TSMOO: Solving Multi-Objective Experimentation with Constrained Thompson Sampling

Krishna Chaitanya Kalagarla
Amazon
Seattle, WA, USA
krizchai@amazon.com

Yi Liu*
Amazon
Seattle, WA, USA
liuyi.feier@gmail.com

Lin Chai
Amazon
Seattle, WA, USA
linchai@amazon.com

Wenyang Liu
Amazon
Seattle, USA
lwenyang@amazon.com

Abstract

Traditional online A/B experimentation limits the number of treatments that can be evaluated concurrently. Bandit-based adaptive experimentation algorithms address this by dynamically reallocating traffic across an order of magnitude more treatments. Yet existing methods involve a fundamental trade-off: single-metric Thompson Sampling-based methods are robust to novelty effects but cannot accommodate multiple launch criteria, while elimination-based multi-objective methods support multiple constraints but risk prematurely removing promising treatments. We introduce TSMOO (Thompson Sampling with Multi-Objective Optimization), a method that bridges this gap by combining multi-metric optimization with continuous learning. Grounded in stochastically constrained best-arm identification, TSMOO extends single-metric Thompson Sampling to multi-objective batch traffic allocation by estimating multi-constraint feasibility at each allocation step, while preserving all treatments throughout exploration. It further incorporates uplift modeling that mitigates temporal effects shared across treatments and control and ensures that the Gaussian distribution assumption holds. In simulations replaying historical real-world experimentation patterns, TSMOO achieves 93-94% success rates in multi-winner settings and 63-66% under novelty effects, outperforming both single-metric and elimination-based baselines.

CCS Concepts

• **General and reference → Experimentation**; • **Theory of computation → Online learning theory**.

Keywords

Adaptive experimentation, multi-armed bandit, best-arm identification, Thompson Sampling

*Work done while at Amazon.

RecSys '26, Minneapolis, MN, USA

ACM ISBN 979-8-4007-2284-4/26/09
https://doi.org/10.1145/3773078.3831812

ACM Reference Format:
Krishna Chaitanya Kalagarla, Yi Liu, Lin Chai, and Wenyang Liu. 2026. TSMOO: Solving Multi-Objective Experimentation with Constrained Thompson Sampling. In *20th ACM Conference on Recommender Systems (RecSys '26), September 27–October 02, 2026, Minneapolis, MN, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3773078.3831812

1 Introduction

1.1 Adaptive Experimentation

Online controlled experiments have become a cornerstone of data-driven product development, enabling companies to evaluate new features, recommendation policies, interface designs and content strategies before broad deployment. Large-scale A/B testing platforms process thousands of such experiments annually, enabling data-driven product decisions [14, 20]. Traditional A/B testing limits experimenters to 4-6 treatments to maintain statistical power [20], requiring multiple sequential experiments to evaluate larger treatment spaces—a significant bottleneck for recommendation systems that must explore large candidate sets of ranking models or personalization strategies. This sequential approach slows down innovation cycles and increases opportunity costs.

To address this limitation, adaptive experimentation platforms have been developed, enabling concurrent testing of up to 40 treatments [2, 5]. Unlike standard A/B tests with fixed traffic allocation, these platforms dynamically reallocate traffic away from underperforming treatments toward promising candidates during an exploration phase (typically 2 weeks), then convert to a standard A/B test for validation of the top performers against control (typically 4 weeks) before deployment. This adaptive approach enables significantly faster innovation cycles than sequential testing. This two-phase structure has an important implication for algorithm design: the exploration algorithm should prioritize treatments most likely to pass the validation criteria, not merely identify the empirically best arm during exploration.

Further, during exploration, platforms typically use metrics attributable within a one-day window (e.g., revenue, signups, clicks), with derived metrics available as guardrails during the validation phase.

1.2 Multi-Metric Launch Criteria

Production experimentation requires treatments to satisfy multiple business criteria simultaneously. Metrics serve two distinct roles:

all metrics must tend toward positive impact for a treatment to be eligible for launch, but a designated *primary metric* (e.g., revenue) is additionally maximized, while *secondary metrics* (e.g., engagement, user satisfaction) need only satisfy minimum thresholds. This creates a constrained optimization problem.

We formalize these thresholds as posterior probability requirements (PPR): during the validation phase, each metric requires that the posterior probability of a positive treatment effect exceeds a specified threshold (e.g., at least 66%). A treatment is eligible for launch only when it meets the PPR threshold for *every* metric. Bayesian A/B testing has seen wide adoption in industrial experimentation [10, 14], and the posterior probability threshold provides a directly interpretable decision criterion for practitioners: "the probability that this treatment is better than control" is a natural basis for a launch decision.

Beyond multi-metric requirements, adaptive experimentation faces additional practical challenges: novelty effects, where early user behavior driven by curiosity differs from steady-state preference, and non-Gaussian reward distributions that violate standard bandit assumptions. These challenges are discussed in detail in Section 2.

1.3 Current Algorithm Landscape

Existing approaches to adaptive experimentation fall into two categories, each with a distinct limitation:

Single-metric Thompson Sampling methods [3] optimize a single primary metric using Thompson Sampling-based traffic allocation. By maintaining all treatments throughout exploration and continuously updating posterior distributions, these methods provide robustness to novelty effects. However, they cannot simultaneously optimize for multiple PPR constraints, limiting applicability to single-metric scenarios.

Sequential Elimination methods (e.g., SE-MOO—Sequential Elimination Multi-Objective Optimization, as introduced by Zhang et al. [22]) evaluate treatments against multiple PPR thresholds, systematically eliminating underperforming treatments in successive rounds. While effective for multi-objective optimization, the elimination approach creates vulnerability to novelty effects, where treatments showing poor early performance may be prematurely removed before reaching steady-state behavior.

Both approaches also share a common limitation: they operate on absolute reward signals, making them susceptible to temporal confounding (e.g., day-of-week effects, seasonality) and non-Gaussian reward distributions common in e-commerce settings. This creates a genuine algorithmic gap: no existing method simultaneously provides novelty robustness, multi-metric constraint handling, and robust reward estimation. TSMOO bridges this gap.

