references

Chapter 1

[1] Herbert A. Simon, “The structure of ill structured problems,” *Artificial Intelligence* 4, no. 3–4 (1973), 181–201.

Chapter 2

[1] M. Held and R.M. Karp, “A dynamic programming approach to sequencing problems,” *Journal of the Society for Industrial and Applied Mathematics* 10, no. 1 (1962), 196–210.

[2] M.T. Goodrich and R. Tamassia, *Algorithm Design and Applications* (Wiley, 2015), 513–514.

[3] N. Damavandi and S. Safavi-Naeini, “A hybrid evolutionary programming method for circuit optimization,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, 52, no. 5 (2005), 902–910.

[4] P. Jensen and B. Jonathan, *Operations research models and methods* (John Wiley & Sons, 2002).

[5] A. Khamis and M. Ashraf, “A differential evolution-based approach to design all-terrain ground vehicle wheels,” *2017 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC),* (IEEE, 2017).

[6] K. Veselić, “Finite catenary and the method of Lagrange,” *SIAM Review* 37, no. 2 (1995), 224–229.

[7] J. Kalcsics and Roger Z. Ríos-Mercado, “Districting problems,” in Gilbert Laporte, Stefan Nickel, and Francisco Saldanha da Gama (eds.), *Location Science* (Springer, 2019), 705–743.

[8] S.M. Almufti, “Historical survey on metaheuristics algorithms,” *International Journal of Scientific World,* 7, no. 1 (2019), 1.

Chapter 4

[1] M. Leighton, R. Wheeler, and C. Holte, “Faster optimal and suboptimal hierarchical search,” *Proceedings of the Fourth Annual Symposium on Combinatorial Search* (Barcelona, 2011).

[2] P. Sanders and D. Schultes, “Highway hierarchies hasten exact shortest path queries,” *Algorithms: ESA 2005* (Springer, 2005), 568–579.

[3] R. Geisberger et al., “Contraction hierarchies: Faster and simpler hierarchical routing in road networks,” *Experimental Algorithms: WEA 2008* (Springer, 2008), 319–333.

Chapter 5

[1] S. Kirkpatrick, C. Gelatt, and M. Vecchi, “Optimization by Simulated Annealing,” *Science* 220, no. 4598 (13 May 1983), 671–680.

[2] Y. Xiang et al., ”Generalized simulated annealing algorithm and its application to the Thomson model,” *Physics Letters A* 233, no. 3 (1997), 216–220.

[3] L. Ingber, “Adaptive simulated annealing (ASA): Lessons learned,” arXiv cs/0001018 (2000).

[4] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, “Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search,” *Applied Soft Computing* 11, no. 4 (June 2011), 3680-3689.

[5] B. Felgenhauer and F. Jarvis, “Enumerating possible Sudoku grids” (2005).

[6] E. Russell and F. Jarvis, “Mathematics of Sudoku II,” *Mathematical Spectrum* 39, no. 2 (2006), 54–58.

Chapter 6

[1] F. Glover, “Tabu search and adaptive memory programming—advances, applications and challenges,” in *Interfaces in Computer Science and Operations Research*, ed., R. Barr, R. Helgason and J. Kennington (Springer, 1997), 1–75.

[2] F. Glover, M. Laguna, and R. Marti, “Principles of tabu search,” in , T.F. Gonzalez, ed., *Handbook of approximation algorithms and metaheuristics* (Chapman & Hall, 2007), chapter 23.

[3] M. Gendreau, “An introduction to tabu search," in Fred Glover, Gary A. Kochenberger, *Handbook of Metaheuristics* (Springer US, 2003).

[4] R. Battiti and G. Tecchiolli, “Training neural nets with the reactive tabu search,” *IEEE transactions on neural networks* 6, no. 5 (1995), 1185–1200.

Chapter 7

[1] El-Ghazali Talbi, *Metaheuristics: From Design to Implementation* (John Wiley & Sons, 2009).

[2] Julian Blank and Kalyanmoy Deb, “Pymoo: Multi-objective optimization in Python,” *IEEE Access* 8 (2020), 89497–89509.

Chapter 8

[1] K. Deb and R.B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex Systems* 9, no. 2 (1995), 115–148.

[2] K. Deb, K. Sindhya, and T. Okabe, “Self-adaptive simulated binary crossover for real-parameter optimization,” in *Proceedings of the 9th annual conference on genetic and evolutionary computation* (July 2007), 1187–1194.

[3] K. Deb, “Multi-objective optimization using evolutionary algorithms: an introduction,” in L. Wang, A.H. Ng, K. Deb (eds.), *Multi-objective evolutionary optimisation for product design and manufacturing* (Springer, 2011).

