

cspy: A Python package with a collection of algorithms for the (Resource) Constrained Shortest Path problem

David Torres Sanchez^{1, 2}

1 STOR-i, Lancaster University, UK. 2 SINTEF Digital, Mathematics and Cybernetics

DOI: [10.21105/joss.01655](https://doi.org/10.21105/joss.01655)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Jack Poulson](#) ↗

Reviewers:

- [@hausen](#)
- [@stsievert](#)

Submitted: 15 July 2019

Published: 07 December 2019

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Introduction

When solving the shortest path problem and considering multiple operational restrictions, one may resort to the (resource) constrained shortest path (CSP) problem. It consists, as its name suggests, in finding, among all paths, the shortest path from source to sink nodes that satisfies a set of constraints for a defined set of resources. Such set of resources and the way they evolve throughout the path are user defined and controlled. This allows the modelling of a wide variety of problems including: the vehicle routing problem with time windows, the technician routing and scheduling problem, the capacitated arc-routing problem, on-demand transportation systems, and, airport ground movement (Bode & Irnich, 2014; Chen, Weiszer, Stewart, & Shabani, 2016; Desrochers & Soumis, 1988; Feillet, Dejax, Gendreau, & Gueguen, 2004; Garaix, Artigues, Feillet, & Josselin, 2010; Irnich & Villeneuve, 2006; Righini & Salani, 2008; Tilk, Rothenbächer, Gschwind, & Irnich, 2017; Zamorano & Stolletz, 2017).

cspy is a Python package that allows you to solve instances of the CSP problem using up to eight different algorithms.

cspy is of interest to the operational research community and others that wish to solve an instance of the CSP problem.

Algorithms

Even though the CSP problem is \mathcal{NP} -hard (Garey & Johnson, 1990), several algorithms have been developed to solve it. The most common algorithms are dynamic programming labelling algorithms. Irnich & Desaulniers (2005) presented an exact algorithm based on DP, the monodirectional forward labelling algorithm, based on the pioneering work by Desrochers & Soumis (1988).

Advanced and efficient algorithms have been developed since. Boland, Dethridge, & Dumitrescu (2006) published a state augmenting algorithm that uses a monodirectional labelling algorithm to find an elementary path (one without repeating nodes). Such algorithm has been implemented by Weyens (2018).

Righini & Salani (2006) introduced a bidirectional labelling algorithm for the SPPRC. The bidirectional algorithm is an extension of the monodirectional algorithm that supports search from both ends of the graph, hence, reducing the computational efforts. More recently, Tilk et al. (2017) developed a bidirectional labelling algorithm with dynamic halfway point. The bidirectional search is bounded for both directions and these bounds are dynamically updated as the search in either direction advances. The algorithm has shown to be significantly more efficient than monodirectional ones (Gschwind, Irnich, Rothenbächer, & Tilk, 2018).

Even with some of the most recent algorithms, solving an instance of the SPPRC can be slow, thus, heuristic algorithms have been developed. Local search or metaheuristics start with a given path and perform a series of moves (edge/node deletion, insertion, or exchange) to obtain another feasible path with lower cost. Some metaheuristics developed include, Tabu search (Desaulniers, Lessard, & Hadjar, 2008), hybrid particle swarm algorithm (Marinakis, Migdalas, & Sifaleras, 2017), and, greedy randomised adaptive search procedure (GRASP) (Ferone, Festa, & Guerriero, 2019).

cspy implements several of these recent exact and metaheuristic algorithms including:

- Bidirectional labeling algorithm with dynamic halfway point (Tilk et al., 2017); which includes the bidirectional labeling algorithm with static halfway point, and the monodirectional forward and backward labeling algorithms;
- Tabu search. Adapted from Desaulniers et al. (2008);
- Greedy elimination procedure;
- Greedy Randomised Adaptive Search Procedure (GRASP). Adapted from Ferone et al. (2019);
- Particle Swarm Optimization with combined Local and Global Expanding Neighborhood Topology (PSOLGENT) Marinakis et al. (2017).

Features

A key component of a CSP problem is to define a set of resources and a function to apply for these to evolve through the graph. Such functions are typically referred to as resource extension functions (REFs). The simplest REF is an additive one, where every time an edge is traversed a constant unit of a certain resource is consumed. However, custom and more generic REFs can be used.

cspy allows for custom and generic REFs to be used. Hence, allowing for a custom consumption of the resources through the graph.

Examples

For some real-world examples, see the repository.

Currently included is an implementation of column generation for the aircraft recovery problem.

Acknowledgements

The author gratefully acknowledges the support of the EPSRC funded EP/L015692/1 STOR-i Centre for Doctoral Training.

