Not All Restarts Are Equal: MAB-Learning at the Right Time Scale for SAT

Jinghu Liang
Laboratoire MIS UR 4290, Université de Picardie Jules Verne, Amiens, France

Sami Cherif
Laboratoire MIS UR 4290, Université de Picardie Jules Verne, Amiens, France

Chu-Min Li
Laboratoire MIS UR 4290, Université de Picardie Jules Verne, Amiens, France

Abstract

Multi-Armed Bandit (MAB) mechanisms have proven effective for adaptive heuristic switching in modern CDCL SAT solvers, with Kissat_MAB and its variants demonstrating strong performance in recent SAT Competitions. However, while strategies like the Luby series generate restarts with high duration variability, standard bandit models treat each restart as a homogeneous unit. This mismatch can bias credit assignment and lead to suboptimal exploration–exploitation trade-offs between short and long restarts. In this paper, we study MAB-based heuristic selection under variable-duration restart policies and propose a duration-aware modification to both bandit feedback and selection mechanisms. Our approach normalizes and conditions rewards on the restarts and adapts exploration and exploitation accordingly, thereby better aligning bandit updates with the solver's restart dynamics.

2012 ACM Subject Classification Theory of computation → Constraint and logic programming

Keywords and phrases Satisfiablity, Branching, Restart, Multi-Armed Bandit

Digital Object Identifier 10.4230/LIPIcs.CP.2026.23

1 Introduction

The Boolean satisfiability problem (SAT) is a canonical NP-complete problem, serving as a foundational tool for verification, planning, and combinatorial optimization [2], among others. Modern state-of-the-art SAT solvers are predominantly based on the Conflict-Driven Clause Learning (CDCL) framework [9].

However, no single heuristic in modern SAT solvers dominates across all problem domains, which has motivated the development of adaptive selection mechanisms [6]. `Kissat_MAB` augments this by casting heuristic selection at restarts as a Multi-Armed Bandit (MAB) reinforcement learning problem [3, 10]. In this paradigm, the solver selects a heuristic at the beginning of each restart, executes the restart using the selected heuristic, and updates the heuristic's reward based on the observed progress. This allows the solver to dynamically choose a relevant heuristic at each restart while balancing exploration and exploitation. Since 2021, `Kissat_MAB` and its variants have consistently been ranked among the top solvers in the SAT Competitions[1], notably winning the Main Track in 2021 and 2025 with a wide margin [3, 4, 11].

In this paper, we analyze the bias introduced by variable restart duration in existing MAB approach and propose a duration-aware reinforcement learning approach. Experimental results on the Main Track benchmarks from SAT Competition 2023–2025 show that our approach consistently improves upon highly competitive solvers.

[1] `https://satcompetition.github.io/`

42nd Conference on Very Important Topics (CP 2026).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:3
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 Duration-Induced Bias in MAB for SAT

Existing MAB mechanisms treat all restart rounds as uniform episodes, while restart policies such as the Luby series [8] generate highly variable restart durations. As a result, heuristics used in short restarts and long restarts are weighted identically, which distorts reward assignment and arm selection.

Compared short and long restarts under both the Decision-Variable [3] and Decision-Conflict [7] reward mechanisms, analysis shows that even when the heuristic's quality remains stable, longer restarts systematically receive smaller rewards. Under these two reward schemes, long restarts can be down-weighted in the MAB.

3 Duration-Aware MAB

We propose a Duration-Aware MAB framework to address the bias caused by variable restart durations, with the goal of better capturing and exploiting the relationship between restart duration and heuristic switching. The proposed framework consists of several key components:

- Duration-Aware Reward: Rewards are normalized so that they become comparable across restarts of different Luby durations. This prevents long restarts from being systematically disadvantaged. In addition, A logarithmic damping term is introduced to avoid the bandit becoming overly biased toward longer restarts.
- Duration-based Exploration: Exploration is no longer based only on the number of restart selections. Instead, it considers the total computational budget controlled by a heuristic.
- Adaptive Exploration Coefficient: The exploration coefficient is dynamically adjusted according to the current reward statistics and the duration of the upcoming restart. Upcoming Longer restarts encourage more exploitation and less risky exploration.
- Momentum Exploration Coefficient: A momentum-based mechanism is introduced to boost exploration when recent rewards improve.

Finally, all components are integrated into a new duration-aware UCB framework. The strategy encourages broader exploration during short restarts and stronger exploitation during long restarts.

