European Journal of Operational Research 292 (2021) 818–829

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

The multiple shortest path problem with path deconfliction

Michael S. Hughes[a,*], Brian J. Lunday[a], Jeffrey D. Weir[a], Kenneth M. Hopkinson[b]

[a] Department of Operational Sciences, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433 USA
[b] Department of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433 USA

ARTICLE INFO

Article history:
Received 6 May 2020
Accepted 21 November 2020
Available online 26 November 2020

Keywords:
Routing
Shortest path problem
Path deconfliction
Multi-objective optimization
Multi-agent

ABSTRACT

To address the increasingly relevant challenge of routing autonomous agents within contested environments, this research formulates and examines the Multiple Shortest Path Problem with Path Deconfliction (MSPP-PD) to balance agent routing efficiency with group vulnerability. Within the general model formulation, multiple agents are routed between respective source and terminus nodes while minimizing both the total distance travelled and a measure of path conflict, where path conflict occurs for any instance of more than one agent traversing an arc and/or node. Within this modeling structure, this research presents and inspects a set of alternative, conceptually-motivated penalty metrics to inhibit path conflict between agents. Illustrative testing demonstrates the distinguishability of different MSPP-PD variants as they relate to optimal agent routing solutions, as well as the non-dominated solutions attainable via different relative priorities over the objective functions. Subsequent empirical testing over a set of synthetic instances demonstrates the effect of different penalty function metrics on both optimal solutions and the computational effort required to identify them. Concluding the work are recommendations about the utility of the MSPP-PD model variants, both individually and collectively.

Published by Elsevier B.V.

1. Introduction

When routing multiple agents over a network from sources to terminus nodes, a common approach is to identify their respective shortest paths. Using such shortest path problem frameworks, each agent selfishly seeks to minimize their total cost, distance, fuel usage, or similar outcome. However, routing agents via their respective shortest paths may not be suitable for certain environments. When routing multiple agents, selfish behavior can yield solutions wherein respective agents' paths conflict. In a coordinated operating environment (e.g., multi-robot path planning and motion coordination (Parker, 2009)), such path conflict can increase the likelihood of inter-agent collisions, failure to complete routes, and/or delays. In a contested environment (e.g., the infiltration of multiple agents towards adversary targets, such as "attritable" vehicles (Department of the Air Force, 2019)), conflicting agent paths allow an adversary to observe – and possibly target or interdict – the agents by observing a limited number of locations where the agents' paths conflict. Alternatively, agents hoping to observe contested environments (e.g., the overnight security problem (Calvo & Cordone, 2003)) find that conflicting agent paths are inefficient.

Therefore, it is important to balance agents' individual motivations with a collective goal of deconflicting their respective paths. This work sets forth, tests, and compares a suite of mathematical programming models, wherein a centralized coordinating entity plans agents' paths to balance the enterprise-level objective of path deconfliction with the agent-level objective of traversing short paths. Furthermore, this work addresses the *spatial* deconfliction of agents' paths for the aforementioned motivating applications. However, it also has use for the related problem of *spatio-temporal* deconfliction of agent routing. To wit, if agents' paths are maximally spatially deconflicted, it is easier to time their respective routing over the network to affect spatio-temporal deconfliction.

1.1. Literature review

Several areas within the published literature inform either the modeling or solution methodologies adopted herein. Given the multi-objective framework, the literature on techniques pertaining multi-objective optimization provide methods to enable our analysis and comparison of model variants when applied to sets of test instances. Within this framework, the modeling is informed by related threads for routing agents over networks. Pertaining to the objective of deconflicting agent paths, published work relating to agent routing with either side constraints or multiple objectives is also informative.

* Corresponding author.
E-mail address: mike6hughes@gmail.com (M.S. Hughes).

https://doi.org/10.1016/j.ejor.2020.11.033
0377-2217/Published by Elsevier B.V.

For the commercial and military applications this work aims to address, decision makers often seek to achieve multiple goals (e.g., minimal resource expenditure, minimal risk). As such, multi-objective optimization models are well suited to represent the myriad of goals. In the absence of an *a priori*, hierarchical prioritization of the multiple objectives, it is of interest to identify solution(s) that are Pareto efficient. Given such a Pareto optimal solution, no improvement can be realized with respect to an objective function value without a degraded outcome for at least one other objective function. Although several techniques exist to explore solutions on a Pareto frontier (e.g., see Ehrgott, 2005), this research utilizes the Weighted Sum Method to solve the underlying multi-objective optimization problem (MOOP) due to its simplicity.

The Weighted Sum Method solves a MOOP by assigning a set of non-negative weights to the respective objective functions to create a weighted sum single-objective function such that the sum of the weights equals 1 (Ehrgott, 2005). The weights correspond to the relative importance or preference of a decision maker regarding each objective function. It is important to consider the scale of each objective function when determining an appropriate set of weights. By varying the set of objective function weights, an iterative solution of the underlying (single-objective) mathematical program can identify a set of alternative solutions and, through the elimination of dominated solutions (e.g., inferior) from this set, a Pareto frontier of non-dominated solutions can be generated.

The problem of efficiently routing agents inherently extends from the *shortest path problem* (SPP), wherein an individual unit of supply is routed via the shortest path from a given origin (i.e., source node) over a network to a particular destination (i.e, terminus node). Dijkstra et al. (1959) developed a basic, efficient algorithm to solve the SPP from a given node to every other node in the network in $O(|A| + |N|\log|N|)$ operations, wherein A is the set of arcs and N is the set of nodes in the network. Subsequently, Dantzig (1960) showed that the shortest path to all nodes in the network from the origin could be discovered via $\frac{n(n-1)}{2}$ comparisons. Due to its significance in routing, the SPP has been well studied and applied across multiple disciplines. A reader interested in SPP algorithms is referred to surveys of SPP algorithms by Dreyfus (1969) and Zhan and Noon (1998).

The related k-SPP has likewise been well studied. The k-SPP attempts to find the 1^{st} through k^{th} shortest paths from a source to a terminus over a network. Hoffnam and Pavley (1959), Bellman and Kalaba (1960), and Sakarovitch (1968) examined the k-SPP variant which allows cycles, whereas the more common k-SPP which does not allow cycles was initially examined by Bock, Kantner, and Haynes (1958), Pollack (1961), Sakarovitch (1968), and Yen (1971). A review of specific k-SPP algorithms is beyond the scope of this research, but an interested reader is referred to an excellent survey by Mohanta and Poddar (2012). The problem explored in this contribution is distinct from the k-SPP because multiple agents are being routed from respective source to terminus nodes while considering multiple, often competing objectives. In a specialized case of the k-SPP that is most closely related to this study, an instance in which all agents must be routed from the same source to the same terminus might traverse the k shortest paths to reduce path conflict because the agents would not all traverse the same path, but there would not be any *specific* consideration for path deconfliction in generating such a solution.

Incorporating multiple objectives into the SPP has been examined in research on both the multi-objective shortest path problem (MOSPP) and the multi-criteria shortest path problem (MCSPP), wherein the objective functions informing an MOSPP model need not be additive, whereas the objective functions for an MCSPP are required to be additive (Reinhardt & Pisinger, 2011). Both problems aim to address multiple goals, but they are distinct in the same manner as multi-critiera decision making (MCDM) and multi-objective decision making (MODM). The MOSPP and MCSPP frameworks are a common tool used to design and operate computer networks (Cidon, Rom, & Shavitt, 1997; Kerbache & Smith, 2000; Silva & Craveirinha, 2004) and transportation systems (Dial, 1979; Disser, Müller-Hannemann, & Schnee, 2008; Modesti & Sciomachen, 1998), as many of these complex systems' performance is based upon multiple, often competing metrics. Because many of these systems have their own unique features, there are numerous formulations and versions of both the MOSPP and MCSPP.

For an MCSPP having two additive, non-negative-valued criteria (i.e., the work explored in this research) and an associated, fixed weighting scheme, solution methods such as Dijkstra's Algorithm will find the optimal path efficiently (Tarapata, 2007). Furthermore, Papadimitriou and Yannakakis (2000) and Vassilvitskii and Yannakakis (2004) demonstrated the ability to efficiently create ε-Pareto curves for problems having two or three criteria, wherein an ε-Pareto curve is a subset of feasible solutions such that, for any Pareto optimal solution, there exists a solution on the ε-Pareto curve that is no more than $(1+\varepsilon)$ away with respect to each of the objective function values. Tarapata (2007) classifies a collection of related work by their number of criteria (i.e., $2, \ldots, n$), type of problem (e.g., types of constraints, network structure), and solution method (i.e., exact, heuristic, metaheuristic). For additional information on MOSPP and, more specifically, MCSPP applications, formulation types, and solution methods, an interested reader is referred to the survey by Tarapata (2007).

Related in the field of agent routing is the travelling salesman problem (TSP), mathematically posed in 1934 by Whitney in a seminar talk at Princeton University (Flood, 1956). Within the TSP, a salesman is given a list of cities and the respective distances between each pair of cities and must discern the shortest possible route to visit each city and returns to the origin city. The TSP has also inspired research to examine more realistic applications such as the sequencing and routing of order pickers in warehouses (Theys, Bräysy, Dullaert, & Raa, 2010) and flowshop scheduling (Bagchi, Gupta, & Sriskandarajah, 2006). Research on the TSP also includes work considering multi-objective frameworks (García-Martínez, Cordón, & Herrera, 2007; Lust & Teghem, 2010; Wang, Guo, Zheng, & Wang, 2015).