1.4 TSMOO: Bridging the Gap

We present TSMOO (Thompson Sampling with Multi-Objective Optimization), an algorithm that bridges this gap. TSMOO makes the following contributions:

- **Algorithmic gap diagnosis**: We identify that existing methods must sacrifice either novelty robustness or multi-metric constraint handling, a structural incompatibility that no prior method resolves simultaneously.
- **Multi-objective continuous exploration**: Building on the BFAI-TS framework [21], we extend Thompson Sampling to batch daily traffic allocation that simultaneously enforces multiple launch criteria without treatment elimination, providing principled theoretical foundations rather than heuristic construction.
- **Validation-aligned uplift estimation**: We reformulate the exploration objective around treatment-vs-control uplift, simultaneously canceling temporal confounding, satisfying Gaussian distributional assumptions, and aligning exploration decisions with validation success criteria.
- **Production-grounded evaluation**: We validate performance through simulations replaying historical real-world experimentation patterns, providing more realistic estimates than fully synthetic Gaussian simulations.

Comprehensive simulations replaying historical real-world experimentation patterns demonstrate TSMOO's effectiveness: 93-94% success rates in multi-winner scenarios and sustained 63-66% performance under novelty effects, compared to 85-90% and 53-58% for single-metric methods, and 89-94% and 49-62% for elimination-based methods respectively.

2 Challenges in Production Adaptive Experimentation

The dynamic nature of production experimentation environments presents four key challenges for reward modeling in online adaptive experimentation. **Challenge 1: Temporal Variation.** User engagement and spending behavior exhibit substantial temporal variation. Treatment reward performance can fluctuate based on the hour of the day, the day within the experiment, weekends, or special events such as sales, introducing significant noise that may obscure the true treatment effect. In our experiments, we aim to identify one or more winning treatments for follow-up validation, ultimately selecting a single treatment for deployment. For example, when testing different webpage designs, showing one design on weekdays and another on weekends could create a confusing and inconsistent user experience. Even if different designs perform best on different days, we still seek a single design that outperforms the control on average. We therefore cannot include time-based covariates in our model to explain temporal variation, since doing so would undermine the goal of selecting one consistent deployable treatment. The temporal effects typically affect both treatment and control groups similarly. Therefore, it is conceptually more efficient to focus on the difference between treatment and control, i.e., the uplift, rather than attempting to explain all variance in treatment performance.

Challenge 2: Non-Gaussian Reward Distributions. The reward distribution can deviate significantly from the Gaussian assumption commonly made by many multi-armed bandit algorithms. For instance, in practice, users do not always make a purchase or click an advertisement, resulting in zero-inflated reward observations. Another problem is that reward can be highly skewed and long-tailed. The bucket-level uplift computation described in Section 3.2 addresses this: by the Central Limit Theorem, bucket-level averages approximate Gaussian distributions regardless of the

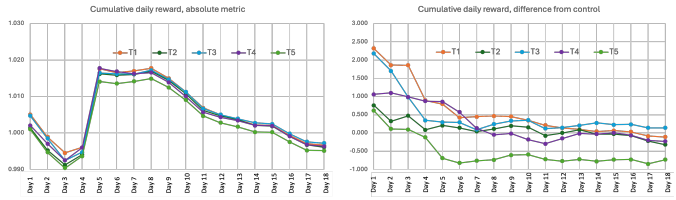

Figure 1: Treatment performance (normalized): absolute reward vs. deviation from control

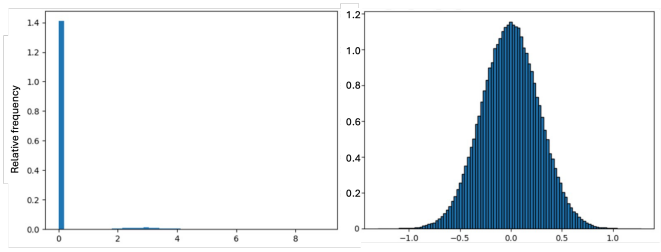

Figure 2: Normality check of treatment's reward (similar for all treatments). Left: metric at individual observation level; Right: metric uplift at bucket level.

underlying reward distribution, enabling valid Bayesian posterior updates.

Challenge 3: Novelty Effects. Novelty effects are almost always present in user-facing content experiments. Treatments typically involve new recommendation strategies or content versions that users have not seen before. Early engagement may be driven by curiosity rather than genuine preference, resulting in a novelty effect at the beginning of the experiment. The extent and duration of novelty effects vary across treatments and cannot be known in advance. For large-scale platforms that receive tens of millions of user visits daily, bandit algorithms may prematurely converge to suboptimal treatments, locking in choices before sufficient exploration has taken place to account for transient novelty-driven engagement. For instance, a treatment showing poor performance in the first week may recover to steady-state behavior by week two; an elimination-based algorithm would remove it early, never observing its true long-term performance.

Challenge 4: Multi-Metric Launch Criteria. Production launches require treatments to demonstrate positive impact across multiple business metrics simultaneously. Deploying a treatment that degrades any metric, even one that improves the primary objective, can have serious consequences for user experience and revenue. This creates a constrained optimization problem where the exploration algorithm must account for all launch criteria, not just the primary metric.

3 Problem Formulation

3.1 Best Feasible Arm Identification Framework

We formulate multi-objective experimentation as a stochastically constrained multi-armed bandit problem [16] within the Best Feasible Arm Identification (BFAI) framework [21], seeking to identify the arm with highest expected reward on a primary metric while ensuring all metrics meet specified thresholds.

