A Survey of Multi-Armed Bandit Algorithms: From Theoretical Foundations to Modern Applications

Chaobo Zhang *

Faculty of Science, University of Sydney, Sydney, Australia

* Corresponding Author Email: czha0212@uni.sydney.edu.au

Abstract. The multi-armed bandit (MAB) problem provides a canonical formulation of sequential decision-making under uncertainty, capturing the exploration-exploitation trade-off. This survey charts the intellectual lineage of MAB algorithms, ranging from their statistical roots to contemporary use as part of modern artificial intelligence. We begin by laying the mathematical groundwork for the problem in statistical terms. Building on this, we survey algorithms for the classical stochastic setting, covering simple heuristics such as -greedy to principled approaches based on optimism (Upper Confidence Bound, UCB) and Bayesian inference (Thompson Sampling). We then discuss the contextual bandit as the tipping point in which side information is introduced to enable large-scale personalization, and conclude with recent trends in bandit research, including adaptations to address real-world constraints, non- stationary environments, and the new frontier of fairness considerations. Overall, this review aims to showcase the broad applicability of bandit algorithms to problems ranging from recommendation systems to clinical trials, and argue for the continuing relevance of bandit methods.

Keywords: Multi-armed bandit, Reinforcement learning, Exploration-exploitation, Thompson sampling, Upper Confidence Bound.

1. Introduction

Sequential decision-making under uncertainty is a canonical problem in artificial intelligence and data science, in which an agent must learn a policy, or best course of action, through interaction with an environment [1]. The multi-armed bandit (MAB) problem is the canonical mathematical formulation of this problem. In its classic formulation, an agent must repeatedly pull from a set of K actions, or a set of "arms", with unknown reward distribution, such that the agent maximizes its cumulative reward over time.

The exploration-exploitation dilemma arises naturally as the core issue in online learning. The agent must decide whether to exploit the arm with the highest reward so far or explore another arm, which may provide a much larger reward. An exploited arm agent will likely follow a suboptimal policy. An explored agent will waste opportunities. The goal of any bandit algorithm is to navigate this compromise.

The intellectual history of the MAB problem looks like this: from a problem in the design of adaptive clinical trials (with the objective of reducing the number of patients assigned to the inferior treatment) to an applied artificial intelligence problem. The MAB problem was first formulated as a problem in a class of papers with the objective of designing adaptive clinical trials the objective of designing adaptive clinical trials was to reduce the number of patients assigned to the inferior treatment. It was formalized as a mathematical problem (designing optimal sequential experiments) [1]. It has been a purely mathematical problem for decades.

However, with the rise of the digital age, its basic tenets have finally prospered in real-life large-scale online systems. Bandit algorithms have become the underlying technology that powers many systems, ranging from news recommendation and online pricing to computational advertising and basic research [2]. A growing literature has surveyed the application of these methods to areas like healthcare, finance, dialogue systems.

In this paper we present a structured survey of multi-armed bandit algorithms, ranging from their original conceptual development as simple stochastic models, to the expressive contextual forms that

support contemporary personalization. Specifically, Section 2 lays out the theoretical background for the MAB problem, with respect to the criterion of regret. Section 3 describes algorithms for the classic stochastic setting. Section 4 discusses contextual bandits, an important extension to incorporate side information. Section 5 covers frontiers and open problems in modern research. Finally, in Section 6 we conclude with some remarks on the future of the field.

2. The Foundational MAB Problem

Before looking at the algorithms solving the MAB problem, we need to formally define what is the problem and how to evaluate the performance of the algorithms. This is the formalization of the philosophical exploration-exploitation dilemma.

The stochastic MAB setting is formally defined by a set of K arms, where each arm i is associated with a fixed but unknown probability distribution with mean μ_i. The protocol unfolds over a series of T discrete time steps, or rounds. In each round $t \in \{1, \ldots, T\}$, the agent selects an arm a_t and observes a corresponding reward r_t, which is drawn independently from the distribution of the chosen arm. A key characteristic of this setting is the principle of bandit feedback: the agent only observes the reward for the arm it selected a_t, and remains ignorant of the rewards it would have received from the other arms.