Also related to our research is the variant of the TSP, the m-Peripatetic Salesman Problem (m-PSP), a problem to determine m edge disjoint Hamiltonian cycles of minimum total cost on a network. First introduced by Krarup (1975), the m-PSP has been applied to planning routes for security guards (Calvo & Cordone, 2003); robust network designs (De Kort, 1993), wherein to protect the network from link failure, several edges-disjoint cycles must be determined; and job scheduling, wherein each job must be processed twice by the same machine but technological constraints prevent the repetition of identical job sequences (De Kort, 1993).

Finally, within the routing literature there has been expansive research on the Vehicle Routing Problem (VRP), which pertains to the design of optimal deliveries and/or collection routes from one or several depots to a number of geographically disparate locations, subject to side constraints (Laporte, 1992). Since Dantzig and Ramser (1959) inspired the field via the truck dispatching problem, the research threads pertaining to VRP formulation variants and solution techniques have evolved over time (e.g., see Kara, Laporte, & Bektas, 2004; Kulkarni & Bhave, 1985; Paessens, 1988; Potvin & Rousseau, 1993; Van Breedam, 1995).

Additionally, endeavors have extended the underlying VRP framework by considering more nuanced objectives and performance metrics, integrating vehicle routing evaluations with other tactical decisions, and addressing essential aspects of modern supply chains at higher granularities of representation (Vidal, Laporte, & Matl, 2019). More closely aligned with our modeling perspective, recent notable VRP research examines equity-based metrics for agent workload (Lehuédé, Péton, & Tricoire, 2020) and a multi-echelon VRP framework to model urban freight deliveries (Anderluh, Nolz, Hemmelmayr, & Crainic, 2019), with the latter work considering multiple objectives and suitable methods to explore the corresponding tradeoffs among solutions. There has also been work to improve the efficiency of reconnaissance mission planning; such work has sought to minimize the time to search an area and/or maximize area coverage subject to limitations (Stodola & Mazal, 2017; Tian, Shen, & Zheng, 2006), the solutions for which tend to limit path conflict because agents search different regions.

Furthermore, there exists a subset of VRP research pertaining to how agent routes relate to and affect one another. Sörensen (2006) measured the distance between two routing solutions to minimize the change to existing routes that truck drivers would have to make when new stops are added. Clímaco and Pascoal (2009) sought to disperse the traffic of a Multiprotocol Label Switching network by finding non-dominated bicriteria shortest pairs of disjoint simple paths (PDSPs), wherein the criteria considered are additive cost functions (i.e., the cost of using the two paths and the number of arcs in their paths). The authors proposed an algorithm for listing PDSPs by order of cost after an adequate modification in the network topology. Jahn, Möhring, Schulz, and Stier-Moses (2005) addressed the inherent discrimination against some users in system-optimal traffic patterns. They adopt a system optimum approach, but ensure solutions are fair and efficient by imposing additional constraints to ensure that users are assigned to paths within some tolerance of normal length (e.g., traversal time in the uncongested network, traversal time in user equilibrium, geographic distance).

Additionally, there has been work based on the VRP to make cash and hazardous material transportation more secure/safe. For instance, Talarico, Sörensen, and Springael (2015b) developed a model to create safe and efficient routing plans for the cash-in-transit (CIT) sector by considering two critical issues: minimization of the travelled cost/time and limiting the exposure of the transported goods to robbery. Alternatively, Talarico, Sörensen, and Springael (2015a) utilized the k-dissimilar VRP (kd-VRP) to finds k feasible solutions of a single VRP, whereas (Ngueveu, Prins, & Calvo, 2009) and Ngueveu, Prins, and Calvo (2010) examined the m-Perpatetic VRP (m-PVRP) to reduce the frequency cash transporters frequenting the same route. The m-PVRP finds a set of minimal total cost routes over m periods on an undirected graph such that each customer is visited exactly once per period and each edge can be used at most once during the m periods. Zografos and Androutsopoulos (2008) studied the potential benefits of integrating routing and emergency response logistical decisions to improve the effectiveness of the hazardous materials emergency response process. Their proposed system offers alternative, bicriteria distribution routes for hazardous materials, identification of the optimal emergency response unit locations, the subsequent routing of emergency response units, and the identification of optimum evacuation plans. The authors used a heuristic to identify the Pareto frontier of alternative routes for hazardous material routing. Their heuristic leveraged the label setting algorithm of Androutsopoulos and Zografos (2008) to solve the k-shortest path problem with time windows in a time varying network.

Ultimately, none of the various SPP, TSP, or VRP models in the literature is adequate to address our problem. Although many of these model variants route multiple agents, to our knowledge no existing SPP models address yet allow for path conflict between agents. By contrast, the nature of multiple-agent variants of the SPP, TSP, and VRP models provide a conceptual complement to our work, in that they seek to route agents in a manner that provides some type of disparate coverage to nodes in the network. However, any deconfliction of paths resulting from these models is either incidental or strictly enforced and, as such, inadequate to address the deliberate objective of agent path deconfliction considered herein, along with route efficiency.

This research makes two contributions to the literature. First, it formulates and examines a mathematical program to balance agent routing efficiency (i.e., minimizing path length) with group vulnerability to an adversary (i.e., minimizing path conflict). Within this modeling framework, it sets forth alternative measures for the presence of conflict among agents' (s,t)-paths (i.e., source-terminus) over directed networks, specific to either the number of arcs and/or nodes over which multiple agents traverse. Second, it leverages a combination of illustrative and empirical testing, respectively, to demonstrate the math programming formulation variants' ability to induce different agent routing solutions and, in turn, to both identify the corresponding effects on agent routing efficiency and examine the relative efficiency with which optimal solutions are identified. Moreover, it identifies via extended testing the practical limitations of selected model variants for instances having larger networks and/or more agents traversing a given network.

The remainder of this paper is organized as follows. Section 2 presents the model formulations and solution methodology. Section 3 presents illustrative and empirical testing, results, and analysis to examine selected characteristics of the model variant's optimal solutions and the computational effort required to identify them. Finally, Section 4 concludes the work and provides recommendations for future research.

2. Model formulations and solution methodologies

This section formulates a suite of mathematical programming models to address the underlying problem, each of which distinctly measures path conflict via a conceptually-motivated construct, and which is minimized in a multi-objective optimization framework, along with the shortest (s,t)-paths of the respective agents. Subsequent analysis illustrates the formulations' respective *practability* (i.e., practical tractability) (Bertsimas & Dunn, 2019).

2.1. Multiple shortest path problem with path deconfliction

The **Multiple Shortest Path Problem with Path Deconfliction (MSPP-PD)** seeks to route agents between respective source nodes and terminus nodes over a directed network to minimize both the total distance travelled and the degree to which the respective agents' paths conflict.

As a precursor to formulating the model, it is necessary to define the following sets, parameters, and decision variables.

Sets

- K: the set of agents to be routed, indexed by k, where $\mathcal{K} = |K|$
- N: the set of nodes within the network, indexed alternatively by i or j, where $\mathcal{N} = |N|$
  - $S \subseteq N$: the set of source nodes, where s^k is the designated source of agent k
  - $T \subseteq N$: the set of terminus nodes, where t^k is the designated terminus of agent k
- A: the set of directed arcs in the network, indexed by (i,j), where $\mathcal{A} = |A|$
- $G(N,A)$: the directed network

Parameters

- d_{ij}: the non-negative length of arc (i,j) (i.e., the distance from node i to node j)

Decision Variables

- x_{ij}^k: a binary decision variable equal to 1 if agent k traverses arc (i,j) and 0 otherwise

- p: a penalty incurred for conflicts between respective agent paths, computed via a function expressed generically as $g(\boldsymbol{x})$, which varies by the specifics of each metric considered

Given these definitions, we set forth the following model for the MSPP-PD:

$$\min_{\boldsymbol{x},p} \; (f(\boldsymbol{x}), p) \tag{1}$$

$$\text{s.t. } f(\boldsymbol{x}) = \sum_{(i,j)\in A} \sum_{k\in K} d_{ij} x_{ij}^k, \tag{2}$$

$$p = g(\boldsymbol{x}), \tag{3}$$

$$\sum_{j:(i,j)\in A} x_{ij}^k - \sum_{j:(i,j)\in A} x_{ji}^k = \begin{cases} 1 & \text{if } i = s^k, \\ -1 & \text{if } i = t^k, \\ 0 & \text{otherwise,} \end{cases} \qquad \forall k \in K, \tag{4}$$

$$x_{ij}^k \in \{0,1\}, \qquad \forall (i,j) \in A, k \in K. \tag{5}$$

For the (potentially) competing objectives of minimizing total distance travelled and minimizing path conflict, a multi-objective optimization framework is appropriate. Constraint (2) computes the first objective function, the sum of (s_k, t_k)-path lengths over all k agents. Constraint (3) computes the second objective function, a measure of the conflict between respective agent paths, via a generalized function $g(\boldsymbol{x})$. Subsequent to this discussion, we present and illustrate conceptually-motivated, alternative representations for $g(\boldsymbol{x})$, a functional representation we modify with subscript and superscript notation specific to each metric. Constraint (4) enforces conservation of flow at nodes such that each agent, $k \in K$, has a feasible (s_k, t_k)-path, and Constraint (5) enforces binary restrictions on the x_{ij}^k variables. Note that, in the absence of Constraint (3), the resulting single-objective formulation solves the shortest path problem for each of the k agents a formulation we hereafter refer to as the **MSPP**. Relative to the solving MSPP via a commercial solver, an optimal solution can be rapidly attained via application of Dijkstra's Algorithm to identify each of the k agents' shortest paths.

2.2. Penalty functions for the MSPP-PD

To minimize path conflict, alternative penalty function metrics merit consideration. Path conflict can be assessed as it pertains to agents' paths when they utilize the same arcs, the same nodes, or both. Note that penalizing agent paths that share an arc effectively penalizes their use of the nodes at both the head and tail of that arc, but it does not preclude node sharing in the absence of arc sharing. By comparison, node-specific penalties effectively penalize agents who share arcs entering or departing a common node. From a modeling perspective, the latter metric is attractive, but we test both penalty structures to assess their relative tractability. (The models herein restrict consideration to the first two options, although they are readily adaptable to simultaneously consider both types of path conflict.)

