Robotica (2018) volume 36, pp. 1781–1803. © Cambridge University Press 2018
doi:10.1017/S0263574718000735

A survey on routing problems and robotic systems

Douglas G. Macharet* and Mario F. M. Campos

Computer Vision and Robotics Laboratory, Department of Computer Science, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil. E-mail: mario@dcc.ufmg.br

(Accepted July 18, 2018. First published online: August 6, 2018)

SUMMARY
Planning paths that are length or time optimized or both is an age-long problem for which numerous approaches have been proposed with varied degree of success depending on the imposed constraints. Among classical instances in the literature, the Traveling Salesman Problem and the Vehicle Routing Problem have been widely studied and frequently considered in the realm of mobile robotics. Understandably, the classical formulation for such problems do not take into account many different issues that arise in real-world scenarios, such as motion constraints and dynamic environments, commonly found in actual robotic systems, and consequently the solutions have been generalized in several ways. In this work, we present a broad and comprehensive review of the classical works and recent breakthroughs regarding the routing techniques ordinarily used in robotic systems and provide references to the most fundamental works in the literature.

KEYWORDS: Traveling Salesman Problem; Vehicle Routing Problem; Dubins vehicle; Dubins TSP; TSP with neighborhoods; k-TSP; Dynamic VRP.

1. Introduction

Recent advances in the research domain of autonomous vehicles have raised a broad range of intriguing questions and uncovered subtle nuances of several classes of already known problems. Path planning is fundamental to navigation, one of the basic problems in mobile robotics, which requires answering a simple question: "How do I get there?" The answer to this question is directly related to the strategy used by the mobile robot to safely arrive at a goal position.

Therefore, finding feasible routes for mobile agents that are either length or time optimized is paramount, and has been the goal of several research fields, including robotics. The literature on the subject is vast and includes works such as refs. [19,49].

In this context, the Traveling Salesman Problem (TSP) remains one of the most studied routing problems, and since it is NP-Hard, several heuristic algorithms to tackle it have been proposed. However, for a large number of real-world scenarios, the use of the classical mathematical formulation for the TSP has shown to be either insufficient to grasp the peculiarities or too simplistic to be useful. In order to overcome some of these limitations, the TSP has been generalized to encompass the following requirements:

- *Motion constraints:* Kinematic and dynamic motion constraints, for example, play a fundamental role in the movement of the majority of vehicles currently in use, such as automobiles and fixed-wing aircrafts. These vehicles have one or more constraints associated with their motion, like minimum turning radius, stall speed, maximum pitch (climb/dive) angles, among others.
- *Neighborhoods:* Instead of reaching the exact coordinates of a point of interest, the vehicle is required to visit any location within an area surrounding that point. This model is more suitable for applications such as Wireless Sensors Networks (WSNs), where communication is expected to be available within a certain region where nodes and mobile agents are located.

* Corresponding author. E-mail: doug@dcc.ufmg.br

- *Multiple vehicles:* The application of multiple robots presents several advantages like increased robustness and, in most cases, reduction in the time required to accomplish a given task. However, it also raises many challenges, such as coordination, control and planning.
- *Dynamic scenarios:* Current literature usually considers static instances, i.e., a vehicle is dispatched to follow an unchangeable and previously planned path, which are assumptions that seldom hold due to the dynamic nature of many real-world applications. In WSNs, for example, new nodes (or those awakening from sleep states) should also be visited, demanding changes to precomputed paths.

All of the above generalizations were developed to tackle real-world problems and have effectively been used in many different applications such as precision agriculture, environmental monitoring, surveillance and exploration of unknown regions.

In this work, we intend to present a broad and comprehensive review of the literature, discussing the fundamental generalizations for both the TSP and Vehicle Routing Problem (Vehicle Routing Problem (VRP)) regarding their use in robotic systems. Therefore, we will detail and discuss the state-of-the-art considering the aforementioned generalizations: (i) motion constraints; (ii) inexact visit position (neighborhoods); (iii) multiple vehicles; and (iv) dynamic scenarios.