To rigorously compare the performance of different bandit algorithms, the standard metric is cumulative regret. Regret quantifies the opportunity cost of learning—the difference between the cumulative reward of an optimal strategy and the cumulative reward achieved by the algorithm. An optimal strategy would, with foresight, always pull the arm with the highest expected reward $\mu^* = \max_i \mu_i$. The cumulative regret of an algorithm A over T rounds, denoted $R_A(T)$, is formally defined as the expected difference between the optimal reward and the achieved reward:

$$R_A(T) = T\mu^* - E\left[\sum_{t=1}^{T} r_t\right] \quad (1)$$

This can be expressed in terms of the "suboptimality gaps" of the arms, $\Delta_i = \mu^* - \mu_i$, and the expected number of times each suboptimal arm i is pulled, $E[N_i(T)]$ (where $N_i(T)$ represents the number of times arm i is selected in T rounds):

$$R_A(T) = \sum_{i=1}^{K} \Delta_i \, E[N_i(T)] \quad (2)$$

The introduction of regret was a pivotal innovation in the field. It provided a precise mathematical language to analyze a previously abstract trade-off. By establishing a single, quantifiable objective function—regret minimization—it enabled the principled design and rigorous analysis of algorithms. The goal of a sophisticated bandit algorithm is to achieve sub-linear regret, meaning that $R_A(T) = o(T)$. This ensures that the average per-round regret, $R_A(T)/T$, approaches zero as the number of trials T grows infinitely large. The strongest theoretical guarantee demonstrates that any effective algorithm must incur a regret of at least $R_A(T) = \Omega(\log T)$[1, 3]. Consequently, algorithms that achieve a regret bound of $O(\log T)$ are considered asymptotically optimal [3, 4].

3. The Evolution of Stochastic Bandit Algorithms

The classic stochastic bandit problem has given rise to a rich family of algorithms. The progression of these methods reflects an increasing sophistication in how they model and manage uncertainty, evolving from simple probabilistic heuristics to principled strategies based on optimism and Bayesian inference.

3.1. Foundational Heuristics

The most direct approaches to managing the exploration-exploitation trade-off involve simple, intuitive rules that explicitly inject randomness into a greedy strategy. The ϵ-greedy algorithm is a widely used baseline renowned for its simplicity. At each trial, the algorithm acts greedily with probability 1-ϵ by selecting the arm with the highest currently observed average reward. With the remaining probability ϵ, it explores by choosing an arm uniformly at random from all K arms. This is straightforward but the choice of the exploration rate is important. If ϵ is fixed then the algorithm will explore at a constant rate forever and this results in linear regret since it will always pull suboptimal arms with strictly positive probability. To achieve the optimal logarithmic regret bound, a decreasing schedule must be employed, such as $\epsilon_t = O(1/t)$ [5]. This enables the algorithm to explore a lot in early rounds (when the variance is high) and then exploit more as we get more confident in our estimates.

The SoftMax algorithm, also called Boltzmann Exploration, provides a continuous heuristic in between exploring and exploiting. Instead of randomly choosing between exploring and exploiting, we pick arms according to a Gibbs/Boltzmann distribution based on their values. The probability of selecting arm i at trial t is given by:

$$p_i(t) = \frac{e^{\hat{\mu}_i(t)/\tau}}{\sum_{j=1}^{K} e^{\hat{\mu}_j(t)/\tau}} \tag{3}$$

Temperature parameter, $\tau > 0$, controls the level of exploration. High temperature makes selection probabilities more uniform - this encourages exploration, while low temperature puts the majority of the probability mass on the arm with the highest estimated mean, this encourages exploitation. As with ϵ-greedy, a decaying schedule must be used to achieve good asymptotic performance [6].

3.2. Principled Algorithmic Paradigms

On top of simple heuristics, more advanced algorithms were then developed using rigorous statistics to guide the exploration more intelligently. This is an advancement in how the algorithms "reason" about the uncertainty, from blind randomness to model-based reasoning.

A dominant approach is embodied by the Upper Confidence Bound (UCB) family of algorithms, which operate on the principle of "optimism in the face of uncertainty". The core idea is to act as if each arm's true mean is as high as plausibly possible given the evidence collected so far. The canonical UCB1 algorithm implements this by calculating an optimistic upper confidence bound for each arm's mean reward and then greedily selecting the arm with the highest bound [5]. At each trial n, it selects the arm j that maximizes the index:

$$I_n(j) = \bar{x}_j + \sqrt{\frac{2\,ln\,n}{n_j}} \tag{4}$$

Here, $\bar{x}_j$ is the sample mean reward for arm j, representing the exploitation term. $\sqrt{\frac{2\,ln\,n}{n_j}}$ is the exploration bonus, which quantifies the uncertainty in the estimate of the mean. This bonus is large for arms that have been pulled infrequently (small n_j), encouraging the algorithm to explore them. The primary theoretical achievement of UCB1 is its provable logarithmic regret bound, $O(log\,T)$ [3], which establishes it as an asymptotically optimal algorithm without requiring any prior knowledge of the reward distributions.