The degree of penalty assessed for a given (node or arc) conflict may also vary. Conflict may be of concern simply for occurring (i.e., via a binary metric), for occurring in relation to the number of agents for a given path conflict (i.e., via a linear metric), or for occurring in relation to the number of potential interactions between agents for a given conflict (i.e., via a quadratic metric).

2.3. Binary path deconfliction penalties

Binary path deconfliction penalties relate to *whether* agents' paths conflict on an arc and/or node. (i.e., these penalties do not consider the magnitude of the path conflict). For instance, if $2, \ldots, n$ agents use arc (i,j) on their respective (s^k, t^k)-paths, an arc binary penalty equal to 1 will be incurred.

To penalize agents' path conflict on arcs in a binary fashion (i.e., to invoke arc binary penalties (ABP)), we introduce the binary decision variable ψ_{ij} that is equal to 1 if more than one agent traverses arc (i,j) on their respective (s^k, t^k)-paths, and 0 otherwise. Augmenting the MSPP formulation in (1)–(5), the penalty function and associated constraints in the MSPP-PD(ABP) variant are as follows:

$$g_{total}^{arc-b}(\boldsymbol{x}) = \sum_{(i,j)\in A} \psi_{ij}, \tag{6}$$

$$\frac{1}{\mathcal{K}} \left[\left(\sum_{k\in K} x_{ij}^k \right) - 1 \right] \le \psi_{ij}, \qquad \forall (i,j) \in A, \tag{7}$$

$$\psi_{ij} \in \{0,1\}, \qquad \forall (i,j) \in A. \tag{8}$$

Constraint (6) calculates the total arc-specific binary penalties over the network. Constraint (7) ensures that ψ_{ij} equals 1 if more than one agent traverses arc (i,j). When the penalty is minimized, arcs traversed by at most one agent will yield $\psi_{ij}^* = 0$. Constraint (8) restricts ψ_{ij} to be binary-valued.

To penalize agents' path conflict on nodes in a binary fashion (i.e., to invoke node binary penalties (NBP)), we define two binary decision variables: r_i^k to be equal to 1 if agent k traverses node i, and 0 otherwise; and ζ_i to be equal to 1 if more than one agent traverse node i, and 0 otherwise. In the MSPP-PD(NBP) variant, we augment the MSPP formulation in (1)–(5) with the following penalty function and associated constraints.

$$g_{total}^{node-b}(\boldsymbol{x}) = \sum_{i\in N} \zeta_i, \tag{9}$$

$$r_i^k \ge x_{ij}^k, \qquad (i,j) \in A, \; k \in K, \tag{10}$$

$$r_i^k \ge x_{ji}^k, \qquad (i,j) \in A, \; k \in K, \tag{11}$$

$$\frac{1}{\mathcal{K}} \left[\sum_{k\in K} \left(r_i^k - 1 \right) \right] \le \zeta_i, \qquad \forall i \in N, \tag{12}$$

$$r_i^k \in \{0,1\}, \qquad \forall i \in N, k \in K, \tag{13}$$

$$\zeta_i \in \{0,1\}, \qquad \forall i \in N. \tag{14}$$

Constraint (9) computes the total node-specific binary penalties over the network. Constraints (10) and (11) collectively ensure that r_i^k is equal to 1 if agent k either arrives at or departs from node i. Constraint (12) requires ζ_i to be equal to 1 if more than one agent traverses node i. Constraints (13) and (14) enforce binary restrictions on the associated variables.

2.4. Linear path deconfliction penalties

Linear path deconfliction penalties consider the *number of agents* involved in a given path conflict (i.e., the magnitude of path conflict). For instance, an arc that is utilized by $2, \ldots, n$ agents would accrue a respective penalty of $1, \dots, n-1$. Penalties of this nature are intended to reduce more significantly path conflict for multiagent routing (i.e., with three or more agents) than the binary penalty functions.

To linearly penalize the number of agents traversing an arc (i.e., to invoke arc linear penalties (ALP)), we define the binary variable ϵ_{ij} to be equal to 1 if arc (i, j) is traversed by any agent, and 0 otherwise. The proposed penalty function and associated constraints to augment the MSPP in (1)–(5) to create the MSPP-PD(ALP) are as follows:

$$g^{arc-l}_{total}(\boldsymbol{x}) = \sum_{(i,j)\in A}\left(-\epsilon_{ij} + \sum_{k\in K} x^k_{ij}\right), \tag{15}$$

$$\frac{1}{\mathcal{K}}\sum_{k\in K} x^k_{ij} \le \epsilon_{ij} \le \sum_{k\in K} x^k_{ij}, \qquad \forall (i,j)\in A, \tag{16}$$

$$\epsilon_{ij} \in \{0,1\}, \qquad \forall (i,j)\in A. \tag{17}$$

Constraint (15) computes the total arc-specific penalties resulting from agents' path conflict over the network. Constraint (16) requires ϵ_{ij} to equal 1 if any agent traverses arc (i, j), and 0 otherwise. Finally, Constraint (17) enforces binary restrictions on the ϵ_{ij}-variables.

To linearly penalize path conflict on a node (i.e., to invoke node linear penalties (NLP)), we define the variable θ_i to be equal 1 if any agent traverses node i, and 0 otherwise. Thus, the proposed penalty function and associated constraints to augment (1)–(5) in the MSPP-PD(NLP) variant are as follows:

$$g^{node-l}_{total}(\boldsymbol{x}) = \sum_{i\in N}\left(-\theta_i + \sum_{k\in k} r^k_i\right), \tag{18}$$

Constraints (10), (11) and (13),

$$\frac{1}{\mathcal{K}}\sum_{k\in K} r^k_i \le \theta_i \le \sum_{k\in K} r^k_i, \qquad \forall i\in N, \tag{19}$$

$$\theta_i \in \{0,1\} \qquad \forall i\in N. \tag{20}$$

Constraint (18) calculates the total penalty incurred by agents traversing common nodes over the network. Constraint (19) enforces θ_i to be equal to 1 if any agent traverses node i, and 0 otherwise. Finally, Constraint (20) enforces binary restrictions on the θ_i-variables.

2.5. Quadratic path deconfliction penalties

To penalize arc-specific path conflict in relation to the *number* of potential interactions between the agents (i.e., to invoke arc quadratic penalties (AQP)), we set forth penalty function (21). The level of penalty function (21) is quadratically proportional to the number of agents traversing arc (i, j). Such an effect is realized by two agents sharing an arc incurring a penalty of 1, three agents incurring a penalty of 3, and n agents sharing an arc incurring a penalty of $\frac{n(n-1)}{2}$ (i.e., the Handshaking Lemma (Yoon & Kim, 2006)). The penalty response is appropriate when considering the number of paired interactions between agents traversing an arc; as more agents traverse the arc, that arc becomes more significant – and vulnerable to adversary observation and targeting – within a contested environment.

$$g^{arc-q}_{total}(\boldsymbol{x}) = \sum_{(i,j)\in A}\sum_{k\in K}\sum_{\substack{k'\in\{K| \\ k>k'\}}} x^k_{ij}x^{k'}_{ij} \tag{21}$$

Penalty function (21) can be linearized by introducing the decision variable $z^{kk'}_{ij}$, where $z^{kk'}_{ij}$ is equal to 1 if both x^k_{ij} and $x^{k'}_{ij}$ are equal to 1, and 0 otherwise. Accordingly, we define the following constraints in lieu of Constraint (21) to complement Eqs. (1)–(5) to formulate the MSPP-PD(AQP) variant:

$$g^{arc-q}_{total}(\boldsymbol{x}) = \sum_{(i,j)\in A}\sum_{k\in K}\sum_{\substack{k'\in\{K| \\ k>k'\}}} z^{kk'}_{ij}, \tag{22}$$

$$z^{kk'}_{ij} \le x^k_{ij}, \qquad \forall (i,j)\in A\,;\, k,k'\in K|k>k', \tag{23}$$

$$z^{kk'}_{ij} \le x^{k'}_{ij}, \qquad \forall (i,j)\in A\,;\, k,k'\in K|k>k', \tag{24}$$

$$z^{kk'}_{ij} \ge x^k_{ij} + x^{k'}_{ij} - 1, \qquad \forall (i,j)\in A\,;\, k,k'\in K|k>k', \tag{25}$$

$$z^{kk'}_{ij} \ge 0, \qquad \forall (i,j)\in A\,;\, k,k'\in K|k>k'. \tag{26}$$

Constraint (22) calculates the total arc-specific, quadratic path penalties over the network via a linear formula. Constraints (23) and (24) collectively ensure that $z^{kk'}_{ij}$ will equal 0 if either x^k_{ij} or $x^{k'}_{ij}$ are 0, whereas Constraint (25) requires that $z^{kk'}_{ij}$ will take on a value of 1 if both agent k and k' traverse the arc. Constraint (26) is a non-negativity constraint for $z^{kk'}_{ij}$-variables. (Although $z^{kk'}_{ij}$ are binary decision variables, the binary nature is enforced via Constraints (23)–(25), so a non-negativity restriction in Constraint (26) is sufficient.)

To quadratically penalize multiple agents traversing the same node (i.e., to invoke node quadratic penalties (NQP)), we again use r^k_i with the associated penalty function and constraints are as follows, wherein Constraint (27) computes the total node-specific, quadratic path penalties over the network.

$$g^{node-q}_{total}(\boldsymbol{x}) = \sum_{i\in N}\sum_{k\in K}\sum_{\substack{k'\in\{K| \\ k>k'\}}} r^k_i r^{k'}_i, \tag{27}$$

Constraints (10),(11) and (13).