Let $\mathcal{A} = \{1, 2, \ldots, k\}$ represent available arms (treatments). When arm i is selected at time t, we observe multidimensional reward $X_{t,i} = [X_{t,i,1}, \ldots, X_{t,i,m}] \in \mathbb{R}^m$ where $X_{t,i,1}$ represents the primary metric reward. We assume reward samples follow normal distributions $N(\mu_{i,j}, \sigma^2_{i,j})$ with independence across time, arms, and metrics. The BFAI objective seeks:

$$i^* = \arg\max_{i \in \mathcal{A}} \ \mu_{i,1} \tag{1}$$

$$\text{s.t.} \quad \mu_{i,j} \geq \gamma_j, \quad \forall j = 1, 2, \ldots, m \tag{2}$$

where γ_j are constraint thresholds derived from PPR requirements: each metric j requires a minimum PPR of $1 - q_j$, i.e., the posterior probability that the treatment effect is positive must exceed $1 - q_j$. In our context, rewards $X_{t,i}$ are scaled uplift values for each metric.

3.2 Uplift Transformation

TSMOO focuses on treatment effects rather than absolute performance metrics. The reward samples $X_{t,i}$ represent uplift values computed as the difference between treatment and control observations matched by temporal periods, which are then scaled for algorithmic use. This uplift transformation addresses the temporal variation and reward distribution challenges described in Section 2.

To implement this uplift calculation reliably, observations are first aggregated into equal-count buckets for both control and treatment groups separately. Bucket-level aggregation is necessary because adaptive traffic allocation causes treatments to receive different traffic volumes, making computing uplift at the individual observation level would be unreliable under such imbalances. The number of buckets B is determined by the control traffic volume divided by a default bucket size; all treatments use the same number of buckets B, so each treatment contributes the same number of uplift samples to the Bayesian posterior update regardless of its traffic allocation. Uplift values are then computed as the element-wise difference between corresponding treatment and control bucket means.

3.3 Bayesian Inference Framework

TSMOO employs Bayesian inference with normal priors $N(\mu_{1,i,j}, \sigma^2_{1,i,j})$ for each arm-metric combination. Posterior distributions follow $N(\mu_{t,i,j}, \sigma^2_{t,i,j})$ with conjugate prior updates for batch observations. We denote the posterior distribution over the vector of means of the m metric measures by:

$$\Pi_t = N(\mu_{t,1,1}, \sigma^2_{t,1,1}) \otimes \ldots \otimes N(\mu_{t,1,m}, \sigma^2_{t,1,m}) \otimes \ldots$$
$$\otimes N(\mu_{t,k,1}, \sigma^2_{t,k,1}) \otimes \ldots \otimes N(\mu_{t,k,m}, \sigma^2_{t,k,m})$$

When arm i is selected $N_{t,i}$ times at time t, yielding observations $\{X^{(1)}_{t,i,j}, X^{(2)}_{t,i,j}, \ldots, X^{(N_{t,i})}_{t,i,j}\}$ with sample mean $\bar{X}_{t,i,j} = \frac{1}{N_{t,i}} \sum_{n=1}^{N_{t,i}} X^{(n)}_{t,i,j}$, the posterior updates become:

$$\mu_{t+1,i,j} = \begin{cases} \frac{\sigma^{-2}_{t,i,j}\mu_{t,i,j} + N_{t,i}\sigma^{-2}_{i,j}\bar{X}_{t,i,j}}{\sigma^{-2}_{t,i,j} + N_{t,i}\sigma^{-2}_{i,j}} & \text{if arm } i \text{ selected at time } t \\ \mu_{t,i,j} & \text{otherwise} \end{cases} \tag{3}$$

$$\sigma^2_{t+1,i,j} = \begin{cases} \frac{1}{\sigma^{-2}_{t,i,j} + N_{t,i}\sigma^{-2}_{i,j}} & \text{if arm } i \text{ selected at time } t \\ \sigma^2_{t,i,j} & \text{otherwise} \end{cases} \tag{4}$$

These updates process batch observations, handling multiple uplift samples (from bucketed data) simultaneously on each exploration day, with $N_{t,i}$ representing the number of bucket-level observations for treatment i on day t.

4 TSMOO Algorithm

TSMOO adapts the BFAI framework for practical multi-objective experimentation: arms correspond to treatment alternatives competing for traffic allocation, rewards are scaled uplift values relative to control, and constraints translate PPR requirements to thresholds. Control maintains fixed traffic allocation (e.g., 10%) to ensure a stable baseline for uplift estimation throughout exploration, while treatments compete for the remaining traffic based on their uplift performance and constraint satisfaction.

4.1 Original BFAI-TS Sequential Algorithm

The theoretical foundation for TSMOO comes from the BFAI-TS algorithm [21], which operates sequentially by selecting one arm at each time step and observing a single sample from that arm, making selection decisions based on feasibility:

Algorithm 1 BFAI-TS Algorithm

Require: Number of arms $k \geq 2$, Bernoulli Parameter $\beta \in (0,1)$, n

while $t \leq n$ **do**
  Sample $\theta \sim \Pi_t$
  Get the feasible set $F \triangleq \{i : \theta_{i,j} \geq \gamma_j \text{ for } j = 1, 2, \ldots, m\}$
  if $F \neq \emptyset$ **then**
    Set $I_t^{(1)} \leftarrow \arg\max \theta_{i,1}$ for $i \in F$
  else
    Choose $I_t^{(1)}$ uniformly from $\{1, 2, \ldots, k\}$
  end if
  Sample $B \sim$ Bernoulli(β)
  if $B = 1$ **then**
    Play $I_t^{(1)}$
  else
    repeat
      Sample $\theta \sim \Pi_t$
      Get the feasible set $F \triangleq \{i : \theta_{i,j} \geq \gamma_j \text{ for } j = 1, 2, \ldots, m\}$
      if $F \neq \emptyset$ **then**
        Set $I_t^{(2)} \leftarrow \arg\max \theta_{i,1}$ for $i \in F$
      else
        Choose $I_t^{(2)}$ uniformly from $\{1, 2, \ldots, k\}$
      end if
    until $I_t^{(2)} \neq I_t^{(1)}$
    Play $I_t^{(2)}$
  end if
  Update posterior Π_{t+1}
end while
Output: Recommended arm I^*

This sequential algorithm forms the basis for TSMOO's batch adaptation. Specifically, TSMOO retains the feasibility checking logic ($F \triangleq \{i : \theta_{i,j} \geq \gamma_j\}$) and Top-Two Thompson sampling structure, while contributing batch traffic allocation (Section 4.3), uplift-based reward modeling with bucket-level aggregation (Section 3.2), and validation-aligned constraint thresholds (Section 4.2).