An alternative and highly effective paradigm is the Bayesian approach, exemplified by Thompson Sampling (TS). First proposed in 1933 but largely overlooked for decades [7] due to computational challenges, TS has seen a modern revival—driven in part by advances in Bayesian computation [8] that have made it applicable to a wide class of reward distributions—and is now celebrated for its

strong empirical performance [9]. The algorithm operates on the principle of posterior probability matching [8]: at each step, it selects an arm according to its posterior probability of being the optimal arm. Operationally, this is achieved by maintaining a posterior probability distribution for the expected reward of each arm i (denoted $P(\mu_i \mid data)$). At each trial, it samples one value $\tilde{\mu}_i$ for each arm from its current posterior and plays the arm with the highest sampled value. This mechanism provides an elegant and natural balance between exploration and exploitation. An arm with high uncertainty (a wide posterior) has a chance of yielding a high sample, leading to exploration. As data accumulates, the posteriors for well-understood arms become narrower and centered around their true means, leading the algorithm to exploit the arm with the highest posterior mean. For binary rewards, a Beta-Bernoulli model is commonly used, where the conjugacy between the Beta prior ($\text{Beta}(\alpha_0, \beta_0)$) and Bernoulli likelihood ($\text{Bernoulli}(\mu_i)$) allows for simple and efficient posterior updates (the posterior becomes $\text{Beta}(\alpha_0 + s_i, \beta_0 + f_i)$, where s_i and f_i are the number of successes and failures for arm i, respectively).

3.3. A Comparative Analysis

The choice between these algorithms involves a nuanced trade-off between theoretical guarantees, computational complexity, and empirical performance. Table 1 provides a concise comparison of their key characteristics.

Table 1. Comparison of Foundational Stochastic Bandit Algorithms

Algorithm	Core Principle	Key Parameter	Regret Bound	Strengths	Weaknesses
ϵ-greedy	Greedy with random exploration	Exploration rate ϵ	Linear (fixed ϵ), $O(\log T)$ (decreasing ϵ)	Simple to implement, robust, empirically effective [6]	Suboptimal for fixed ϵ, requires careful tuning of decay schedule
UCB1	Optimism in the face of uncertainty	Confidence level (implicit)	$O(\log T)$	Asymptotically optimal with strong theoretical guarantees [5], no tuning needed	Can be overly conservative, may over-explore in initial phases, leading to slower convergence
Thompson Sampling	Posterior probability matching	Prior distribution (e.g., Beta for binary rewards)	$O(\log T)$	Excellent empirical performance, naturally balances trade-off, flexible	Requires a probabilistic model and prior specification, can be computationally intensive without conjugacy [8]

A crucial theme that emerges from empirical studies is the gap between asymptotic theoretical guarantees and practical performance in finite-horizon settings. While UCB1 is proven to be asymptotically optimal, extensive empirical studies have shown that well-tuned heuristics like ϵ-greedy are often 'hard to beat' and can outperform theoretically sound algorithms in experiments with a limited number of trials [10]. This discrepancy arises because asymptotic analysis can hide large constant factors in the regret bound. A theoretically optimal algorithm like UCB1 may still be in a slow "warm-up" phase, accumulating significant initial regret through what might be excessive exploration. In contrast, a simpler algorithm, though its exploration is less targeted, may converge to a "good enough" solution more quickly, resulting in lower overall regret within a limited timeframe. This highlights the importance of empirical evaluation in addition to theoretical analysis when choosing an algorithm for a real-world application.

4. Context Bandits

The stochastic bandit framework assumes a stationary environment in which one arm is optimal everywhere. In contrast, the real-world application often of interest may have an optimal action that varies across situation or "context." This shortcoming inspired the formulation of the contextual bandit problem, an extension that ushered the MAB framework from being a method for identifying a single global optimum to being a technology enabling large-scale, adaptive personalization [11].

In the contextual bandit setting, at the start of trial t, the agent measures the feature vector x_t(the context) and then chooses an arm. Now the rewards depend on both the chosen arm and observed context. The goal is to learn an optimal policy $\pi(x)$ that takes contexts x as input and outputs arms, and maximizes reward. This changes the fundamental question from "What is the best action overall?" to "What is the best action for this context?"

Contextual bandits are an intermediate setting between simple stateless MAB and full reinforcement learning [11]. Bandits are stateful since x_t the context, but the agent's action does not affect the sequence of future contexts - making the problem more tractable than full RL while still vastly more powerful than simple MAB.