The aforementioned formulation of the node quadratic penalty function can be linearized by defining $w^{kk'}_i$ to be equal to 1 if the paths for agent k and k' both include node i, and 0 otherwise. The linearized constraint representation follows, yielding a the MSPP-PD(NQP) variant when augmenting Eqs. (1)–(5).

$$g^{node-q}_{total}(\boldsymbol{x}) = \sum_{i\in N}\sum_{k\in K}\sum_{\substack{k'\in\{K| \\ k>k'\}}} w^{kk'}_i \tag{28}$$

Constraints (10),(11),(13),

$$r^k_i \le \sum_{(i,j)\in A} x^k_{ij} + \sum_{(j,i)\in A} x^k_{ji}, \qquad \forall i\in N, k\in K \tag{29}$$

$$w^{kk'}_i \le r^k_i, \qquad \forall i\in N;\ k,k'\in K|k>k', \tag{30}$$

$$w^{kk'}_i \le r^{k'}_i, \qquad \forall i\in N;\ k,k'\in K|k>k', \tag{31}$$

$$w^{kk'}_i \ge r^k_i + r^{k'}_i - 1, \qquad \forall i\in N;\ k,k'\in K|k>k', \tag{32}$$

$$w^{kk'}_i \ge 0, \qquad \forall i\in N;\ k,k'\in K|k>k'. \tag{33}$$

Constraint (28) calculates the total node-specific, quadratic path penalties over the network via a linear formula. Constraint (29) forces r^k_i to be 0 if agent k either enters or leaves node i. Constraints (30) and (31) collectively require $w^{kk'}_i$ to be 0 if either r^k_i or $r^{k'}_i$ equal 0, whereas Constraint (32) requires that $w^{kk'}_i$ equal 1 if both agent k and k' traverse node i. Constraint (33) is a non-negativity constraint for $w^{kk'}_i$-variables.

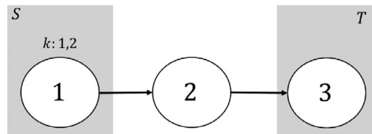

Fig. 1. $G(N, A)$ instance to show lack of TU property in arc-specific quadratic path conflict penalty MSPP-PD variant.

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 2. Constraint matrix for the MSPP-PD(AQP) on instance depicted in Fig. 1. The square submatrix defined by the entries in Rows 3,4,9,10, and 11 and Columns 1,2,3,4, and 6 (i.e., the submatrix boxed in red) has a determinant which equals 2, indicating a lack of total unimodularity.

2.5.1. *On the total unimodularity of constraint matrices in MSPP and MSPP-PD formulation variants*

A beneficial property of many network flow problems is a constraint matrix that is totally unimodular (TU). For such a TU problem structure, if the parameters in the right-hand-side of the constraints of an instance are integer-valued, the binary restrictions on arc-routing decision variables can be relaxed, and the optimal solution to the relaxed problem will assuredly be integer (binary) feasible (Wolsey & Nemhauser, 1999). An $m \times n$ integral matrix A is TU if the determinant of each square submatrix of A is equal to 0,1, or -1 (Wolsey & Nemhauser, 1999).

The MSPP has a TU constraint matrix because, for an n-agent routing problem, it can be decomposed into n shortest path problems, each of which has a TU constraint matrix (Wolsey & Nemhauser, 1999). However, a closer examination is warranted to assess whether the TU property exists for the MSPP-PD variants.

2.6. *Arc-specific path conflict penalty constraints*

The arc-specific path conflict penalty constraints – binary, linear and quadratic – require all x_{ij}^k-variables to be binary-valued. For the MSPP-PD having either binary or linear penalty frameworks (i.e., MSPP-PD(ABP) or MSPP-PD(ALP)), Constraint (7) causes elements of the constraint matrix to be equal to $\frac{1}{K}$; columns associated with x_{ij}^k-variables become fractional. Therefore, the MSPP-PD(ABP) nor MSPP-PD(ALP) manifest the TU property.

For the MSPP-PD having arc-specific quadratic path conflict penalty constraints (i.e., MSPP-PD(AQP)), all entries of the constraint matrix are 0, 1, or -1, so an example is necessary to show the TU property does not hold. Let $G(N, A)$ be the network depicted in Fig. 1, and let K be a set of two agents such that $s_1 = s_2 = 1$ and $t_1 = t_2 = 3$. Fig. 2 displays the constraint matrix for the MSPP-PD(AQP), wherein the first six columns respectively correspond to the decision variables x_{12}^1, x_{12}^2, x_{23}^1, x_{23}^2, z_{12}^{12}, and z_{23}^{12}, and the last six columns relate to slack variables, collectively representing the constraint matrix when Constraints (4) and (23)–(25) are expressed in the form of equality constraints. Of note, the first six rows are flow balance constraints, and the final six rows relate to computing the arc-specific quadratic path conflict penalty constraints. Considering the square submatrix defined by the entries in Rows 3,4,9,10, and 11 and Columns 1,2,3,4, and 6 (i.e., the submatrix boxed in red), the determinant equals 2. Therefore, the constraint matrix for the MSPP-PD(AQP) is not TU.

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 3. Constraint matrix for the MSPP-PD(NQP) on instance depicted in Fig. 1. The square submatrix defined by the entries in Rows 3 and 11 and Columns 1 and 3 (i.e., the submatrix boxed in red) has a determinant which equals -2, indicating a lack of total unimodularity.

2.7. *Node-specific path conflict penalty constraints*

Within the node-specific MSPP-PD formulations, the decision variable r_i^k is used in the same manner as the decision variable x_{ij}^k within the arc-specific path conflict penalty constraints. As such, TU does not hold for the corresponding binary and linear penalty metrics (i.e., MSPP-PD(NBP) or MSPP-PD(NLP)) due respectively to Constraint (12). Since it is possible to isolate x_{ij}^k-variables via r_i^k-variables, to show that the constraints for the node-specific quadratic path conflict penalty (i.e., MSPP-PD(NQP)) are not TU, it is necessary to determine whether the constraint matrices relating the x_{ij}^k-variables to r_i^k-variables are TU. Using the MSPP-PD instance depicted in Fig. 1, the constraint matrix relating x_{ij}^k and r_i^k for the MSPP-PD(NQP) is seen in Fig. 3, wherein the first six columns respectively correspond to $x_{12}^1, x_{12}^2, x_{23}^1, x_{23}^2, r_2^1$, and r_2^2, and the last six columns represent slack variables in Constraints (10), (11), and (29) when expressed in the form of equality constraints. The first six rows are flow balance constraints, and the last six rows correspond to Constraints (10), (11), and (29). Considering the square submatrix defined by the entries in Rows 3 and 11 and Columns 1 and 3 (i.e., the submatrix boxed in red), the determinant equals -2. Therefore, the constraint matrix for the MSPP-PD(NQP) is not TU. Additionally, the example implies that all submatrices of this instance of the MSPP-PD(NQP) related to x_{ij}^k are not TU.

On the TU Property of Submatrices of Node-specific Binary and Linear Path Conflict Penalty Constraints (i.e., MSPP-PD(NBP) and MSPP-PD(NLP))

Although the constraint matrices for the node-specific path conflict penalties are not TU, there is an interesting property that emerges for submatrices of the node-specific linear and binary path penalty constraints (i.e., MSPP-PD(NBP) and MSPP-PD(NLP)). Since it is possible to isolate x_{ij}^k-variables via r_i^k-variables, it is necessary to determine whether the submatrices relating the x_{ij}^k-variables to r_i^k-variables are TU for the MSPP-PD(NBP) and MSPP-PD(NLP) variants because the existence of such a property would allow the relaxation of the binary restriction on x_{ij}^k and assure faster convergence to an optimal solution by conventional algorithms (i.e., branch and bound). Considering the constraint matrix in Fig. 3, if Rows and Columns 11 and 12 are removed, the resulting constraint matrix represents the constraints for the MSPP-PD(NBP) and MSPP-PD(NLP). Within the augmented matrix is a representation of constraints related to x_{ij}^k-variables and their relationship with r_i^k-variables (i.e., flow balance constraints and r_i^k

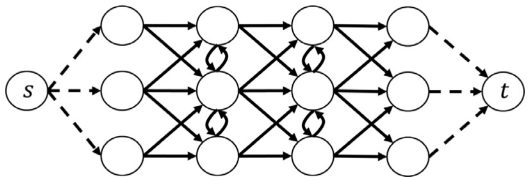

Fig. 4. Example topology of a 3 × 4 grid network (Israeli & Wood, 2002).

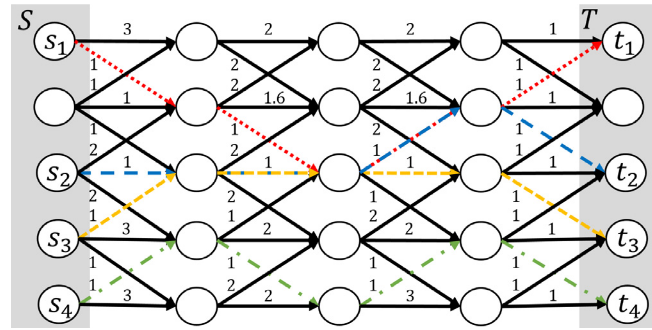

Fig. 5. Optimal solution to a 4-agent $(m, n) = (5, 5)$ MSPP instance with $f(\boldsymbol{x}^*) = 16$.

definition constraints). This matrix is TU, and since the instance is generalizable to all networks because the matrix's form would not change based on the addition of nodes, arcs, or agents, the binary restrictions on x_{ij}^k can be relaxed for the MSPP-PD(NBP) and MSPP-PD(NLP) formulations.