4.2 Translating Validation Requirements to Constraint Thresholds

A key challenge is translating PPR validation requirements into mathematical constraint thresholds for the BFAI framework. The validation phase compares the recommended treatment I to control using T_v independent uplift samples, testing for significant improvement across all metrics where each metric j requires a minimum PPR of $1 - q_j$.

For each metric j, we need to ensure that treatments selected by TSMOO will pass the PPR validation test. This requires converting the Bayesian posterior probability requirement (PPR) into a threshold in the BFAI framework, requiring appropriate uplift value scaling.

To model the validation process, we assume that for each metric j, the treatment effect has a normal prior distribution $\mu^{raw}_{I,j} \sim N(0, \chi^2_j)$ with known variance $\sigma^{2,v}_{I,j}$. After observing treatment effect validation samples $\{X_{1,I,j}, \ldots, X_{T_v,I,j}\}$, the validation posterior distribution becomes:

$$\bar{P}_{I,j} \sim N\left(\frac{T_v \bar{\sigma}^2_{I,j}}{\sigma^{2,v}_{I,j}}(\bar{\mu}^{raw}_{I,j}),\ \bar{\sigma}^2_{I,j}\right) \tag{5}$$

where $\bar{\mu}^{raw}_{I,j}$ is the empirical treatment effect mean and $\bar{\sigma}^2_{I,j} = \left(\frac{T_v}{\sigma^{2,v}_{I,j}} + \frac{1}{\chi^2_j}\right)^{-1}$.

Treatment I passes validation for metric j when $P(\mu^{raw}_{I,j} > 0) \geq 1 - q_j$, which implies:

$$\bar{\mu}^{raw}_{I,j} \geq -\frac{\sigma^v_{I,j}\Phi^{-1}(q_j)}{\sqrt{T_v}}\sqrt{1 + \frac{\sigma^{2,v}_{I,j}}{T_v\chi^2_j}} \tag{6}$$

Since $\bar{\mu}^{raw}_{I,j} \sim N\left(\mu^{raw}_{I,j}, \frac{\sigma^{2,v}_{I,j}}{T_v}\right)$, satisfying the above with confidence $1 - \delta$ requires:

$$\mu^{raw}_{I,j} \geq -\frac{\sigma^v_{I,j}\Phi^{-1}(q_j)}{\sqrt{T_v}}\sqrt{1 + \frac{\sigma^{2,v}_{I,j}}{T_v\chi^2_j}} - \frac{\sigma^v_{I,j}\Phi^{-1}(\delta)}{\sqrt{T_v}} \tag{7}$$

In practice, for large T_v, we simplify the implementation by treating $\bar{\mu}^{raw}_{I,j}$ as the true mean, avoiding the conservative adjustment term with δ. Defining the scaled uplift mean as:

$$\mu_{I,j} = \frac{\mu^{raw}_{I,j}}{\frac{\sigma^v_{I,j}}{\sqrt{T_v}}\sqrt{1 + \frac{\sigma^{2,v}_{I,j}}{T_v\chi^2_j}}} \tag{8}$$

we obtain the constraint threshold:

$$\mu_{I,j} \geq \gamma_j = -\Phi^{-1}(q_j) \tag{9}$$

This derivation establishes both the constraint thresholds required by the BFAI framework and the scaling transformation that ensures these thresholds align with actual PPR validation requirements.

4.3 Batch Allocation via Top-Two Thompson Sampling

Production systems require daily traffic allocation rather than per-request arm selection. TSMOO adapts the sequential BFAI-TS algorithm [21] to batch settings by translating arm selection probabilities into traffic allocation proportions. We take $\beta = 0.5$ as the default choice, following [21].

The Top-Two Thompson Sampling approach operates by decomposing arm selection into two distinct scenarios. In each round, we first sample from posterior distributions to identify a primary candidate arm. Then, with probability β, we select this first choice directly. With probability $1 - \beta$, we perform a second sampling to identify an alternative arm, ensuring we don't select the same arm twice.

The allocation probability for arm i at time t is:

$$\phi_{t,i} = \beta A_i + (1 - \beta) \sum_{j \neq i} A_j \frac{A_i}{1 - A_j} \tag{10}$$

where $A_i = \frac{c_t}{k} + P_{t,i}$ represents the total probability that arm i would be selected in a single Thompson sample, where $P_{t,i}$ is the posterior probability that arm i is both feasible and optimal, c_t is the probability that no arm is feasible, and k is the number of arms. The first term βA_i represents arm i being selected as first choice, while the second term captures selection as second choice when another arm j is selected first.

Algorithm 2 EstimateSelectionProbs

Input: $P_t \in \mathbb{R}^k$: Prob. of each arm being feasible and optimal
Input: $P_t^{\text{inf}} \in \mathbb{R}^k$: Prob. of each arm being infeasible and optimal with respect to threshold gap
Input: $\beta \in (0, 1)$: Exploration parameter
Input: ϵ: Tolerance for numerical stability
$A_i \leftarrow P_t^{\text{inf}} + P_{t,i}$ for $i \in [k]$
if $\exists i : A_i > 1 - \epsilon$ **then**
  $i^* \leftarrow \arg\max_i A_i$
  $\phi_{t,i^*} \leftarrow \beta$
  if $A_{i^*} = 1$ **then**
    $\phi_{t,i} \leftarrow (1 - \beta)/(k - 1)$ for $i \neq i^*$
  else
    $\phi_{t,i} \leftarrow (1 - \beta)A_i / \sum_{j \neq i^*} A_j$ for $i \neq i^*$
  end if
else
  for $i = 1$ **to** k **do**
    $\phi_{t,i} \leftarrow \beta A_i + (1 - \beta) \sum_{j \neq i} A_j \frac{A_i}{1 - A_j}$
  end for
end if
Normalize ϕ_t such that $\sum_i \phi_{t,i} = 1$
Output: ϕ_t: Estimated probability of selecting each arm

The special case (when $A_i > 1 - \epsilon$ for some i) handles the situation where one arm is highly likely to be optimal, requiring a modified distribution to avoid numerical instability.