2. Motion Planning

Motion planning is a fundamental task in robotics and especially for autonomous mobile robots. Therefore, such problem has been broadly studied and a wide variety of different techniques can be found in the literature.[19,49,88]

Consider a world $\mathcal{W}$, where either $\mathcal{W} = \mathbb{R}^2$ or $\mathcal{W} = \mathbb{R}^3$, which contains a region with obstacles $\mathcal{O} \subset \mathcal{W}$. The set of configurations that a robot may achieve in its environment constitutes a space that can be defined as $\mathcal{C}_{\text{free}} = \mathcal{C} \setminus \mathcal{C}_{\text{obs}}$, where $\mathcal{C}$ is the set of all possible configurations and $\mathcal{C}_{\text{obs}}$ configurations that collide with an obstacle. The basic motion planning problem can be formally defined as finding a path from $\mathbf{q}_I \in \mathcal{C}_{\text{free}}$ to $\mathbf{q}_G \in \mathcal{C}_{\text{free}}$, where $\mathbf{q}_i = \langle \mathbf{p}_i, \psi_i \rangle$ represents the *Initial* and *Goal* configurations, respectively. A *complete* algorithm must compute a (continuous) path, $\tau : [0, 1] \to \mathcal{C}_{\text{free}}$, such that $\tau(0) = \mathbf{q}_I$ and $\tau(1) = \mathbf{q}_G$.[49]

Several reactive navigation strategies can be found in the literature, such as Artificial Potential Fields,[45] Vector Field Histogram[16] and the Dynamic Window Approach.[33] These techniques do not rely on any previous information about the environment and consider only the information provided by its proprioceptive sensors to decide the next action to take. Even though it might be possible for such robots to reactively traverse their environments, the competence of planning and computing paths is an important feature, especially to overcome limitations such as local minima.[47]

Considering the availability of maps, Roadmap approaches such as Visibility Graphs,[56] Cell Decomposition[19,49] and Voronoi Diagrams[93] are the most fundamental in the area. As an alternative to deterministic planning algorithms, random methods generally known as Probabilistic Road Maps (PRMs) or Sampling-based motion algorithms[49] have also been developed. Among several random-sampling motion planning methodologies, Rapidly-Exploring Random Trees (RRTs)[50] which consists of generating trajectory trees that grow quickly through a known environment, have been largely employed.

Another class of reactive algorithms that have been extensively studied in the past years encompasses navigation methods with collision avoidance based on the velocity space,[18,36,91,104] such as the Velocity Obstacles (VO),[32] Reciprocal VO[99,101] and Optimal Reciprocal Collision Avoidance[100] methods. In these methods, the robot uses the velocities of other robots in its neighborhood to calculate the velocity with which it should move in order to prevent collision with other robots.

Nonetheless, there are several problems that require the vehicle to traverse a set of points instead of just reaching a single destination point. Transportation and logistics applications, surveillance and monitoring tasks as well as agricultural activities (e.g. pesticide spraying), among others. In such cases, an efficient and effective visiting route may be obtained considering different characteristics of the vehicle and the environment, resulting in an in an even greater challenge.

3. Motivation

A possible taxonomy for the different approaches concerning routing problems may group the works in the literature taking into account, for example, the number (one or several) or type of vehicles (holonomic or non-holonomic) involved. Environmental characteristics may also be considered, such as the presence or absence of obstacles and if the environment is dynamic or static.

The TSP is a fundamental combinatorial optimization problem and has been widely studied.[4] The problem may be defined to determine the shortest path (sequence of visit) that passes through a set of previously defined points (cities), starting at any given point and returning to the starting point after visiting all points once. More formally, the problem is to determine the shortest Hamiltonian cycle.[82]

Since the TSP is an NP-hard problem, proposed solutions are often heuristic based. This is explained by the fact that due to its complexity, the time required to compute solutions for instances with a few dozen points may become prohibitive especially if the application depends on a result within a short period of time. Consequently, it is possible to choose between the quality and the efficiency of a solution. Among the most well-known and used heuristics algorithms are those of Christofides[20] and Lin-Kernighan.[55]

One of the main shortcomings when using the classical model proposed by the TSP in robotic systems is the fact that this model does not incorporate other information besides the target positions, for example, the restrictions of the vehicles used such as the minimum turning radius among others. For this reason, several studies have suggested generalizations to the TSP hence, incorporating certain restrictions regarding the vehicle or the environment, making it more comparable to real-world scenarios. Some of these possible variations for the TSP that applies to robotic systems are discussed in the following sections.