3. Computational and model variant comparison experiments

Several different experiments for the model variants were conducted to derive relative insights. Section 3.1 identifies and supports the network topology adopted for test instances. Section 3.2 examines a synthetic scenario to demonstrate the comparative value of the impact of both different MSPP-PD model variants and relative objective function weights on the optimal solution. Section 3.3 inspects the effect network congestion has on the optimal solutions' total distance for different MSPP-PD model variants. Finally, Section 3.4 investigates the practability of the different MSPP-PD model variants.

3.1. Test instance network topology

To derive empirical insights about the MSPP-PD variants, it is appropriate to generate and solve synthetic instances. The motivating problem of routing agents need not occur on a directed network, but we formulate the MSPP-PD on such a network because any two- or three-dimensional space can be reasonably decomposed into a mesh of nodes, wherein adjacent nodes are connected by arcs to form a planar graph. Applying this technique to a two-dimensional space with regularly-spaced nodes entails a spatial tessellation. A *regular* tessellation consists of decomposing a two-dimensional space into identically sized shapes (e.g., squares, triangles, or hexagons (Israeli & Wood, 2002; Yousefi & Donohue, 2004)). Small, regular shapes yield a fine tessellation, which correspond to a larger network instance that may be computationally challenging to solve, whereas larger regular shapes entail a smaller instance that may not accurately represent the granularity of routing decisions that sufficiently represents planar space. Depending on the threat to agents within the contested environment, an appropriate internodal distance for tessellation may be inferred, where closer nodes could be simultaneously observed and/or targeted by an adversary.

In the literature, there exist precedents for a grid or hexagonal tessellation of a two-dimensional space. Israeli and Wood (2002) utilized a $m \times n$ grid network topology framework to test their Shortest-Path Network Interdiction model, wherein m is the number of rows and n is the number of columns between a source s and terminus t. Such a framework has been adopted by others, such as Lunday and Sherali (2012). An example of a 3 × 4 grid network can be seen in Fig. 4. Note that, in Fig. 4, it is possible to have agents traverse two arcs that cross but which do not incur a penalty via the model formulation. Such an outcome can be prevented by transforming the network; wherever arcs cross, one can add a node and decompose the each of the corresponding arcs into two shorter arcs that respectively terminate and begin at the new node).

For this research, we adapt the network topology used by Israeli and Wood (2002) with a variation to make the network acyclic. Although the MSPP-PD model variants can address instances with networks having cycles, testing is restricted to directed acyclic graphs (DAGs) for two reasons. First, DAGs are most appropriate primary motivating problem for this research. Among the motivations discussed in Section 1 for this study, of foremost interest to this research is the routing of semi-autonomous munitions (i.e., agents) from launch sites (i.e., sources) towards a set of targets (i.e., terminus nodes). In practice, agents in such a problem are routed from one region of a network to another and are unlikely to make use of cycles. Second, no individual agent will traverse a cycle; doing so would yield no improvement to either objective function.

3.2. Comparison of optimal solutions for MSPP-PD variants and relative objective function weights

To illustrate how the MSPP-PD model variants and relative objective function weights affect optimal solutions, specifically on acyclic grid networks, we consider a synthetic instance having four agents traversing a 5 × 5 network. Defining w_d and w_p as the respective weights on the distance- and penalty-related objective functions, Fig. 5 depicts the network topology and arc lengths, with the respective agents' paths in the optimal solution to the MSPP identified via different colored-and-dashed/dotted paths. This optimal solution to the MSPP has a total minimum distance of 16 and has both arc-and-node conflicts among the paths traversed by Agents 1, 2, and 3. Of note, there do exist alternative optimal solutions to the MSPP having varying degrees of path conflict; Fig. 5 shows the reported solution attained via the GAMS modeling language and CPLEX (Version 12.8.0.0) commercial solver. Examining optimal solutions to the MSPP-PD for the same instance, the optimal agent paths differ according to both the type of penalty metric and the relative emphasize on the minimizing penalties vis-á-vis total distance travelled. Herein, this research presents a subset of the testing results to maintain brevity while illustrating the compelling outcomes.

Considering the optimal MSPP solution depicted in Fig. 5, a decision-maker may seek to deconflict the agents' paths while maintaining the total distance of 16. In such a scenario, solving the MSPP-PD with the binary penalty variant can generate alternative routing solutions having lesser path conflict. For instance, if concerned about conflict on the arcs, the optimal solution to the MSPP-PD(ABP) with equal weighting for the objective functions identifies the routing solution depicted in Fig. 6, having a total distance of 16 and a penalty of 1. By comparison, the routing depicted in Fig. 5 incurs a penalty of 2 as a suboptimal solution to MSPP-PD(ABP).

In an alternative scenario, wherein a decision-maker is interested in minimizing both objective functions but their relative priorities are not known *a priori*, it is of interest to identify sets of

Table 1
Non-dominated solutions to a 4-agent, $(m, n) = (5, 5)$ MSPP-PD(NQP) instance for $w_p \in [0.01, 0.99]$ and $w_d = 1 - w_p$.

Solution	w_p	$f(\boldsymbol{x}^*)$	p^*	Optimal Set of Paths
1	[0.01,0.20]	16	5	See Fig. 5
2	[0.21,0.54]	17	1	See Fig. 7
3	[0.55,0.99]	18.2	0	See Fig. 8

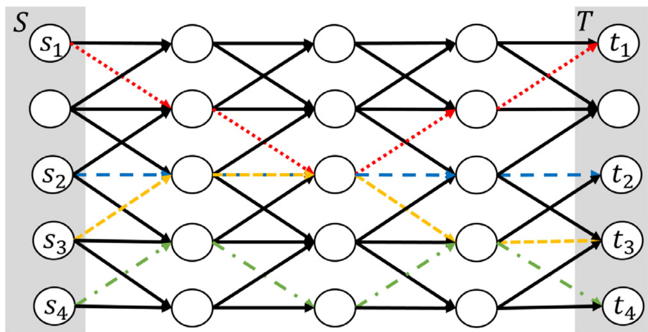

Fig. 6. Optimal solution to a 4-agent, $(m, n) = (5, 5)$ MSPP-PD(ABP) instance with $(f(\boldsymbol{x}^*), p^*) = (16, 1)$ for $(w_d, w_p) = (0.5, 0.5)$.

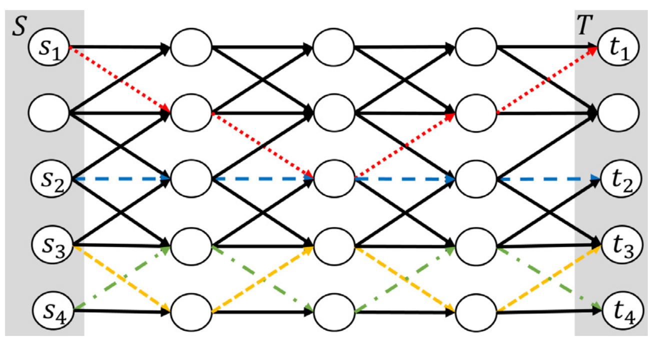

Fig. 7. Optimal solution to a 4-agent, $(m, n) = (5, 5)$ MSPP-PD(NQP) instance with $(f(\boldsymbol{x}^*), p^*) = (17, 1)$ for $w_p \in [0.21, 0.54]$ and $w_d = 1 - w_p$.

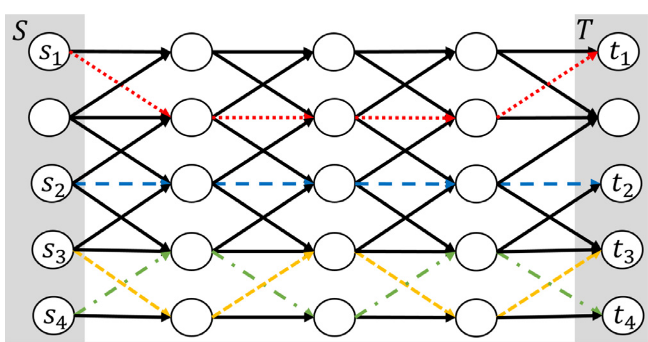

Fig. 8. Optimal solution to a 4-agent, $(m, n) = (5, 5)$ MSPP-PD(NQP) instance with $(f(\boldsymbol{x}^*), p^*) = (18.2, 0)$ for $w_p \in [0.55, 0.99]$ and $w_d = 1 - w_p$.

non-dominated solutions on the Pareto frontier. Such a set of solutions can be presented to a decision-maker to elicit relative objective function priorities *a posteriori* via the identification of a preferred solution. This process can be demonstrated for the MSPP-PD(NQP) model variant, yielding three non-dominated solutions. The three solutions are conveyed collectively by Table 1 and Figs. 5, 7, and 8.

Regarding the set of Pareto optimal solutions displayed in Table 1, the commercial solver returned the solution in Fig. 6 for $w_p = [0.01, 0.20]$, but the solution displayed in Fig. 5 is an alternative optimal solution. For an analyst, this implies that utilizing different penalty frameworks (i.e., binary, linear, or quadratic) also has merit for identifying alternatives for a decision maker.

3.3. Examination of network congestion on optimal solutions for MSPP-PD model variants

Increasing the number of agents being routed over a given network will increase the potential for conflict between agents' paths, as well as penalties incurred, if the network is sufficiently congested; dispersed agents will not face conflict in the same manner as a condensed set. However, it is not obvious how increasing the number of agents traversing the network (i.e., network congestion) affects the optimal solutions to the MSPP-PD variants. Examining the optimal solutions will identify the incurred cost, in terms of total agent distance, to reduce conflict among agents' paths.