4.4 Feasibility Probability Estimation

To implement the BFAI decision logic in batch form, TSMOO estimates the probability of each arm selection scenario from the original sequential algorithm. Unlike the theoretical BFAI-TS algorithm, which assumes at least one feasible arm always exists, practical experimentation may encounter situations where no treatment satisfies all constraint metrics. TSMOO addresses this by selecting the "least infeasible" arm when no treatments meet all constraints.

The decision-making process estimates probabilities using $N = 10^6$ Monte Carlo samples drawn from posterior distributions $\theta_{i,j} \sim N(\mu_{t,i,j}, \sigma^2_{t,i,j})$. The algorithm estimates three key probabilities: c_t (probability that no treatment is feasible), $P_{t,i}$ (probability that treatment i is both feasible and optimal), and $P_{inf,t,i}$ (probability that treatment i is the "least infeasible" when no treatments satisfy all constraints).

For each sample $\theta^{(n)}$, the feasible set is $F^{(n)} = \{i : \theta_{i,j}^{(n)} \geq \gamma_j \text{ for all } j\}$. The enhanced infeasible selection strategy works as follows: when all treatments violate at least one constraint, rather than selecting uniformly, TSMOO identifies the treatment that comes closest to satisfying all constraints by calculating constraint gaps $(\theta_{i,j} - \gamma_j)$ for each treatment-metric combination, taking the minimum gap across metrics for each treatment, and selecting the treatment with the largest minimum gap.

This "least infeasible" approach provides more intelligent exploration when no treatments currently meet all launch criteria.

Algorithm 3 EstimateFeasibilityProbs

Input: Posterior distribution Π_t, constraint thresholds γ, sample size N
Output: c_t, P_t, $P_{inf,t}$
Initialize count_infeasible $\leftarrow 0$
Initialize optimal_counts $\leftarrow$ zero array of length k
Initialize optimal_infeasible_counts $\leftarrow$ zero array of length k
for $n = 1$ **to** N **do**
  $\theta \sim \Pi_t$ {Sample from posterior distributions}
  $F \leftarrow \{i : \theta_{i,j} \geq \gamma_j \text{ for all } j\}$ {Identify feasible arms}
  if F is empty **then**
    count_infeasible $\leftarrow$ count_infeasible +1
    Calculate constraint gaps: gaps = $\theta - \gamma$
    min_gaps = min(gaps, axis = 1)
    $i^* \leftarrow \arg\max$(min_gaps) {Best infeasible arm}
    optimal_infeasible_counts[i^*] $\leftarrow$ optimal_infeasible_counts[i^*] +1
  else
    $i^* \leftarrow \arg\max_{i \in F} \theta_{i,1}$ {Best feasible arm}
    optimal_counts[i^*] $\leftarrow$ optimal_counts[i^*] +1
  end if
end for
$c_t \leftarrow$ count_infeasible$/N$
$P_t \leftarrow$ optimal_counts$/N$
$P_{inf,t} \leftarrow$ optimal_infeasible_counts$/N$

4.5 Complete Algorithm

Algorithm 4 presents the complete TSMOO procedure for daily operation.

Algorithm 4 TSMOO Algorithm

Input:
k: Number of treatments, n: Exploration duration in days
M: Daily traffic number, c: Control traffic fraction
β: Bernoulli exploration parameter, γ: Constraint thresholds
Initialize Gaussian posteriors Π_i for each treatment $i \in [k]$
Set uniform allocation: $\phi_{1,i} = 1/k$ for each treatment i {Initial allocation}
for $t = 1$ **to** n **do**
  // Allocate daily traffic according to latest ϕ_t
  Allocate fixed traffic to control: $\phi_{\text{control}} = c$
  $n_i \leftarrow M \times (1 - c) \times \phi_{t,i}$ for each treatment i {Treatment samples}
  // Collect observations and compute uplifts
  Collect $n_c = M \times \phi_{\text{control}}$ control samples
  for each treatment i **do**
    Collect n_i treatment samples
    Compute m_i uplift samples (after bucketing) from treatment and control observations
    Update posterior: $\Pi_i \leftarrow \text{UpdatePosterior}(\Pi_i, \{X_{i,1}, \ldots, X_{i,m_i}\})$
  end for
  // Calculate allocation for next day
  if $t < n$ **then**
    $(c_t, P_t, P_{inf,t}) \leftarrow \text{EstimateFeasibilityProbs}(\Pi, \gamma, N)$
    $\phi_{t+1} \leftarrow \text{EstimateSelectionProbs}(P_t, P_{inf,t}, c_t, \beta, \epsilon)$
  end if
end for
Return: Final posteriors Π, recommended treatments (top-3 by P_n)

5 Empirical Validation

5.1 Overview

We validated TSMOO through realistic simulations replaying historical real-world experimentation patterns, with 12 treatments and 2 metrics (a primary revenue metric and a secondary revenue metric). The scaling methodology described below enables precise control over treatment effect magnitudes to achieve target PPR values.

We extracted data from the first three days of uniform allocation in a large-scale online experiment to create three distinct baseline distributions (Versions 1-3), each capturing different natural variations in treatment performance. This scenario diversity reflects real experimental conditions. We resample these distributions with replacement to simulate treatment behavior while preserving natural reward distributions. Our evaluation spans 8 scenarios with 300 simulation runs each for stationary settings and 200 runs for non-stationary settings. Each simulation runs for 14 days, matching typical exploration duration.