A prominent and powerful algorithm for this setting is LinUCB, which generalizes the UCB principle to a linear model of reward splementation. This setting is particularly applicable when the feature space is large. LinUCB was also used to show dramatic performance wins in personalized news recommendation systems [12]. LinUCB operates under the assumption that the expected reward of an arm a is a linear function of its feature vector $x_{t,a}$: $E[r_{t,a} \mid x_{t,a}] = x_{t,a}^T \theta_a^*$, where θ_a^* is an unknown weight vector for that arm. The algorithm uses collected data to estimate these weight vectors (typically via ridge regression, which minimizes $\sum_{(x,r)\in D_a}(r - x^T\hat{\theta}_a)^2 + \lambda\|\hat{\theta}_a\|^2$, where D_a is the dataset for arm a and λ is a regularization parameter) and constructs confidence bounds around the estimates. At each trial, it selects the arm that maximizes the UCB of the predicted reward:

$$a_t = arg\max_{a\in A_t}\left(x_{t,a}^T\hat{\theta}_a + \alpha\sqrt{x_{t,a}^T A_a^{-1} x_{t,a}}\right) \quad (5)$$

The first term, $x_{t,a}^T\hat{\theta}_a$, represents the predicted reward (exploitation), while the second term, $\alpha\sqrt{x_{t,a}^T A_a^{-1} x_{t,a}}$, quantifies the uncertainty of that prediction (exploration). Here, $A_a = \sum_{(x,r)\in D_a} x\,x^T + \lambda I$ (I is the identity matrix) is the design matrix for arm a, and α controls the confidence level. The success of LinUCB in a large-scale industrial application provided landmark evidence of the practical value of bandit algorithms, catalyzing a powerful feedback loop. Industry success fueled academic interest and investment [12], which in turn produced more sophisticated models to solve practical challenges discovered during deployment [11].

5. Modern Frontiers and Open Challenges

The evolution of MAB research serves as a proxy for the field's maturity and real-world adoption. Early challenges were primarily mathematical, such as proving regret bounds [4]. Today's challenges are increasingly socio-technical and operational, reflecting the graduation of MAB from a theoretical problem to a deployed technology grappling with the complexities of the real world.

5.1. Addressing Real-World Constraints

Standard bandit models assume the only goal is to maximize a numerical reward. However, practical applications often involve additional constraints. The bandits with knapsacks model extends the MAB framework to scenarios with resource limitations. In this setting, each pull of an arm a consumes a certain amount of one or more resources (denoted $c_{t,a} \in R^d$ for d resources), and the agent must maximize its total reward without exceeding a predefined budget $B \in R^d$ for each

resource. This model was directly motivated by applications like online advertising, where a monetary budget is a primary constraint [2]. Similarly, the combinatorial bandits model addresses situations where the agent must select a set of arms (denoted $S_t \subseteq \{1, \ldots, K\}$) at each round, such as a slate of recommended products or a ranked list of search results. The reward is a function of the entire chosen set ($r_t = f(S_t)$), and the challenge lies in exploring the exponentially large space of possible combinations (2^K possible sets for K arms).

5.2. Adapting to Dynamic Environments

One issue that makes it difficult to apply bandit algorithms in practice for a long time is that real rewards are rarely stationary. Users change their preferences, markets evolve, and content quickly dates. This change over time is what we call concept drift or non-stationarity, and it violates the basic IID assumption of the standard MAB problem [9]. Designing algorithms that are robust to these sorts of changes (not only modeling that they will happen and detecting changes, e.g., changes in $\mu_i(t)$ and applying statistical changes tests, but also adapting behavior, e.g., resetting exploration counts and using a sliding window of recent data) is an important issue that is actively being studied for the long-term success of deployed bandit systems.

5.3. Towards Responsible AI

The classical stochastic bandit problem has yielded many families of algorithms. The evolution of these policies reflects our evolving understanding of, and ability to model and track uncertainty, from simple probabilistic heuristics to more sophisticated forms of optimism and Bayesian updating.