To examine the impact different penalty metric variants of the MSPP-PD have on the optimal agent distance travelled for varying levels of network congestion, a total of 1000 MSPP instances were generated to compare the optimal total distances travelled for analysis. For each instance, we utilize a network having the same topology as the networks in Figs. 5–8, but with $(m, n) = (6, 6)$ and arc lengths generated stochastically according to a uniform distribution (i.e., $d_{ij} = U[0, 2]$).[1]

Additionally, varying levels of network congestion are considered for each of the 1000 instances. We consider network congestion via the number of agents traversing a given network for a fixed network topology. For these experiments' network topology, which manifests in Figs. 5–8, the number of agents relative to the number of rows is important to the analysis. The selected ratios are 0.5, 1, 1.5, and 2 (i.e., 3, 6, 9, and 12 agents for a network having $m = 6$). We assumed the agents' sources and termini are ordinally sorted from top-to-bottom among the respective sets S and T, meaning that agents traversing the network directly from left-to-right (i.e., without regard to minimizing total distance travelled) would incur no penalties $\mathcal{K} \leq m$. As the ratio of agents to rows increases, the agents' sources and termini are added alternatively to the odd and even rows. For example, in the 9-agent case, each odd row features two agents' sources and termini while each even row possesses one.

A comparison of the mean optimal total distance travelled for the optimal solutions of the 1000 randomly generated instances of MSPP and MSPP-PD model variants with $(w_d, w_p) = (0.5, 0.5)$ and the selected network congestion levels yields interesting insights that can be seen in Fig. 9. From left-to-right along the horizontal axis within Fig. 9, the three-letter abbreviations correspond to the penalty metric for the MSPP-PD variant (e.g., ABP for arc binary penalty, NLP for node linear penalty). For each given network congestion level, a color-coded bar displays the mean optimal total distance travelled over model variants. Intuitively, higher network congestion levels yield higher total distances travelled among optimal solutions.

To compare any two models' solutions for a given network congestion level, the relative mean difference of total distance travelled over the 1000 instances can identify which model is creating longer paths to avoid path conflict penalties. Illustrated by Fig. 9, we find that the relationship between the model variants over different levels of network congestion is dynamic. As the network congestion varies, the degree to which each of the models' greatest mean optimal total distance travelled differs relative to the MSPP solution changes. For instance, the greatest mean difference in optimal total distance travelled in the 6-agent case is the MSPP-PD(NLP), but it is MSPP-PD(ALP) for the 9-agent scenario. Addi-

[1] The instances can be found at https://github.com/mike6hughes/Multiple-Shortest-Path-Problem-with-Path-Deconfliction

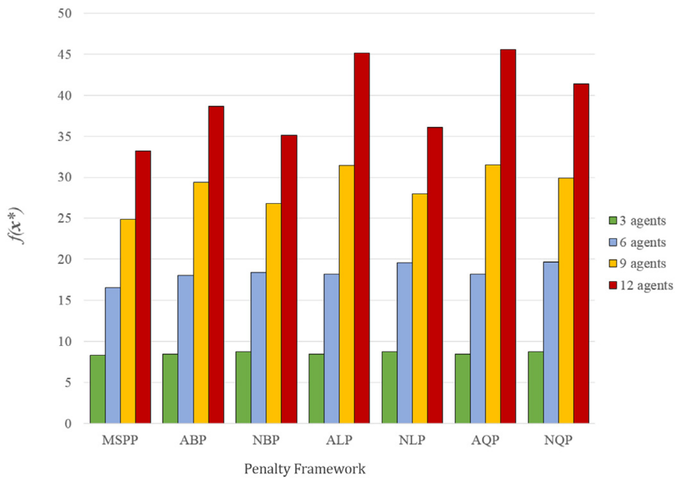

Fig. 9. Mean optimal total distance travelled for combinations of MSPP or MSPP-PD model variants and network congestion level.

Table 2
Average agent distance travelled for the optimal solutions of the MSPP or MSPP-PD model variants and network congestion levels.

	MSPP or MSPP-PD Model Variant						
No. of Agents	MSPP	ABP	NBP	ALP	NLP	AQP	NQP
3	2.76	2.83	2.92	2.83	2.92	2.83	2.92
6	2.76	3.01	3.06	3.03	3.27	3.04	3.28
9	2.76	3.26	2.98	3.50	3.11	3.50	3.32
12	2.76	3.22	2.92	3.76	3.01	3.80	3.45

tionally, the increasing differences in means that coincide with increased network congestion indicate that certain MSPP-PD variants may inadequately incentivize path deconfliction for highly congested networks; a trend that can be examined via the average agent distance travelled.

Table 2 presents the average agent distance travelled over the 1000 optimally solved instances for each model variant and network congestion level. The results indicate that, when conflict penalties become unavoidable, the objective of minimizing distance dominates the optimal solution for certain models as distance-minimizing paths are leveraged to overcome path conflict penalties that are not weighted heavily in comparison. For instance, the MSPP-PD(ABP), MSPP-PD(NBP), and MSPP-PD(NLP) model variants' average agent path length decreases after a certain level of network congestion with the addition of more agents. In contrast, the average agent distance travelled for the other three MSPP-PD model variants strictly increase with increasing levels of congestion, implying these model variants' penalty metrics still impel path deconfliction for equal objective weights. The potentially inadequate path deconfliction of the MSPP-PD(ABP), MSPP-PD(NBP), and MSPP-PD(NLP) may be overcome by a thorough exploration of the Pareto frontier or appropriate scaling of the two objectives (e.g., via preemptive objective function weighting (Sherali & Soyster, 1983)).

To further analyze the dynamic relationships between model variants, Tables 3 and 4 report 95% confidence intervals for the difference between optimal total distances travelled for each pairwise combination of the MSPP and MSPP-PD model variants, with 6- and 9-agents. Considering the MSPP or MSPP-PD model variant represent in each row (r) and column (c) of Table 3, for a given instance denote $D_{rc} = f(\boldsymbol{x}_r^*) - f(\boldsymbol{x}_c^*)$ as the difference in total distance travelled by agents in the respective optimal solutions for the model variants. Considering the mean, $\bar{D}_{rc}$, and sample standard deviation, s_{rc}, of this measure over the $n = 1000$ instances tested, Table 3 reports the confidence intervals at the $\alpha = 0.05$ level using a t-distribution (i.e., $\bar{D}_{rc} \pm t_{0.975,999}\frac{s}{\sqrt{n}}$). Each of the confidence intervals were examined using both the Kolmogorov–Smirnoff and Anderson-Darling tests for normality at the $\alpha = 0.05$ significance level; there was insufficient evidence to reject a normality assumption for any of the 21 distributions.

The intervals seen in Tables 3 and 4 can be either less than zero, greater than zero, or contain zero. A confidence interval less than zero indicates that the column model produces a set of agent paths that is longer due to path deconfliction. For instance, comparing the MSPP-PD(ALP) to the MSPP-PD(NLP) when 6-agents traverse the network results in a difference confidence interval of (-1.47, -1.33), suggesting that the MSPP-PD(NLP) is creating longer agent paths due to path deconfliction. With opposite effect, a confidence interval greater than zero indicates that the column model creates sets of agent paths that are shorter. When comparing the MSPP-PD(ALP) to the MSPP-PD(NLP) with 9-agents traversing the network, we find that the confidence interval is (3.41, 3.60). This finding shows that the MSPP-PD(NLP) is prescribing the shorter sets of agent paths, a reversal from the 6-agent scenario model relationship. Finally, in the case where zero is contained within the interval, the two compared model variants are constructing similar sets of agent paths. For the most part, the models are producing distinct solutions. However, between the two scenarios depicted in Tables 3 and 4, only one model variant pair has a confidence interval containing zero. In the 6-agent case, the MSPP-PD(ALP) and the MSPP-PD(AQP) produce similar results. In the 9-agent scenario, the difference remains small (i.e., (-0.08, -0.03)), but zero is not contained within the interval. Cumulatively, the intervals from the two tables further illustrate that, as network congestion increases (i.e., the number of agents increase), the order of models which increase the total distance travelled the most from the MSPP solution changes and the differences in the model variants' optimal paths are dynamic.

3.4. An exploration of the model variants' practability

Finally, it is important to consider the practability of these models for solving larger instances of the MSPP-PD. This testing examines average convergence time of the model variants in two stages: first on the instances examined in Section 3.3 and subsequently on larger instances that better identify the variants' respective limitations. For all testing, the GAMS modeling language is used to call CPLEX (Version 12.8.0.0) commercial solver with an Intel(R) Core(TM) i5-3320m CPU @2.60GHz with 8.00 GB of RAM on 64-bit operating system.

Table 5 reports the time to convergence (sec) quartiles for the 1000 instances of the MSPP and MSPP-PD model variants from the experiment in Section 3.3. These results confirm the intuition that increasing network congestion (i.e., increasing the number of agents) increases the required computational effort to solve the different MSPP-PD model variants. Of note, the increased convergence time may be a result of not only the increase in problem size, but added symmetry within the problem between the homogeneous agents with identical source and termini nodes. Moreover, the results provide insight into which models are the most practable and which models have the potential to become intractable for larger problem instances. The MSPP-PD(ALP) is clearly the most efficient, as the median time to solve the most congested network is about 3 times faster than the next most efficient model–the MSPP-PD(NLP). In contrast, as the size of the problem grows, the MSPP-PD(NQP) has a much slower median solve time, failing to identify an optimal solution for 0.11% of the instances examined within 10 min.

To better frame the practability conversation, a second stage of testing considers different combinations of increasing larger net-

Table 3
95% confidence intervals for the MSPP and MSPP-PD model variant differences in optimal total distance travelled with 6-agents for the 1000 generated instances (row model - column model).