5.2 Simulation Methodology

Scaling Factor Determination. For controlled scenario creation, we apply multiplicative scaling to achieve target PPR values: $Y_{t,i,\text{new}} = Y_{t,i} \cdot \alpha_i$. The scaling factor α is determined by solving:

$$\text{PPR}_{\text{target}} = \Phi\left(\frac{\alpha \cdot \mu_t - \mu_c}{\sqrt{\frac{\sigma_c^2 + \alpha^2 \cdot \sigma_t^2}{n}}}\right) \tag{11}$$

This yields a quadratic equation in α:

$$(n\mu_t^2 - z_{\text{target}}^2\sigma_t^2)\alpha^2 - (2n\mu_t\mu_c)\alpha + (n\mu_c^2 - z_{\text{target}}^2\sigma_c^2) = 0 \tag{12}$$

Solving this equation provides precise control over treatment effect magnitudes while maintaining realistic variance structures. The multiplicative approach allows independent control of effect sizes for each metric.

Controlled Scenario Creation. For each simulation run:

(1) Resample from actual treatment distributions (preserving real variance structures)
(2) Apply multiplicative scaling to achieve target PPR values: $Y_{t,i,\text{new}} = Y_{t,i} \cdot \alpha_i$
(3) The scaling factor α_i is determined by solving for the desired PPR level (described above)

Non-Stationary Modeling. Non-stationary settings implement time-varying scaling factors to simulate novelty effects:

$$\alpha_{i,t} = \begin{cases} 1 - 2\varepsilon_i & \text{if treatment } i \text{ is winning and } t \in \{1,2,3\} \\ 1 + 2\varepsilon_i & \text{if treatment } i \text{ is non-winning and } t \in \{1,2,3\} \\ 1 + \varepsilon_i & \text{if } t \geq 5 \\ \frac{1}{2}(\alpha_{i,3} + \alpha_{i,5}) & \text{if } t = 4 \end{cases} \tag{13}$$

This creates realistic temporal dynamics where initial user reactions differ from long-term behavior. The symmetric perturbation $\pm 2\varepsilon_i$ provides controlled evaluation of algorithmic robustness to temporal performance shifts.

5.3 Validation Duration Calibration

The constraint thresholds γ_j depend on T_v, the effective validation duration, which must be calibrated to align the statistical power of daily exploration updates with the longer validation window used in production (e.g., 28 days). We calibrate T_v empirically by comparing the statistical strength of daily versus 28-day treatment assessments across historical experiments.

Since the constraint thresholds $\gamma_j = -\Phi^{-1}(q_j)$ are expressed in z-score space, we convert treatment performance to z-scores for comparison. For each historical experiment h and duration d, we calculate:

$$z_{h,d} = \frac{\hat{\mu}_{h,d}}{\sqrt{\hat{\sigma}_{h,d}^2 / n_{h,d}}} \tag{14}$$

where $\hat{\mu}_{h,d}$ is the treatment mean over d days, $\hat{\sigma}_{h,d}^2$ is the treatment effect variance, and $n_{h,d}$ is the sample size.

The scaling ratio between 1-day and 28-day z-scores is then calculated as:

$$\text{scaling\_ratio} = \frac{z_{h,28}}{z_{h,1}} \tag{15}$$

This ratio represents the square root of the "effective validation duration", the number of statistical days equivalent to one actual day for constraint evaluation purposes. Table 1 shows empirical scaling ratios from historical experiments using a revenue metric.

Table 1: Validation Duration Calibration from Historical Experiments

Experiment	Scaling Ratio	Effective Duration
Experiment A	3.37	11.34
Experiment B	3.40	11.58
Experiment C	3.79	14.42
Experiment D	3.28	10.78
Experiment E	3.22	10.36

Derivation of T_v. Based on the empirical scaling ratios in the table above, we determined an effective validation duration of 12 days for constraint calibration:

$$T_v = \frac{\text{effective\_validation\_duration} \times \text{daily\_traffic\_estimate}}{\text{number of treatments in validation}} \quad (16)$$

5.4 Experimental Scenarios

We designed eight scenarios to systematically test algorithm performance across the spectrum of possible experimental outcomes. Each scenario represents a different configuration of "winners" (treatments satisfying 66% PPR for both metrics):

- **Single Winner (Primary Leader)**: One treatment satisfies PPR and achieves highest primary metric performance
- **Single Winner (Non-Primary Leader)**: One treatment satisfies PPR but does not lead on primary metric
- **Multiple Winners (Type 1, 2)**: Three treatments satisfy PPR with varying performance patterns
- **No Winner (Moderate)**: No treatments satisfy PPR; primary metric leader has moderate secondary performance
- **No Winner (Severe)**: No treatments satisfy PPR; primary metric leader has very poor secondary performance
- **No Winner (Strong 2nd)**: No treatments satisfy PPR; primary metric leader has strongest secondary performance
- **No Clear Winner**: No treatments achieve strong performance on either metric

Most scenarios have non-stationary variants simulating novelty effects, where winning treatments initially underperform (days 1-3) while non-winning treatments appear inflated, with gradual convergence to steady-state by day 5.

5.5 Evaluation Metrics

We compare TSMOO against two baselines representing the existing algorithm landscape described in Section 2: **Uplift-TS** (uplift-based Thompson Sampling adapted to batch daily allocation for single-metric optimization, providing novelty robustness but without multi-metric constraint handling) and **SE-MOO** [22] (sequential elimination with multi-metric PPR thresholds, supporting multiple constraints but vulnerable to novelty effects). These two baselines represent the two ends of the existing trade-off, making them the natural comparison points for TSMOO.