4. Non-holonomic Vehicle Routing

As mentioned earlier, a large number of vehicles present kineto-dynamic constraints that are classified as non-holonomic.[49,88] These vehicles may have one or more constraints associated with their motion, among which are the minimum turning radius, maximum torsion and maximum pitch (climb/dive) angles. Therefore, when considering these types of vehicles, such as an Ackerman steering vehicle or a fixed-wing Unmanned Aerial Vehicle (UAV), consideration of these restrictions as well as the use of different methods to plan and estimate the cost of a path is paramount. In the context of mobile manipulators, other types of kinematic constraints may arise, which can be tackled, for example, with optimal control approaches,[25,96] by jointly considering the trajectory generation and the control problems.

Here, we mainly consider the kinematic model for a vehicle with a minimum turning radius ρ, moving on a plane, which may be given by

$$\dot{\mathbf{s}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \nu \ \cos(\theta) \\ \nu \ \sin(\theta) \\ \omega \end{bmatrix}, \tag{1}$$

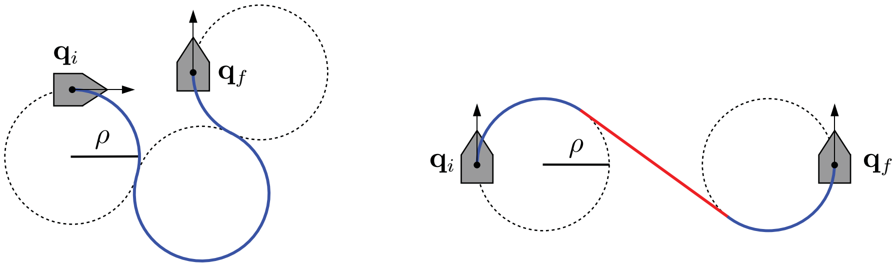
where ν ($\nu \in \mathbb{R}^+$) represents the linear velocity and ω the angular velocity ($\omega \in \{-\nu/\rho,\ 0,\ \nu/\rho\}$).

As far as the reactive approaches are considered, the original VO concept was later extended to handle robots subject to non-holonomic kinematic constraints.[65,104] However, although attainable by the robot, paths planned by on-line navigation strategies may fall short with respect to generating the shortest path.

The generation of paths for non-holonomic vehicles is an important theme in robotics.[49,88] Several studies dealing with this problem make use of the so-called probabilistic algorithms. In ref. [3], a method was proposed to generate trajectories[(1)] for fixed-wing UAVs in environments with obstacles based on the use of the RRT algorithm. In ref. [58], the path planning (time is not considered) is carried out using Genetic Algorithm (GA) and Bézier curves. This type of algorithm (probabilistic) has been used in many studies,[21,42,44,48,66] since its main advantage is the possibility of using simplified representations for all of the constraints involved.

[(1)]A trajectory differs from a path in that for the former time is essential, whereas for the latter it is essentially immaterial.

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press



(a) Short case. (b) Long case.

Fig. 1. Example of Dubins curves with a minimum turning radius ρ for the two basic cases connecting an initial configuration $\mathbf{q}_i$ and a final configuration $\mathbf{q}_f$. (a) Short case. (b) Long case.

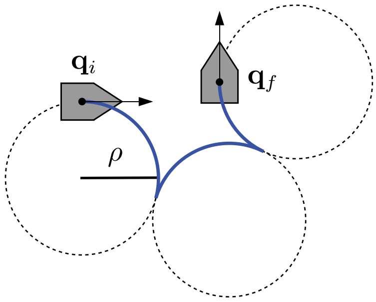


Fig. 2. Example of a Reeds–Shepp curve with a minimum turning radius ρ connecting an initial configuration $\mathbf{q}_i$ and a final configuration $\mathbf{q}_f$.

The aforementioned probabilistic methodologies allow for a feasible path to be generated but so far there has been little concern with a formal investigation of the theoretical lower bounds for the path length, and thus they favor feasibility over optimality.

Taking just the curvature constraint into account, most planners focusing on generating minimum length paths for this type of vehicle make use of Dubins curves for modeling the routes, instead of only straight lines. In its seminal work,[26] Dubins showed a way to compute the shortest path between two points with assigned orientations in the two-dimensional space and considering a vehicle with minimum turning radius constraint. The solution is a composition of curves (C) to the left (L) or right (R) with the minimum turning radius, and straight lines (S). There are basically two types of paths: the short case which is a composition of three arcs (CCC), and the long case which includes a straight line between arcs (CSC), or a sub-path of a path of either one of these two types. Figure 1 shows two examples of Dubins curves connecting different configurations.