	ABP	NBP	ALP	NLP	AQP	NQP
MSPP	(−1.52, −1.43)	(−1.87, −1.73)	(−1.67, −1.57)	(−3.10, −2.94)	(−1.68, −1.57)	(−3.18, −3.02)
ABP		(−0.40, −0.24)	(−0.18, −0.11)	(−1.62, −1.47)	(−0.18, −0.12)	(−1.70, −1.55)
NBP			(0.09, 0.26)	(−1.32, −1.13)	(0.09, 0.26)	(−1.40, −1.21)
ALP				(−1.47, −1.33)	(−0.01, 0.00)	(−1.55, −1.41)
NLP					(1.33, 1.46)	(−0.10, −0.05)
AQP						(−1.54, −1.41)

Table 4
95% confidence intervals for the MSPP and MSPP-PD model variant differences in optimal total distance travelled with 9-agents for the 1000 generated instances (row model - column model).

	ABP	NBP	ALP	NLP	AQP	NQP
MSPP	(−4.60, −4.39)	(−2.03, −1.87)	(−6.69, −6.49)	(−3.16, −3.00)	(−6.74, −6.54)	(−5.10, −4.88)
ABP		(2.42, 2.67)	(−2.23, −1.96)	(1.29, 1.53)	(−2.28, −2.01)	(−0.65,−0.34)
NBP			(−4.77, −4.51)	(−1.25, −1.02)	(−4.82, −4.56)	(−3.19, −2.89)
ALP				(3.41, 3.60)	(−0.08,−0.03)	(1.50, 1.69)
NLP					(−3.65,−3.47)	(−2.00,−1.82)
AQP						(1.56, 1.74)

Table 5
Time to convergence (sec) quartiles for the 1000 instances of the MSPP and MSPP-PD model variants from the experiment in Section 3.3.

No. of	Value	MSPP or MSPP-PD Model Variant						
Agents		MSPP	ABP	NBP	ALP	NLP	AQP	NQP
3	**Min**	0.015	0.015	0.015	0.015	0.015	0.046	0.031
	Q_1	0.015	0.016	0.016	0.015	0.031	0.078	0.062
	Q_2	0.016	0.016	0.031	0.016	0.031	0.078	0.062
	Q_3	0.016	0.031	0.031	0.031	0.032	0.078	0.063
	Max	0.063	0.250	0.078	0.062	0.078	0.125	0.203
6	**Min**	0.015	0.015	0.015	0.015	0.015	0.062	0.062
	Q_1	0.015	0.016	0.062	0.016	0.046	0.093	0.094
	Q_2	0.016	0.016	0.078	0.031	0.062	0.094	0.125
	Q_3	0.016	0.031	0.141	0.031	0.078	0.109	0.156
	Max	0.047	0.078	0.593	0.078	9.064	9.282	8.782
9	**Min**	0.015	0.015	0.031	0.015	0.031	0.110	0.265
	Q_1	0.015	0.063	0.109	0.031	0.078	0.25	1.389
	Q_2	0.016	0.109	0.187	0.031	0.094	0.328	1.685
	Q_3	0.016	0.156	0.327	0.031	0.124	0.436	2.117
	Max	0.031	0.453	0.873	0.109	0.468	1.107	11.294
12	**Min**	0.015	0.015	0.031	0.015	0.031	0.515	5.71
	Q_1	0.015	0.266	0.124	0.031	0.078	3.03	43.341
	Q_2	0.016	0.53	0.218	0.031	0.093	5.569	72.369
	Q_3	0.016	1.108	0.39	0.046	0.109	8.60725	128.9815
	Max	0.032	2.668	1.45	0.078	0.374	34.274	600.058*

* Optimal solution not identified for 11 instances within the 10 min time limit.

work sizes and network congestion. For this experiment, we utilized the same network topology of Section 3.3 with $m = n$, considering networks of sizes $m = 6, 8, 10, 12$. The number of agents was determined by the previously discussed network congestion ratios (i.e., 0.5, 1, 1.5, and 2), wherein an instance with a network congestion ratio of 1 when $m = 6$ would feature 6 agents. For this exploration, a smaller sample of ten instances of each network size-and-congestion-level was examined for all model variants[2]; the resulting average convergence times are reported in Table 6, given a maximum run time of 10 min (i.e., 600 s).

Table 6 illustrates how both network size and congestion ratio combine to increase the time to convergence for all MSPP-PD model variants. However, the different models' are affected to varying degrees. For example, the MSPP-PD(ALP) and MSPP-PD(NBP) are the only formulations that were able to converge for all instances. In comparison, the required computational effort for other model variants increases more rapidly relative to MSPP-PD(ALP), they are ill-suited to efficiently explore a Pareto frontier of routing scenarios featuring large networks and/or higher levels of network congestion. Additionally, we see that the arc-specific penalty metrics converge faster than their node counterparts with the exception of the binary frameworks, wherein the node-based framework enables quicker convergence.

The computational results depicted in Tables 5 and 6 both suggest that the MSPP-PD(ALP) is the fastest of the proposed model variants. Therefore, we recommend a decision-maker adopt the MSPP-PD(ALP) model variant to deconflict the routing of agents over a network, unless a specific application motivates the adoption of a different penalty metric. In such a situation, however, we propose that any conceptual preference for a node penalty framework can be overcome by modifying the problem via node splitting (Ahuja, Magnanti, & Orlin, 1993) and penalizing only the arcs corresponding to the travel between the newly bifurcated pairs of nodes, enabling the use of a corresponding arc-based penalty model.

[2] The instances can be found at https://github.com/mike6hughes/Multiple-Shortest-Path-Problem-with-Path-Deconfliction .

Table 6
MSPP-PD model variants' average convergence time (sec) over varying network size and congestion levels.

Network Size	Network Congestion Ratio				Network Size	Network Congestion Ratio			
ABP	0.5	1	1.5	2	NBP	0.5	1	1.5	2
6	0.0203	0.0328	0.1075	0.9252	6	0.0281	0.0904	0.2434	0.3402
8	0.0282	0.1092	3.0047	6.3055	8	0.05	1.036	2.6814	2.8613
10	0.0561	1.203	9.4537	65.1336	10	0.1218	7.1369	22.1254	21.3053
12	0.0906	7.7969	124.0489	306.5965**	12	0.2061	47.6631	62.7702	77.5714
ALP	0.5	1	1.5	2	NLP	0.5	1	1.5	2
6	0.0171	0.0249	0.0279	0.0609	6	0.0281	0.0516	0.0967	0.1046
8	0.0328	0.0422	0.0763	0.1232	8	0.05	0.237	0.6973	0.7193
10	0.1044	0.0828	0.1871	0.4369	10	0.1574	3.2619	32.651	71.9806
12	0.1028	0.1621	0.485	1.0656	12	0.2418	92.4243	590.5043***	581.3407†
AQP	0.5	1	1.5	2	NQP	0.5	1	1.5	2
6	0.0732	0.0967	0.3321	5.7439	6	0.0593	0.1436	1.5288	73.3018
8	0.2091	0.3072	87.8595	600.153‡	8	0.05	1.3713	592.9643†	600.0938‡
10	0.5881	1.3259	600.4668‡	600.3807‡	10	0.1574	177.548*	600.1855‡	600.1749‡
12	1.2604	23.3364	600.9142‡	600.7614‡	12	0.2418	600.3965‡	600.3277‡	600.3387‡

Number of instances that did not converge: * = 1, ** = 5, *** = 8, † = 9, ‡ = 10

4. Conclusions and recommendations

Many applications require the routing of multiple agents over a network, wherein the spatial deconfliction of their paths is also of interest. This research formulated the Multiple Shortest Path Problem with Path Deconfliction (MSPP-PD), as well as several variants of metrics to penalize conflicting agent paths according to the location, type of interactions, and degree to which path conflict occurs. Anecdotal testing demonstrated the potential for the MSPP-PD variants to prescribe different agent routing solutions, and empirical testing illustrated the effect of the different penalty metrics on both the total distance travelled by agents and the required computational effort to identify the corresponding solutions.

For a practicioner, we recommend penalizing path conflict when agents traverse a common arc (in lieu of a common node) and the adoption of penalty metrics based on the *number of agents* involved in a path conflict (i.e., linear). The computational testing results in Section 3.4 portend practical model tractability for larger and more congested instances. Such rapid identification of optimal solutions will enable exploratory analyses via tradeoffs in objectives on a Pareto frontier of efficient solutions.

A recommended extension to this research is the considering of symmetry defeat mechanisms to potentially improve convergence rates if larger instances must be solved. An intended sequel to this research will examine modify the spatial deconfliction framework for examining the paths of agents to a spatio-temporal framework that considers interagent distances at any time. Such a framework will merit an examination of limits to interagent communication, alternative solution methodologies (e.g., online optimization) and, ultimately, consideration of adversarial actions to impair the desired outcomes for agent routing.

Disclaimer

The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government.

Acknowledgements

The authors thank the editor and three reviewers for their detailed and constructive comments that improved both the content and presentation of this paper.