As MABs continue to be deployed in settings where fairness may be a concern (hiring, loan applications, medical treatments, etc.), it is easy to see how a simple bandit algorithm maximizing overall reward may learn to discriminate against different groups (e.g., favoring one group G_1 over another G_2 in reward accumulation), potentially exacerbating existing societal inequalities. This has led to recent work exploring how to incorporate fairness constraints into bandit settings. The goal is to design algorithms that perform well on the primary optimization objective (minimizing $R_A(T)$) while also respecting population-level constraints on how resources or opportunities should be doled out to different arms or sets of users (e.g., ensuring $\sum_{t=1}^{T} I(a_t \in A_{G_1})/|G_1| \approx \sum_{t=1}^{T} I(a_t \in A_{G_2})/|G_2|$ for demographic groups G_1 and G_2). We see this as a natural next step for MAB research given the challenges it presents for responsible and ethical AI [2].

5.4. Bridging the Theory-Practice Gap

The tension between asymptotic theoretical guarantees and finite-horizon empirical performance remains an open and important challenge [10]. The quest for theoretically optimal algorithms has led to exciting advances [5]. However, the proven track record of simple, robust, well-tuned heuristics [6] cannot be forgotten. Future algorithms should strive to combine—strong theoretical support (e.g., $O(\log T)$ regret) and fast practical convergence. These could be more adaptive in their exploration (e.g., adjusting ϵ or α based on real-time uncertainty) or they might leverage prior experience (e.g., a pre-trained model) to warm-up more quickly in the real-world with so few trials.

6. Conclusion

The multi-armed bandit problem has come a long way from its roots in sequential statistical analysis to its position today as a pivotal statistical formulation for modern machine learning offering a coherent approach to the exploration-exploitation challenge. This survey has gone from looking at simple heuristics such as ϵ-greedy to optimal solutions such as UCB and powerful Bayesian approaches such as Thompson Sampling, at each stage looking at the philosophical approach and mathematical formulation taking uncertainty into account.

A major new step forward in the field is contextual bandits, represented by the LinUCB algorithm. Through the introduction of side information (context x_t), contextual bandits converted MAB from a tool to find a single global optimum (μ^*) into a tool to enable large-scale adaptive personalization. This leap unlocked a vast array of applications, from recommender systems that learn individual user tastes to adaptive clinical trials that improve patient outcomes.

The ongoing dialogue between theoretical rigor and practical performance continues to be a key driver of innovation. As the field matures, research frontiers are expanding to address the complex realities of real-world deployment, pushing towards algorithms that are not only optimal (in terms of $O(log\,T)$ regret) but also robust to dynamic environments, mindful of resource constraints (e.g., bandits with knapsacks), and committed to principles of fairness. The multi-armed bandit problem thus remains a rich and vibrant research area at the forefront of the quest to build more intelligent, adaptive, and responsible AI systems.

References

[1] Robbins, H. Some Aspects of the Sequential Design of Experiments. Bulletin of the American Mathematical Society, 1952, 58 (5): 527-535.

[2] Bouneffouf, D., Rish, I., & Aggarwal, C. Survey on Applications of Multi-Armed and Contextual Bandits. In 2020 IEEE Symposium Series on Computational Intelligence (SSCI), 2020.

[3] Lai, T. L., & Robbins, H. Asymptotically efficient adaptive allocation rules. Advances in applied mathematics, 1985, 6 (1): 4-22.

[4] Bubeck, S., & Cesa-Bianchi, N. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. Foundations and Trends® in Machine Learning, 2012, 5 (1): 1–122.

[5] Auer, P., Cesa-Bianchi, N., & Fischer, P. Finite-time Analysis of the Multiarmed Bandit Problem. Machine Learning, 2002, 47: 235–256.

[6] Vermorel, J., & Mohri, M. Multi-armed Bandit Algorithms and Empirical Evaluation. In European Conference on Machine Learning (ECML 2005), 2005: 437-448.

[7] Thompson, W. R. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. Biometrika, 1933, 25 (3/4): 285–294.

[8] Scott, S. L. A modern Bayesian look at the multi-armed bandit. Applied Stochastic Models in Business and Industry, 2010, 26 (6): 639–658.

[9] Russo, D. J., Van Roy, B., Kazerouni, A., Osband, I., & Wen, Z. A Tutorial on Thompson Sampling. Foundations and Trends® in Machine Learning, 2018, 11 (1): 1–96.

[10] Kuleshov, V., & Precup, D. Algorithms for the multi-armed bandit problem. Journal of Machine Learning Research, 2000, 1: 1–48.

[11] Slivkins, A. Introduction to Multi-Armed Bandits. Foundations and Trends® in Machine Learning, 2019, 12 (1-2): 1–286.

[12] Li, L., Chu, W., Langford, J., & Schapire, R. E. A Contextual-Bandit Approach to Personalized News Article Recommendation. In Proceedings of the 19th International Conference on World Wide Web (WWW '10), 2010: 661–670.