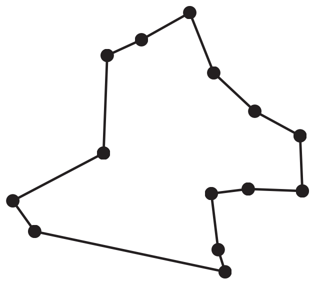
Dubins curves have basically six possible ways to connect straight line segments and arcs: LSL, RSR, RSL, LSR, RLR and LRL. A classification addressing these different possibilities is presented in ref. [87], making it possible to find the optimal path by means of a logic manipulation of the candidate curves without the need to compute all six possible options.

The Dubins vehicle model is often used in robotic systems since it encompasses a large class of non-holonomic vehicles that range from Ackerman steering cars to fixed-wing airplanes. The model initially considered forward-only motion; however, this was extended by Reeds and Shepp,[81] where backward motion is also allowed. Figure 2 illustrates a Reeds–Shepp path.

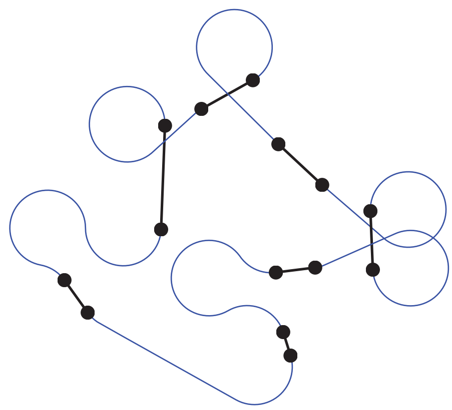
The problem of generating minimum length circuits for visiting a set of points while respecting the curvature constraint of the vehicle and making use of Dubins curves was initially introduced in ref. [85], and was called Dubins TSP (DTSP), which is a generalization of the TSP in which a path is composed of Dubins curves. This problem has two basic requirements: (i) the path connecting any two points should be a Dubins curve; (ii) the Dubins curves that converge on the same point must have the same orientation.

Given $\mathcal{Q}_\Sigma = \langle \mathcal{P}_\Sigma, \Psi_\Sigma \rangle$ some permutation Σ of configurations $\mathbf{q}_i = \langle \mathbf{p}_i, \psi_i \rangle$ to represent the complete state of the vehicle in the SE(2) domain. Having $\mathcal{P}_\Sigma$ as the constant position vector and Ψ_Σ

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press



(a) ETSP solution.



(b) DTSP solution.

Fig. 3. Comparison of the resulting circuits obtained for the (a) ETSP and (b) DTSP with orientations determined by the Alternating Algorithm [85].

the orientation vector of this permutation. The path planning problem can be formally stated as

$$\underset{\Psi,\ \Sigma}{\text{minimize}}\, \mathcal{L}_\rho(\mathcal{Q}_\Sigma) \tag{2}$$

with

$$\mathcal{L}_\rho(\mathcal{Q}_\Sigma) = \mathcal{D}_\rho(\mathbf{q}_N, \mathbf{q}_1) + \sum_{i=1}^{N-1} \mathcal{D}_\rho(\mathbf{q}_i, \mathbf{q}_{i+1}), \tag{3}$$

where $\mathcal{D}_\rho : \mathrm{SE}(2) \times \mathrm{SE}(2) \to \mathbb{R}^+$ represents the length of the shortest feasible path attainable by a Dubins vehicle with a minimum turning radius ρ.

In ref. [85], the path is calculated in three steps. In the first step, the visiting sequence is obtained by computing the solution for one instance of the Euclidean TSP (considering the set of points). The second step is the generation and calculation of the Dubins curves connecting the points. However, in order to generate the curves, it is first necessary to assign an orientation to each of the points—noting that the calculation of these orientations is a challenge on its own. Then, the third step, a simple heuristic is used to generate the orientations called Alternating Algorithm (AA), where the points are alternately connected by line segments and Dubins curves. Formally, the orientation of a waypoint i, expressed as ψ_i, is determined as follows:

$$\psi_i = \begin{cases} \mathrm{dir}(\mathbf{p}_i, \mathbf{p}_{i+1}) & \text{if } i \text{ is odd} \\ \mathrm{dir}(\mathbf{p}_{i-1}, \mathbf{p}_i) & \text{if } i \text{ is even} \end{cases} \quad 1 \le i \le N, \tag{4}$$

where $\mathbf{p}_i = (x_i, y_i)$ (with $1 \le i \le N$) and $\mathrm{dir}(\mathbf{p}_i, \mathbf{p}_j)$ is defined as the direction (unit vector) from a given waypoint toward another, i.e.