References

Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network flows: Theory, algorithms and applications*. Upper Saddle River, NJ: Prentice Hall.
Anderluh, A., Nolz, P. C., Hemmelmayr, V. C., & Crainic, T. G. (2019). Multi-objective optimization of a two-echelon vehicle routing problem with vehicle synchronization and 'grey zone' customers arising in urban logistics. *European Journal of Operational Research*. https://doi.org/10.1016/j.ejor.2019.07.049.
Androutsopoulos, K. N., & Zografos, K. G. (2008). Solving the *k*-shortest path problem with time windows in a time varying network. *Operations Research Letters, 36*(6), 692–695.
Bagchi, T. P., Gupta, J. N., & Sriskandarajah, C. (2006). A review of TSP based approaches for flowshop scheduling. *European Journal of Operational Research, 169*(3), 816–854.
Bellman, R., & Kalaba, R. (1960). On *k*th best policies. *Journal of the Society for Industrial and Applied Mathematics, 8*(4), 582–588.
Bertsimas, D., & Dunn, J. (2019). *Machine learning under a modern optimization lens* (1st ed.). Charlestown, MA: Dynamic Ideas LLC.
Bock, F., Kantner, H., & Haynes, J. (1958). *An algorithm (the r – th best path algorithm) for finding and ranking paths through a network*. Armour Research Foundation Technology Center.
Calvo, R. W., & Cordone, R. (2003). A heuristic approach to the overnight security service problem. *Computers & Operations Research, 30*(9), 1269–1287.
Cidon, I., Rom, R., & Shavitt, Y. (1997). Multi-path routing combined with resource reservation. In *Proceedings of infocom'97: 1* (pp. 92–100). IEEE.
Clímaco, J. C., & Pascoal, M. M. (2009). Finding non-dominated bicriteria shortest pairs of disjoint simple paths. *Computers & Operations Research, 36*(11), 2892–2898.
Dantzig, G. B. (1960). On the shortest route through a network. *Management Science, 6*(2), 187–190.
Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science, 6*(1), 80–91.
De Kort, J. B. (1993). A branch and bound algorithm for symmetric 2-peripatetic salesman problems. *European Journal of Operational Research, 70*(2), 229–243.
Department of the Air Force (2019). United states air force fiscal year 2019 budget overview. *Office of the Secretary of the Air Force*, 20.
Dial, R. B. (1979). A model and algorithm for multicriteria route-mode choice. *Transportation Research Part B: Methodological, 13*(4), 311–316.
Dijkstra, E. W., et al. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik, 1*(1), 269–271.
Disser, Y., Müller-Hannemann, M., & Schnee, M. (2008). Multi-criteria shortest paths in time-dependent train networks. In *International workshop on experimental and efficient algorithms* (pp. 347–361). Springer.
Dreyfus, S. E. (1969). An appraisal of some shortest-path algorithms. *Operations research, 17*(3), 395–412.
Ehrgott, M. (2005). *Multicriteria optimization*: 491 (2nd ed.). Berlin-Heidelberg, Germany: Springer Science & Business Media.
Flood, M. (1956). The traveling-salesman problem. *Operations Research, 4*(1), 61–75.
García-Martínez, C., Cordón, O., & Herrera, F. (2007). A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. *European Journal of Operational Research, 180*(1), 116–148.
Hoffnam, W., & Pavley, R. (1959). A method for the solution of the *N*th best problem. *Journal of Association for Computer Machinery, 6*(4), 506–514.
Israeli, E., & Wood, R. K. (2002). Shortest-path network interdiction. *Networks: An International Journal, 40*(2), 97–111.
Jahn, O., Möhring, R. H., Schulz, A. S., & Stier-Moses, N. E. (2005). System-optimal routing of traffic flows with user constraints in networks with congestion. *Operations Research, 53*(4), 600–616.
Kara, I., Laporte, G., & Bektas, T. (2004). A note on the lifted Miller–Tucker–Zem-

lin subtour elimination constraints for the capacitated vehicle routing problem. *European Journal of Operational Research, 158*(3), 793–795.
Kerbache, L., & Smith, J. M. (2000). Multi-objective routing within large scale facilities using open finite queueing networks. *European Journal of Operational Research, 121*(1), 105–123.
Krarup, J. (1975). The peripatetic salesman and some related unsolved problems. In *Combinatorial programming, methods and applications* (pp. 173–178). Dordrecht, Holland: D. Rediel.
Kulkarni, R., & Bhave, P. R. (1985). Integer programming formulations of vehicle routing problems. *European Journal of Operational Research, 20*(1), 58–67.
Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research, 59*(3), 345–358.
Lehuédé, F., Péton, O., & Tricoire, F. (2020). A lexicographic minimax approach to the vehicle routing problem with route balancing. *European Journal of Operational Research, 282*(1), 129–147.
Lunday, B. J., & Sherali, H. D. (2012). Minimizing the maximum network flow: Models and algorithms with resource synergy considerations. *Journal of the Operational Research Society, 63*(12), 1693–1707.
Lust, T., & Teghem, J. (2010). The multiobjective traveling salesman problem: Asurvey and a new approach. In *Advances in multi-objective nature inspired computing* (pp. 119–141). Springer.
Modesti, P., & Sciomachen, A. (1998). A utility measure for finding multiobjective shortest paths in urban multimodal transportation networks. *European Journal of Operational Research, 111*(3), 495–508.
Mohanta, K., & Poddar, B. (2012). Comprehensive study on computational methods for *k*-shortest paths problem. *International Journal of Computer Applications, 40*(14), 22–26.
Ngueveu, S. U., Prins, C., & Calvo, R. W. (2009). A hybrid tabu search for the m-peripatetic vehicle routing problem. In *Matheuristics* (pp. 253–266). Springer.
Ngueveu, S. U., Prins, C., & Calvo, R. W. (2010). Lower and upper bounds for the m-peripatetic vehicle routing problem. *4OR, 8*(4), 387–406.
Paessens, H. (1988). The savings algorithm for the vehicle routing problem. *European Journal of Operational Research, 34*(3), 336–344.
Papadimitriou, C. H., & Yannakakis, M. (2000). On the approximability of trade-offs and optimal access of web sources. In *Proceedings 41st annual symposium on foundations of computer science* (pp. 86–92). IEEE.
Parker, L. E. (2009). Path planning and motion coordination in multiple mobile robot teams. *Encyclopedia of Complexity and System Science*, 5783–5800.
Pollack, M. (1961). Solutions of the kth best route through a network-a review. *Journal of Mathematical Analysis and Applications, 3*(3), 547–559.
Potvin, J.-Y., & Rousseau, J.-M. (1993). A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. *European Journal of Operational Research, 66*(3), 331–340.
Reinhardt, L. B., & Pisinger, D. (2011). Multi-objective and multi-constrained non-additive shortest path problems. *Computers & Operations Research, 38*(3), 605–616.
Sakarovitch, M. (1968). The *k* shortest shortest chains in a graph. *Transportation Research, 2*(1), 1–11.
Sherali, H. D., & Soyster, A. L. (1983). Preemptive and nonpreemptive multi-objective programming: Relationship and counterexamples. *Journal of Optimization Theory and Applications, 39*(2), 173–186.
Silva, R., & Craveirinha, J. (2004). An overview of routing models for MPLS networks. In *Proc. 1st workshop multicriteria modelling in telecommunication network planning and design, faculty of economics of the university of Coimbra* (pp. 17–24). Coimbra, Portugal
Sörensen, K. (2006). Route stability in vehicle routing decisions: a bi-objective approach using metaheuristics. *Central European Journal of Operations Research, 14*(2), 193–207.
Stodola, P., & Mazal, J. (2017). Model of optimal cooperative reconnaissance and its solution using metaheuristic methods. *Defence Science Journal, 67*(5), 529.
Talarico, L., Sörensen, K., & Springael, J. (2015a). The *k*-dissimilar vehicle routing problem. *European Journal of Operational Research, 244*(1), 129–140.
Talarico, L., Sörensen, K., & Springael, J. (2015b). Metaheuristics for the risk-constrained cash-in-transit vehicle routing problem. *European Journal of Operational Research, 244*(2), 457–470.
Tarapata, Z. (2007). Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms. *International Journal of Applied Mathematics and Computer Science, 17*(2), 269–287.
Theys, C., Bräysy, O., Dullaert, W., & Raa, B. (2010). Using a TSP heuristic for routing order pickers in warehouses. *European Journal of Operational Research, 200*(3), 755–763.
Tian, J., Shen, L., & Zheng, Y. (2006). Genetic algorithm based approach for multi-UAV cooperative reconnaissance mission planning problem. In *International symposium on methodologies for intelligent systems* (pp. 101–110). Springer.
Van Breedam, A. (1995). Improvement heuristics for the vehicle routing problem based on simulated annealing. *European Journal of Operational Research, 86*(3), 480–490.
Vassilvitskii, S., & Yannakakis, M. (2004). Efficiently computing succinct trade-off curves. In *International colloquium on automata, languages, and programming* (pp. 1201–1213). Springer.
Vidal, T., Laporte, G., & Matl, P. (2019). A concise guide to existing and emerging vehicle routing problem variants. *European Journal of Operational Research*.
Wang, Z., Guo, J., Zheng, M., & Wang, Y. (2015). Uncertain multiobjective traveling salesman problem. *European Journal of Operational Research, 241*(2), 478–489.
Wolsey, L. A., & Nemhauser, G. L. (1999). Integer and combinatorial optimization. *Wiley-Interscience Series in Discrete Mathematics and Otimization*: 55. New York, NY: John Wiley & Sons.
Yen, J. Y. (1971). Finding the *k* shortest loopless paths in a network. *Management Science, 17*(11), 712–716.
Yoon, Y. S., & Kim, J. K. (2006). A relationship between bounds on the sum of squares of degrees of a graph. *Journal of Applied Mathematics and Computing, 21*(1–2), 233–238.
Yousefi, A., & Donohue, G. (2004). Temporal and spatial distribution of airspace complexity for air traffic controller workload-based sectorization. In *AIAA 4th aviation technology, integration and operations (ATIO) forum* (p. 6455). Chicago, Illinois
Zhan, F. B., & Noon, C. E. (1998). Shortest path algorithms: an evaluation using real road networks. *Transportation Science, 32*(1), 65–73.
Zografos, K. G., & Androutsopoulos, K. N. (2008). A decision support system for integrated hazardous materials routing and emergency response decisions. *Transportation Research Part C: Emerging Technologies, 16*(6), 684–703.