[4] E. Zitzler, “Evolutionary algorithms for multiobjective optimization: Methods and applications,” Tik-Schriftenreihe 30 (Swiss Federal Institute of Technology Zurich, 1999).

[5] A.P. Wierzbicki, “The use of reference objectives in multiobjective optimization,” in *Multiple criteria decision making theory and application: Proceedings of the third conference* (Springer 1980), 468–486.

[6] Z. Gaing, “A particle swarm optimization approach for optimum design of PID controller in AVR system,” *IEEE transactions on energy conversion* 19, no. 2 (2004): 384–391.

Chapter 9

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International conference on neural networks (ICNN'95)* 4 (1995), 1942–1948.

[2] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 IEEE international conference on systems, man, and cybernetics: Computational cybernetics and simulation* vol. 5 (1997), pp. 4104–4108.

[3] M. Clerc, "Discrete particle swarm optimization, illustrated by the traveling salesman problem," in G. Onwubolu and B. Babu (eds.), *New optimization techniques in engineering: Studies in fuzziness and soft computing*, vol. 141 (Springer, 2004), 219–239.

[4] A. Ratnaweera, S.K. Halgamuge, and H.C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Transactions on Evolutionary Computation* 8, no. 3 (2004), 240–255.

Chapter 10

[1] Dervis Karaboga, "An idea based on honey bee swarm for numerical optimization," *Technical report TR06*, vol. 200 (Erciyes University, 2005), 1–10.

Chapter 11

[1] S. Dehaene, *How we learn: Why brains learn better than any machine… for now* (Penguin, 2021).

[2] T.M. Mitchell, *Machine learning* (McGraw-Hill, 1997).

[3] P. Domingos, *The master algorithm: How the quest for the ultimate learning machine will remake our world* (Basic Books, 2015).

[4] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intelligent Systems* 24, no. 2 (2009), 8–12.

[5] W. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," arXiv preprint arXiv:1709.05584 (2017).

[6] K. Broadwater and N. Stillman, *Graph Neural Networks in Action* (Manning, 2023).

[7] Oliver Lange and Luis Perez, "Traffic prediction with advanced Graph Neural Networks," *Google DeepMind* blog, www.deepmind.com/blog/traffic-prediction-with-advanced-graph-neural-networks.

[8] A. Vaswani et al., "Attention is all you need," *Advances in neural information processing systems*, 30 (2017).

[9] "The Annotated Transformer," *Harvard NLP*, https://nlp.seas.harvard.edu/2018/04/03/attention.html.

[10] O. Vinyals, M. Fortunato, and N Jaitly, "Pointer networks," arXiv preprint arXiv:1506.03134 (2015).

[11] L. Graham, "An efficient algorithm for determining the convex hull of a finite planar set," *Information Processing Letters* 1 (1972), 132–133.

[12] A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Information Processing Letters* 2 (1973), 18–21.

[13] C. Barber, D.P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," ACM *Transactions on Mathematical Software* 22, no. 4 (1996), 469–483.

[14] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: A methodological tour d'horizon," *European Journal of Operational Research* 290, no. 2 (2021), 405–421.

[15] A. Hottung, S. Tanaka, and K. Tierney, "Deep learning assisted heuristic tree search for the container pre-marshalling problem," *Computers & Operations Research* 113 (2020).

[16] N. Vesselinova et al., "Learning combinatorial optimization on graphs: A survey with applications to networking," *IEEE Access* 8 (2020), 120,388–120,416.

[17] Chengrun Yang et al., "Large Language Models as Optimizers," arXiv preprint arXiv:2309.03409 (2023).

[18] B. Amos, "Tutorial on amortized optimization for learning to optimize over continuous domains," arXiv: 2202.00665 (2022).

[19] S. Cohen, B. Amos, and Y. Lipman, "Riemannian convex potential maps," *Proceedings of the 38th International Conference on Machine Learning, Proceedings of Machine Learning Research* (2021), 2028–2038.

[20] Chaitanya K. Joshi, Thomas Laurent, and Xavier Bresson, "Graph Neural Networks for the Travelling Salesman Problem," from the "Boosting Combinatorial Optimization using Machine Learning" session at the INFORMS annual meeting 2019.

[21] Chaitanya K. Joshi, Quentin Cappart, Louis-Martin Rousseau, Thomas Laurent, code for "Learning the Travelling Salesperson Problem Requires Rethinking Generalization," *27th International Conference on Principles and Practice of Constraint Programming* (CP 2021), GitHub repo, https://github.com/chaitjo/learning-tsp.

[22] Diego Vicente, "Using Self-Organizing Maps to solve the Traveling Salesman Problem," *Diego Vicente* blog post, https://diego.codes/post/som-tsp/.