4 Conclusion

The proposed solver, `DA_kissat_mab`, achieves 4.3% reduction in PAR-2 score, compared with both `Kissat_4.0.2`[1] and `AE_Kissat_MAB`[4], and 4.29% PAR-2 reduction, Compared with the original UCB-based MAB. These results show that the duration-aware mechanism improves heuristic switching and provides more reliable exploration–exploitation behavior than existing MAB frameworks.

As future work, we plan to more tightly couple restart scheduling with heuristic switching. Beyond using restart duration to guide heuristic selection, a natural extension is to adapt the duration of subsequent restarts based on the observed performance of the currently selected heuristic, enabling a joint co-adaptation mechanism. Moreover, current MAB-based selection heuristics typically treat each restart as an independent trial and reward the used heuristic accordingly. It would therefore be relevant to incorporate mechanisms that account for cross-restart dependencies when evaluating heuristic performance. More broadly, we intend to investigate similar component-aware MAB strategies in optimization settings, and particularly within modern Branch-and-bound solvers for MaxSAT [5].

References

1. Armin Biere, Tobias Faller, Katalin Fazekas, Mathias Fleury, Nils Froleyks, and Florian Pollitt. Cadical, gimsatul, isasat and kissat entering the sat competition 2024. *Proc. of SAT Competition*, 8, 2024.
2. Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2021. doi:10.3233/FAIA336.
3. Mohamed Sami Cherif, Djamal Habet, and Cyril Terrioux. Combining VSIDS and CHB using restarts in SAT. In Laurent D. Michel, editor, *27th International Conference on Principles and Practice of Constraint Programming, CP 2021, Montpellier, France (Virtual Conference), October 25-29, 2021*, volume 210 of *LIPIcs*, pages 20:1–20:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. URL: https://doi.org/10.4230/LIPIcs.CP.2021.20, doi:10.4230/LIPICS.CP.2021.20.
4. Hang Ding, Mao Luo, Chu-Min Li, Shunwei Li, Runyao Chen, Caiquan Xiong, and Xinyun Wu. A self-optimizing framework for sat solvers via population evolution and large language model collaboration. In *Proceedings of SAT Competition 2025: Solver and Benchmark Descriptions*, pages 15–17, 2025.
5. Chu-Min Li, Zhenxing Xu, Jordi Coll, Felip Manyà, Djamal Habet, and Kun He. Boosting branch-and-bound maxsat solvers with clause learning. *AI Commun.*, 35(2):131–151, 2022. doi:10.3233/AIC-210178.
6. Jia Hui Liang, Hari Govind V. K., Pascal Poupart, Krzysztof Czarnecki, and Vijay Ganesh. An empirical study of branching heuristics through the lens of global learning rate. In Serge Gaspers and Toby Walsh, editors, *Theory and Applications of Satisfiability Testing - SAT 2017 - 20th International Conference, Melbourne, VIC, Australia, August 28 - September 1, 2017, Proceedings*, volume 10491 of *Lecture Notes in Computer Science*, pages 119–135. Springer, 2017. doi:10.1007/978-3-319-66263-3_8.
7. Jinjin Liu, Jianmin Zhang, and Yan Sun. Kissat mab-dc in sat competition 2024. *Proceedings of SAT Competition 2024 : Solver, Benchmark and Proof Checker Descriptions*, page 20, 2024.
8. Michael Luby, Alistair Sinclair, and David Zuckerman. Optimal speedup of las vegas algorithms. In *Second Israel Symposium on Theory of Computing Systems, ISTCS 1993, Natanya, Israel, June 7-9, 1993, Proceedings*, pages 128–133. IEEE Computer Society, 1993. doi:10.1109/ISTCS.1993.253477.
9. João P. Marques Silva and Karem A. Sakallah. GRASP: A search algorithm for propositional satisfiability. *IEEE Trans. Computers*, 48(5):506–521, 1999. doi:10.1109/12.769433.
10. Zhihui Xie, Xu Liu, and Shuai Li. Multi-armed bandit algorithms for the boolean satisfiability problem: A survey. In *Proceedings of the International Symposium on Combinatorial Search*, volume 18, pages 221–229, 2025.
11. Jiongzhi Zheng, Kun He, Zhuo Chen, Jianrong Zhou, and Chu-Min Li. Combining hybrid walking strategy with kissat mab, cadical, and lstech-maple. *SAT COMPETITION*, 2022:20, 2022.