References

Bode, C., & Irnich, S. (2014). The shortest-path problem with resource constraints with $(k, 2)$ -loop elimination and its application to the capacitated arc-routing problem. *European Journal of Operational Research*, 238(2), 415–426. doi:[10.1016/j.ejor.2014.04.004](https://doi.org/10.1016/j.ejor.2014.04.004)

Boland, N., Dethridge, J., & Dumitrescu, I. (2006). Accelerated label setting algorithms for the elementary resource constrained shortest path problem. *Operations Research Letters*, 34(1), 58–68. doi:[10.1016/j.orl.2004.11.011](https://doi.org/10.1016/j.orl.2004.11.011)

- Chen, J., Weiszer, M., Stewart, P., & Shabani, M. (2016). Toward a More Realistic, Cost-Effective, and Greener Ground Movement Through Active Routing—Part I: Optimal Speed Profile Generation. *IEEE Transactions on Intelligent Transportation Systems*, 17(5), 1196–1209. doi:[10.1109/TITS.2015.2477350](https://doi.org/10.1109/TITS.2015.2477350)
- Desaulniers, G., Lessard, F., & Hadjar, A. (2008). Tabu search, partial elementarity, and generalized k-path inequalities for the vehicle routing problem with time windows. *Transportation Science*, 42(3), 387–404. doi:[10.1287/trsc.1070.0223](https://doi.org/10.1287/trsc.1070.0223)
- Desrochers, M., & Soumis, F. (1988). A generalized permanent labelling algorithm for the shortest path problem with time windows. *INFOR: Information Systems and Operational Research*, 26, 191–212. doi:[10.1080/03155986.1988.11732063](https://doi.org/10.1080/03155986.1988.11732063)
- Feillet, D., Dejax, P., Gendreau, M., & Gueguen, C. (2004). An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks*, 44(3), 216–229. doi:[10.1002/net.20033](https://doi.org/10.1002/net.20033)
- Ferone, D., Festa, P., & Guerriero, F. (2019). An efficient exact approach for the constrained shortest path tour problem. *Optimization Methods and Software*, 0(0), 1–20. doi:[10.1080/10556788.2018.1548015](https://doi.org/10.1080/10556788.2018.1548015)
- Garaix, T., Artigues, C., Feillet, D., & Josselin, D. (2010). Vehicle routing problems with alternative paths: An application to on-demand transportation. *European Journal of Operational Research*, 204(1), 62–75. doi:[10.1016/j.ejor.2009.10.002](https://doi.org/10.1016/j.ejor.2009.10.002)
- Garey, M. R., & Johnson, D. S. (1990). *Computers and intractability; a guide to the theory of np-completeness*. New York, NY, USA: W. H. Freeman & Co.
- Gschwind, T., Irnich, S., Rothenbächer, A.-K., & Tilk, C. (2018). Bidirectional labeling in column-generation algorithms for pickup-and-delivery problems. *European Journal of Operational Research*, 266(2), 521–530. doi:[10.1016/j.ejor.2017.09.035](https://doi.org/10.1016/j.ejor.2017.09.035)
- Irnich, S., & Desaulniers, G. (2005). Column generation. In (pp. 33–65). Springer US. doi:[10.1007/0-387-25486-2_2](https://doi.org/10.1007/0-387-25486-2_2)
- Irnich, S., & Villeneuve, D. (2006). The Shortest-Path Problem with Resource Constraints and k-Cycle Elimination for $k \geq 3$. *INFORMS Journal on Computing*, 18(3), 391–406. doi:[10.1287/ijoc.1040.0117](https://doi.org/10.1287/ijoc.1040.0117)
- Marinakis, Y., Migdalas, A., & Sifaleras, A. (2017). A hybrid particle swarm optimization-variable neighborhood search algorithm for constrained shortest path problems. *European Journal of Operational Research*, 261(3), 819–834. doi:[10.1016/j.ejor.2017.03.031](https://doi.org/10.1016/j.ejor.2017.03.031)
- Righini, G., & Salani, M. (2006). Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization*, 3(3), 255–273. doi:[10.1016/j.disopt.2006.05.007](https://doi.org/10.1016/j.disopt.2006.05.007)
- Righini, G., & Salani, M. (2008). New dynamic programming algorithms for the resource constrained elementary shortest path problem. *Networks*, 51(3), 155–170. doi:[10.1002/net.20212](https://doi.org/10.1002/net.20212)
- Tilk, C., Rothenbächer, A. K., Gschwind, T., & Irnich, S. (2017). Asymmetry matters: Dynamic half-way points in bidirectional labeling for solving shortest path problems with resource constraints faster. *European Journal of Operational Research*, 261(2), 530–539. doi:[10.1016/j.ejor.2017.03.017](https://doi.org/10.1016/j.ejor.2017.03.017)
- Weyens, T. (2018). pylgrim—Elementary Shortest Path Problem with or without Resource Constraint. Retrieved from <https://github.com/ToonWeyens/pylgrim>
- Zamorano, E., & Stolletz, R. (2017). Branch-and-price approaches for the Multiperiod Technician Routing and Scheduling Problem. *European Journal of Operational Research*, 257(1), 55–68. doi:[10.1016/j.ejor.2016.06.058](https://doi.org/10.1016/j.ejor.2016.06.058)