$$\mathrm{dir}(\mathbf{p}_i, \mathbf{p}_j) \triangleq \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|}. \tag{5}$$

Figure 3 shows the network of connected points generated using only the Euclidean metric (Fig. 3a) and considering a vehicle with curvature constraints and using Dubins curves (Fig. 3b). As it can be seen, the curvature constraint and the distribution of points may significantly impact the length of the final path.

An alternative method to assign orientations to each of the points is presented in ref. [64], and is referred to as the Mean Angle Algorithm. The algorithm prioritizes the connection of adjacent points whose distances are less than 2ρ (i.e. the points will have the same orientation), with straight lines, and the orientation of the remaining points are set to the mean angle formed by the line segments

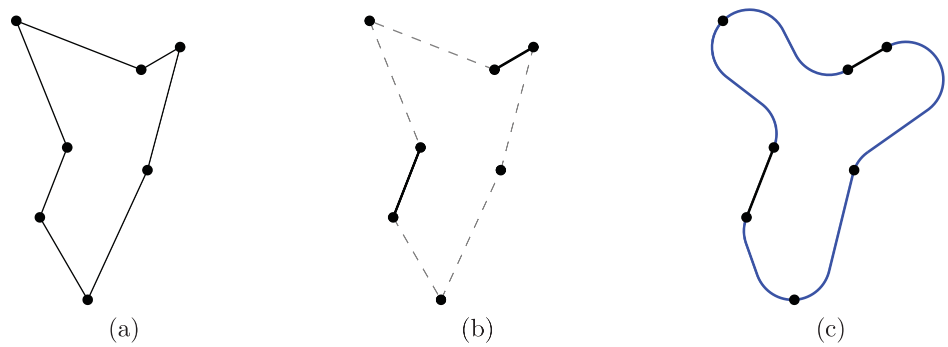


Fig. 4. Steps of the Mean Angle Algorithm [64]. (a) Input waypoints, showing the circuit obtained by the ETSP without non-holonomic constraints; (b) straight segments linking adjacent waypoints whose distances are less than 2ρ; (c) remaining segments filled with Dubins curves considering the mean angle of the previous and following waypoints.

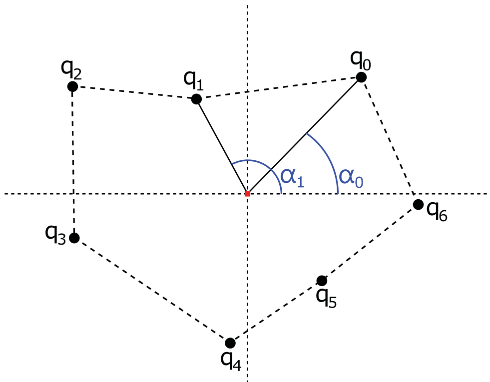


Fig. 5. Illustration of the determination of the visiting sequence according to the angle between the points and the geometric center (red square) of all points [94].

connecting the current point (vertex of the angle) and the previous and the following neighboring waypoints. Figure 4 presents the steps of the Mean Angle Algorithm.