[23] M. McGough, "Pointer Networks with Transformers," *Towards Data Science* (2021), https://towardsdatascience.com/pointer-networks-with-transformers-1a01d83f7543.

Chapter 12

[1] R. Sutton and A. Barto, *Reinforcement Learning: An introduction*, second edition (MIT Press, 2018).

[2] J. Schulman et al., "Proximal policy optimization algorithms," arXiv preprint (2017), arXiv:1707.06347.

[3] C. Joshi, T. Laurent, and X. Bresson, "Graph Neural Networks for the Travelling Salesman Problem," in Boosting Combinatorial Optimization using Machine Learning, NFORMS Annual Meeting 2019.

[4] M. Nazari, et al., "Reinforcement learning for solving the vehicle routing problem," arXiv preprint (2018), arXiv:1802.04240.

[5] A. Delarue, R. Anderson, and C. Tjandraatmadja, "Reinforcement learning with combinatorial actions: An application to vehicle routing," arXiv preprint (2020), arXiv:2010.12001.

[6] C. Wan, Y. Li, and J. Wang, "RLOR: A flexible framework of deep reinforcement learning for operation research," arXiv preprint (2023), arXiv:2303.13117.

[7] A. Alabbasi, A. Ghosh, and V. Aggarwal, "DeepPool: Distributed model-free algorithm for ride-sharing using deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems* 20, no. 12 (2019), 4714–4727.

[8] C. Jiayu et al., "DeepFreight: A model-free deep-reinforcement-learning-based algorithm for multi-transfer freight delivery," in *Proceedings of the International Conference on Automated Planning and Scheduling* 31 (2021), 510–518.

[9] T. Oda and C. Joe-Wong, "MOVI: A model-free approach to dynamic fleet management," in *IEEE INFOCOM 2018: IEEE Conference on Computer Communications* (2018), 2708–2716.

[10] J. Ruan et al., “Deep reinforcement learning-based traffic signal control,” in *2023 IEEE International Conference on Smart Mobility* (2023), 21–26.

[11] K. Lin et al., “Cooperative variable speed limit control using multi-agent reinforcement learning and evolution strategy for improved throughput in mixed traffic,” in *2023 IEEE International Conference on Smart Mobility* (2023), 27–32.

[12] K. Li and J. Malik, “Learning to optimize,” arXiv preprint (2016), arXiv:1606.01885.

[13] S. Toll et al., “Efficient Elevator Algorithm,” *Chancellor’s Honors Program Projects*, University of Tennessee (2020).

[14] S. Schneider et al., “mobile-env: An open platform for reinforcement learning in wireless mobile networks,” in NOMS 2022–2022 IEEE/IFIP Network Operations and Management Symposium, 2022. GitHub repo: https://github.com/stefanbschneider/mobile-env/tree/main.

Appendix A

[1] C. Guéret, C. Prins, and M. Sevaux, *Linear programming* (HAL, 2000).

Appendix C

[1] Jacques A. Ferland and Gilles Guénette, “Decision support system for the school districting problem,” *Operations Research* 38, no. 1 (1990), 15–21.

[2] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, “Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search,” Applied Soft Computing 11, no. 4 (June 2011), 3680-3689.

[3] N. Kamarudin and M. Ab Rashid, “Modelling of simple assembly line balancing problem type 1 (SALBP-1) with machine and worker constraints,” *Journal of Physics: Conference Series* 1049, no. 1 (2018), p. 012037.

[4] K. Sutner, “Linear cellular automata and the Garden-of-Eden,” *The Mathematical Intelligencer* 11, no. 2 (1989), 49–53.

[5] C. Guéret, C. Prins, and M. Sevaux, *Linear programming* (HAL, 2000).

[6] A. Martin et al., “Local termination criteria for swarm intelligence: A comparison between local stochastic diffusion search and ant nest-site selection,” in *Transactions on Computational Collective Intelligence XXXII* (2019), 140–166.

[7] R. Lovelace, “Open source tools for geographic analysis in transport planning,” *Journal of Geographical Systems* 23 (2021), 547–578.

[8] M. Daskin, *Network and Discrete Location: Models, Algorithms, and Applications*, second edition (John Wiley, 2013).

[9] W. Herbawi and M. Weber, “A genetic and insertion heuristic algorithm for solving the dynamic ridematching problem with time windows,” in *Proceedings of the 14th annual conference on Genetic and evolutionary computation* (2012), 385–392.

[10] A. Traue et al., “Toward a reinforcement learning environment toolbox for intelligent electric motor control,” *IEEE Transactions on Neural Networks and Learning Systems* 33, no. 3 (2020), 919–928. Also see the gym-electric-motor (EGM) package available at https://upb-lea.github.io/gym-electric-motor/index.html.