Table 2: Stationary Environment Success Rate (%)

(a) Version 1

Scenario	TSMOO	Uplift-TS	SE-MOO
1 Winner (Primary)	62.2	40.3	61.1
1 Winner (Non-Prim)	46.9	28.8	48.3
3 Winners (Type 1)	93.4	84.0	89.2
3 Winners (Type 2)	94.1	89.6	92.0
0 Winner (Moderate)	48.3	46.9	42.0
0 Winner (Severe)	26.0	47.6	11.8
0 Winner (Strong 2nd)	50.3	39.9	45.1
No Clear Winner	28.8	33.3	27.1

(b) Version 2

Scenario	TSMOO	Uplift-TS	SE-MOO
1 Winner (Primary)	62.2	43.1	64.1
1 Winner (Non-Prim)	48.3	36.8	56.3
3 Winners (Type 1)	93.8	85.1	92.0
3 Winners (Type 2)	93.4	87.9	93.4
0 Winner (Moderate)	48.6	52.1	43.8
0 Winner (Severe)	27.1	53.5	13.5
0 Winner (Strong 2nd)	45.1	41.3	46.5
No Clear Winner	43.8	42.0	40.3

(c) Version 3

Scenario	TSMOO	Uplift-TS	SE-MOO
1 Winner (Primary)	57.6	47.6	61.8
1 Winner (Non-Prim)	49.7	35.1	52.1
3 Winners (Type 1)	92.4	87.9	94.1
3 Winners (Type 2)	93.1	89.6	89.2
0 Winner (Moderate)	46.2	48.3	44.1
0 Winner (Severe)	26.0	49.3	13.9
0 Winner (Strong 2nd)	49.0	42.0	46.5
No Clear Winner	36.8	32.6	44.4

Algorithm Outputs: Each algorithm recommends its top-3 treatments after 14 days: TSMOO uses maximum posterior probability of winning via Monte Carlo sampling, Uplift-TS uses posterior Inverse Coefficient of Variation i.e., ICV (μ/σ) scores on primary metric, and SE-MOO uses treatments surviving sequential elimination rounds.

Ground Truth: When treatments satisfy the PPR threshold ($\geq$66% for both metrics), all satisfying treatments constitute the winner set. When no treatments satisfy the PPR threshold, the primary metric leader is the winner.

Success Criterion: A run succeeds when the algorithm's top-3 recommendations intersect with the ground truth winner set.

5.6 Results

Tables 2 and 3 present complete results across all scenarios and data versions.

Table 3: Non-Stationary Environment Success Rate (%)

(a) Version 1

Scenario	TSMOO	Uplift-TS	SE-MOO
1 Winner (Primary)	32.3	23.4	24.5
3 Winners (Type 1)	66.2	58.3	62.0
0 Winner (Strong 2nd)	34.4	29.2	28.7
No Clear Winner	31.8	24.5	27.1

(b) Version 2

Scenario	TSMOO	Uplift-TS	SE-MOO
1 Winner (Primary)	27.1	22.9	17.2
3 Winners (Type 1)	63.5	53.1	49.0
0 Winner (Strong 2nd)	32.8	32.3	29.7
No Clear Winner	31.8	26.6	26.0

5.7 Key Findings

Novelty Robustness: TSMOO consistently outperforms both baselines in non-stationary scenarios, maintaining 27-66% success rates while Uplift-TS degrades to 22-58% and SE-MOO to 17-62%. This robustness advantage is most pronounced in multi-winner scenarios (63-66% vs 49-58%). Continuous posterior updates without treatment elimination enable adaptation to changing treatment effects.

Multi-Winner Stationary Performance: TSMOO achieves 93-94% success rates in multi-winner scenarios, compared to Uplift-TS (84-90%) and SE-MOO (89-94%).

Balanced Performance: TSMOO provides consistently comparable or better performance across diverse scenarios, making it suitable for general deployment where experimental characteristics vary unpredictably.

Algorithmic Trade-offs in Edge Cases: In non-primary winner scenarios, SE-MOO slightly outperforms TSMOO (48-56% vs 46-50%) since TSMOO prioritizes primary metric optimization, a design choice aligned with typical launch decisions. In severe constraint violations, Uplift-TS outperforms both multi-objective algorithms (47-53% vs 26-28%) due to exclusive primary metric focus; TSMOO's constraint awareness appropriately avoids treatments with poor secondary performance.

6 Related Work

Online experimentation at scale has been extensively studied in industry [14, 20], and adaptive experimentation platforms have been deployed in production [2, 5], motivating the development of algorithms that address the practical challenges described in Section 2.

The Top-Two framework provides a principled approach for best arm identification by maintaining competition between the empirically best arm and promising alternatives [19]. Top-Two Thompson Sampling extends the efficient TS heuristic, originally designed for regret minimization, to pure exploration settings. Unlike elimination-based methods such as Sequential Halving [11], Top-Two approaches maintain all candidates throughout exploration [9], a property TSMOO leverages for robustness to novelty effects.

Non-stationarity in bandit problems has been addressed through sliding window and discounted reward approaches [6], which adapt by restricting or down-weighting historical observations. However, these methods are designed for gradual or adversarial distribution shifts; they do not address the transient novelty effects common in user-facing experiments, where treatment performance temporarily diverges from steady-state behavior before stabilizing. TSMOO addresses this distinct non-stationarity pattern by maintaining all treatments throughout exploration without elimination, allowing posteriors to converge to steady-state estimates rather than discounting early observations.

The stochastically constrained best arm identification framework [21] extends Thompson Sampling (TS) to settings where the optimal arm must satisfy threshold constraints across multiple metrics. Unlike standard TS, this framework utilizes a Top-Two sampling rule (BFAI-TS) to balance exploration between the current feasible leader and its challengers, ensuring that the probability of false selection decays at an exponential rate. While related simulation approaches such as constrained OCBA [17] and constrained ranking and selection [7] address similar problems, they focus on optimizing frequentist sample allocation ratios rather than establishing the asymptotic optimality and posterior convergence guarantees provided by this Bayesian framework. Related work on better-than-control arm identification [18] has studied identifying arms that outperform a known control across subpopulations, though using a weighted-sum multi-metric objective rather than threshold-based constraints, and without batch allocation. Unlike BFAI-TS [21], which operates sequentially on absolute arm rewards, TSMOO adapts the framework to batch allocation with uplift-based better-than-control objectives and a least-infeasible selection strategy for production settings.

Multi-objective bandits have been studied through Pareto front identification [1, 4], which seeks non-dominated arms without prioritization, and feasible arm identification [12, 13], which addresses known polyhedron constraints in both fixed-budget and fixed-confidence settings. TSMOO differs by treating the primary metric as the optimization objective while secondary metrics serve as stochastic constraints, aligning with experimentation where treatments must demonstrate primary impact while satisfying guardrails. Multi-objective optimization in recommendation systems has also been studied through scalarization [8], though requiring pre-specified utility weights rather than the threshold-based constraints used in production experimentation. TSMOO's constraint-based formulation is particularly natural for recommendation platforms, where treatments must simultaneously satisfy guardrails on engagement, revenue, and user experience.

