

Recent Developments in Data-Driven Algorithms for Discrete Optimization

Elias B. Khalil

Department of Mechanical and Industrial Engineering, University of Toronto
SCALE AI Research Chair in Data-Driven Algorithms for Modern Supply Chains
khalil@mie.utoronto.ca

Abstract

The last few years have witnessed a renewed interest in “data-driven algorithm design” (Balcan 2020), the use of Machine Learning (ML) to tailor an algorithm to a distribution of instances. More than a decade ago, advances in algorithm configuration (Hoos 2011) paved the way for the use of historical data to modify an algorithm’s (typically fixed, static) parameters. In discrete optimization (e.g., satisfiability, integer programming, etc.), exact and inexact algorithms for NP-Hard problems often involve heuristic search decisions (Lodi 2013), abstracted as parameters, that can demonstrably benefit from tuning on historical instances from the application of interest.

While useful, algorithm configuration may be insufficient: setting the parameters of an algorithm upfront of solving the input instance is still a static, high-level decision. In contrast, we have been exploring a suite of ML and Reinforcement Learning (RL) approaches that tune iterative optimization algorithms, such as branch-and-bound for integer programming or construction heuristics, *at the iteration level* (Khalil et al. 2016, 2017; Dai et al. 2017; Chmiela et al. 2021; Gupta et al. 2022; Chi et al. 2022; Khalil, Vaezipoor, and Dilkina 2022; Khalil, Morris, and Lodi 2022; Alomrani, Moravej, and Khalil 2022; Cappart et al. 2021; Gupta et al. 2020).

We will survey our most recent work in this area:

1. **New methods for learning in MILP branch-and-bound** (Gupta et al. 2020, 2022; Chmiela et al. 2021; Khalil, Vaezipoor, and Dilkina 2022; Khalil, Morris, and Lodi 2022);
2. **RL for online combinatorial optimization and large-scale linear programming** (Alomrani, Moravej, and Khalil 2022; Chi et al. 2022);
3. **Neural network approximations for stochastic programming** (Dumouchelle et al. 2022).

References

Alomrani, M. A.; Moravej, R.; and Khalil, E. B. 2022. Deep Policies for Online Bipartite Matching: A Reinforcement Learning Approach. *Transactions of Machine Learning Research (TMLR)*.

Balcan, M.-F. 2020. Data-driven algorithm design. *arXiv preprint arXiv:2011.07177*.

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Cappart, Q.; Chételat, D.; Khalil, E. B.; Lodi, A.; Morris, C.; and Veličković, P. 2021. Combinatorial optimization and reasoning with graph neural networks. *arXiv preprint arXiv:2102.09544*.

Chi, C.; Aboussalah, A. M.; Khalil, E. B.; Wang, J.; and Sherkat-Masoumi, Z. 2022. A Deep Reinforcement Learning Framework For Column Generation. In *NeurIPS*.

Chmiela, A.; Khalil, E. B.; Gleixner, A.; Lodi, A.; and Pokutta, S. 2021. Learning to schedule heuristics in branch and bound. *NeurIPS*, 34: 24235–24246.

Dai, H.; Khalil, E. B.; Zhang, Y.; Dilkina, B.; and Song, L. 2017. Learning Combinatorial Optimization Algorithms over Graphs. In *NeurIPS*.

Dumouchelle, J.; Patel, R.; Khalil, E. B.; and Bodur, M. 2022. Neur2SP: Neural Two-Stage Stochastic Programming. In *NeurIPS*.

Gupta, P.; Gasse, M.; Khalil, E. B.; Kumar, M. P.; Lodi, A.; and Bengio, Y. 2020. Hybrid Models for Learning to Branch. *NeurIPS*.

Gupta, P.; Khalil, E. B.; Chételat, D.; Gasse, M.; Bengio, Y.; Lodi, A.; and Kumar, M. P. 2022. Lookback for learning to branch. *Transactions of Machine Learning Research (TMLR)*.

Hoos, H. H. 2011. Automated algorithm configuration and parameter tuning. In *Autonomous search*, 37–71. Springer.

Khalil, E. B.; Dilkina, B.; Nemhauser, G.; Ahmed, S.; and Shao, Y. 2017. Learning to Run Heuristics in Tree Search. In *IJCAI*.

Khalil, E. B.; Le Bodic, P.; Song, L.; Nemhauser, G.; and Dilkina, B. 2016. Learning to Branch in Mixed Integer Programming. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*.

Khalil, E. B.; Morris, C.; and Lodi, A. 2022. MIP-GNN: A Data-Driven Framework for Guiding Combinatorial Solvers. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*.

Khalil, E. B.; Vaezipoor, P.; and Dilkina, B. 2022. Finding backdoors to integer programs: A Monte Carlo Tree Search framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 3786–3795.

Lodi, A. 2013. The heuristic (dark) side of MIP solvers. In *Hybrid metaheuristics*, 273–284. Springer.