefully managed.

Similarly, static allocation, although conservative, provides a level of robustness that remains valuable. Based on these observations, a hybrid strategy appears to be a promising direction. Combining the stability of static allocation with the efficiency of adaptive methods may offer a balanced solution. For example, an experiment could begin with uniform traffic distribution and transition to adaptive allocation once sufficient data has been collected. Overall, the findings suggest that the

design of A/B testing systems should not rely on a single allocation strategy. Instead, it should consider the trade-offs between efficiency, reliability, and adaptability, and choose an approach that aligns with the specific goals of the experiment.

10. FUTURE WORK

Future work could explore more advanced adaptive strategies such as Thompson Sampling or Upper Confidence Bound (UCB) methods [2]. These approaches explicitly account for uncertainty and may provide a better balance between exploration and exploitation. Another direction is to validate these findings using real-world data. While simulations provide valuable insights, practical deployment may reveal additional challenges. Hybrid approaches also present an interesting possibility. Starting with static allocation and gradually transitioning to adaptive strategies could help mitigate early-stage bias while still improving efficiency.

REFERENCES

- [1] Quin, Federico, Weyns, Danny, Galster, Matthias, Silva, Camila Costa, "A/B Testing: A Systematic Literature Review," 2023. <https://arxiv.org/pdf/2308.04929v1>
- [2] Kaufmann, Emilie, Cappé, Olivier, Garivier, Aurélien, "On the Complexity of A/B Testing," Conference on Learning Theory, Jun 2014, Barcelona, Spain. JMLR: Workshop and Conference Proceedings, 35, pp.461-481, 2014. <https://arxiv.org/pdf/1405.3224v2>
- [3] Gong, Junpeng, Wang, Chunkai, Li, Hao, Ma, Jinyong, Li, Haoxuan, He, Xu, "Beyond Normality: Reliable A/B Testing with Non Gaussian Data," 2025. <https://arxiv.org/pdf/2510.23666v1>
- [4] Nie, Keyu, Kong, Yinfei, Yuan, Ted Tao, Burke, Pauline Berry, "Dealing With Ratio Metrics in A/B Testing at the Presence of Intra-User Correlation and Segments," 2019. <https://arxiv.org/pdf/1911.03553v2>
- [5] Zhou, Yifan, Liu, Yang, Li, Ping, Hu, Feifang, "Cluster-Adaptive Network A/B Testing: From Randomization to Estimation," 2020. <https://arxiv.org/pdf/2008.08648v1>
- [6] Zhang, Qiong, "On the Asymptotics of Graph Cut Objectives for Experimental Designs of Network A/B Testing," 2023. <https://arxiv.org/pdf/2309.08797v1>
- [7] Chawla, Shuchi, Hartline, Jason D., Nekipelov, Denis, "A/B Testing of Auctions," 2016. <https://arxiv.org/pdf/1606.00908v1>
- [8] Quin, Federico, Weyns, Danny, "SEABYTE: A Self-adaptive Micro-service System Artifact for Automating A/B Testing," 2022. doi:10.1145/3524844.3528081
- [9] Zhang, Wenlin, Li, Xiangyang, Ge, Qiyuan, Dong, Kuicai, Jia, Pengyue, Li, Xiaopeng, Zhang, Zijian, Wang, Maolin, Wang, Yichao, Guo, Huifeng, Tang, Ruiming, Zhao, Xiangyu, "Exploring Recommender System Evaluation: A Multi-Modal User Agent Framework for A/B Testing," 2026. <https://arxiv.org/pdf/2601.04554v1>