In practice, single-metric Thompson Sampling methods provide novelty robustness through continuous updates without elimination, while the sequential elimination methods introduced by Zhang et al. [22], building on variance-aware sequential halving [15], achieve strong stationary performance but risk premature treatment removal. TSMOO combines continuous learning with multi-constraint handling, adapting the BFAI framework to batch allocation without elimination.

7 Conclusion

TSMOO addresses a key gap in adaptive experimentation by combining Thompson Sampling robustness with multi-objective optimization, grounded in uplift-based reward estimation that simultaneously cancels temporal confounding and aligns exploration with validation criteria. Built on the BFAI theoretical foundation, TSMOO provides principled handling of multiple posterior probability constraints while maintaining novelty robustness through continuous learning without treatment elimination.

Empirical validation demonstrates strong performance: 93-94% success rates in multi-winner scenarios and sustained effectiveness under novelty effects. The algorithm has been designed for production deployment in an industrial adaptive experimentation platform serving recommendation systems at scale.

Future directions include theoretical guarantees for the batch adaptation of BFAI-TS, computational enhancements through variance reduction in Monte Carlo feasibility estimation, and extension to non-Gaussian reward distributions for broader applicability.

References

[1] Peter Auer, Chao-Kai Chiang, Ronald Ortner, and Madalina Drugan. 2016. Pareto front identification from stochastic bandit feedback. In *Artificial intelligence and statistics*. PMLR, 939–947.

[2] William Black, Ercument Ilhan, Andrea Marchini, and Vilda Markeviciute. 2023. Adaptex: a self-service contextual bandit platform. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 426–429.

[3] Olivier Chapelle and Lihong Li. 2011. An Empirical Evaluation of Thompson Sampling. *Advances in neural information processing systems* 24 (2011).

[4] Madalina M Drugan and Ann Nowe. 2013. Designing multi-objective multi-armed bandits algorithms: A study. In *The 2013 international joint conference on neural networks (IJCNN)*. IEEE, 1–8.

[5] Tanner Fiez, Houssam Nassif, Yu-Cheng Chen, Sergio Gamez, and Lalit Jain. 2024. Best of three worlds: Adaptive experimentation for digital marketing in practice. In *Proceedings of the ACM Web Conference 2024*. 3586–3597.

[6] Aurélien Garivier and Eric Moulines. 2011. On upper-confidence bound policies for switching bandit problems. In *International conference on algorithmic learning theory*. Springer, 174–188.

[7] Susan R Hunter and Raghu Pasupathy. 2013. Optimal sampling laws for stochastically constrained simulation optimization on finite sets. *INFORMS Journal on Computing* 25, 3 (2013), 527–542.

[8] Olivier Jeunen, Jatin Mandav, Ivan Potapov, Nakul Agarwal, Sourabh Vaid, Wenzhe Shi, and Aleksei Ustimenko. 2024. Multi-objective recommendation via multivariate policy learning. In *Proceedings of the 18th ACM Conference on Recommender Systems*. 712–721.

[9] Marc Jourdan, Rémy Degenne, Dorian Baudry, Rianne de Heide, and Emilie Kaufmann. 2022. Top two algorithms revisited. *Advances in Neural Information Processing Systems* 35 (2022), 26791–26803.

[10] Shafi Kamalbasha and Manuel JA Eugster. 2020. Bayesian A/B testing for business decisions. In *International Data Science Conference*. Springer, 50–57.

[11] Zohar Karnin, Tomer Koren, and Oren Somekh. 2013. Almost optimal exploration in multi-armed bandits. In *International conference on machine learning*. PMLR, 1238–1246.

[12] Julian Katz-Samuels and Clay Scott. 2018. Feasible arm identification. In *International conference on machine learning*. PMLR, 2535–2543.

[13] Julian Katz-Samuels and Clayton Scott. 2019. Top feasible arm identification. In *The 22nd international conference on artificial intelligence and statistics*. PMLR, 1593–1601.

[14] Ron Kohavi, Diane Tang, and Ya Xu. 2020. *Trustworthy online controlled experiments: A practical guide to a/b testing*. Cambridge University Press.

[15] Anusha Lalitha Lalitha, Kousha Kalantari, Yifei Ma, Anoop Deoras, and Branislav Kveton. 2023. Fixed-budget best-arm identification with heterogeneous reward variances. In *Uncertainty in Artificial Intelligence*. PMLR, 1164–1173.

[16] Tor Lattimore and Csaba Szepesvári. 2020. *Bandit algorithms*. Cambridge University Press.

[17] Loo Hay Lee, Nugroho Artadi Pujowidianto, Ling-Wei Li, Chun-Hung Chen, and Chee Meng Yap. 2012. Approximate simulation budget allocation for selecting the best design in the presence of stochastic constraints. *IEEE Trans. Automat. Control* 57, 11 (2012), 2940–2945.

[18] Yoan Russac, Christina Katsimerou, Dennis Bohle, Olivier Cappé, Aurélien Garivier, and Wouter M Koolen. 2021. A/b/n testing with control in the presence of subpopulations. *Advances in Neural Information Processing Systems* 34 (2021), 25100–25110.

[19] Daniel Russo. 2016. Simple bayesian algorithms for best arm identification. In *Conference on learning theory*. PMLR, 1417–1418.

[20] Dan Siroker and Pete Koomen. 2015. *A/B testing: The most powerful way to turn clicks into customers*. John Wiley & Sons.

[21] Le Yang, Siyang Gao, Cheng Li, and Yi Wang. 2025. Stochastically constrained best arm identification with Thompson sampling. *Automatica* 176 (2025), 112223.

[22] Qining Zhang, Tanner Fiez, Yi Liu, and Wenyang Liu. 2025. Multi-metric adaptive experimental design under fixed budget with validation. *arXiv preprint arXiv:2506.03062* (2025).