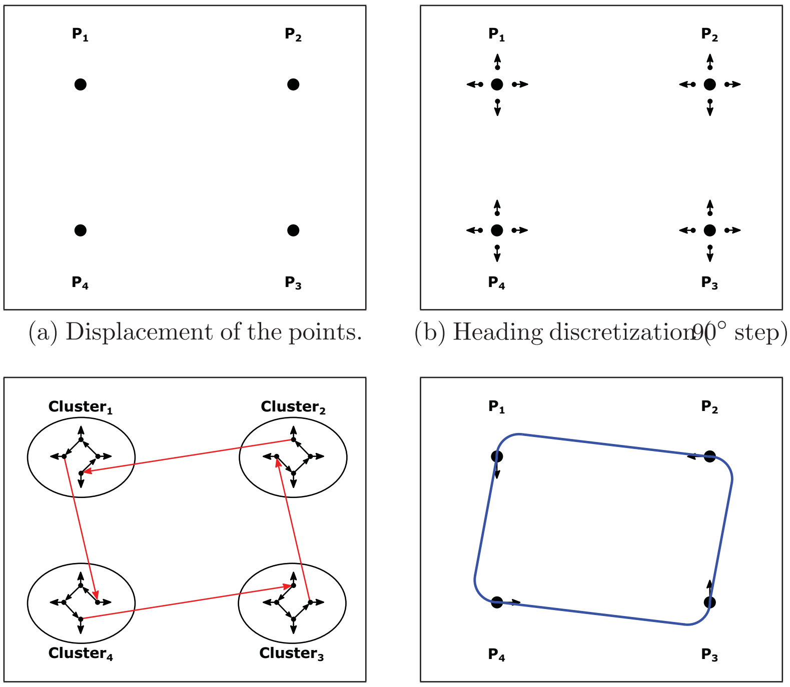
The main differences between the methods dealing with the DTSP are the determination of the sequence of the visit, and the calculation of the orientations associated with the points. We use the term *coupled* to identify the approaches that determine both sequence and orientations in a combined manner, and *decoupled* the methods that determine these variables separately.

Most studies in the literature follow the same decoupled steps for the generation of routes,[57,64,85] i.e. they initially use the visiting sequence obtained from the Euclidean TSP (ETSP), then determine the orientations and finally they connect the points with Dubins curves. However, considering only the Euclidean distance metric to determine the order of the visit does not necessarily guarantee good results when connecting the points using Dubins curves. Actually, this may lead the vehicle to perform many maneuvers (especially in cases where the points are closely located to each other). It was proven in ref. [53] that following a tour based on the ETSP ordering cannot achieve an approximation ratio better than $\Omega(n)$.

Ref. [94] proposed an alternative approach which is not based on the visit sequence obtained by the ETSP. The technique initially calculates the geometric center of all points. Next, the angle between each point and the geometric center is calculated. Finally, the sequence of the visit is determined by considering the ascending order of the points according to the previously obtained angles (Fig. 5).

A similar approach, which considered angular deviations, was presented in ref. [68], where the sequence of visits is obtained by determining the Euclidean circuit with the smallest overall angular cost. The determination of the circuit following this metric called Angle-TSP[1] may be also considered as a generalization of the TSP. The angular cost of a circuit is the sum of all changes in direction (curves) that the vehicle must perform during its navigation. The main idea is that circuits having few corners, i.e. those that are more linear and smooth, tend to produce shorter Dubins curves.

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press



(a) Displacement of the points. (b) Heading discretization (90° step).

(c) Example solution to the GATSP. (d) Final path.

Fig. 6. Illustration of a solution to the DTSP with a discrete optimization approach. A finite set of predefined headings are assigned to each point, next, this problem is mapped into an instance of the GATSP. (a) Displacement of the points. (b) Heading discretization (90° step). (c) Example solution to the GATSP. (d) Final path.

The DTSP has also been addressed using discrete optimization approaches. In ref. [52], the sequence is obtained by directly using the length of the Dubins curves between the points. The orientations of all points are initially set to zero (or to a fixed random value) and all $n(n-1)$ curves interconnecting all points are calculated and connected, obtaining a complete graph. Then, a solution of an instance of the Asymmetric TSP (ATSP) is calculated and used to determine the shortest path in that graph.

The previous approach was later extended to consider a complete heading discretization.[51] The technique functions by choosing *a priori* finite set of K possible headings at each waypoint. Next, a graph is formed considering a collection of N clusters, each cluster containing K nodes corresponding to the headings. Finally, a tour through all clusters containing exactly one point in each cluster is determined. This cluster visiting problem is known as the Generalized ATSP (GATSP) and can be reduced to a classical ATSP over NK nodes using the Noon and Bean transformation.[72]

Figure 6 illustrates the process of addressing the DTSP with a discrete optimization technique. The initial set of points (Fig. 6a) is represented as a collection of points with different headings (Fig. 6b). The problem can then be reduced to an instance of the ATSP (Fig. 6c). Finally, by considering the determined headings, the Dubins curves are calculated (Fig. 6d).

The solution based on heading discretization yields very good results as the steps of the resolution increases, albeit with significant impact on the computational time induced by the number of points and resolution (number of steps) of discretization.

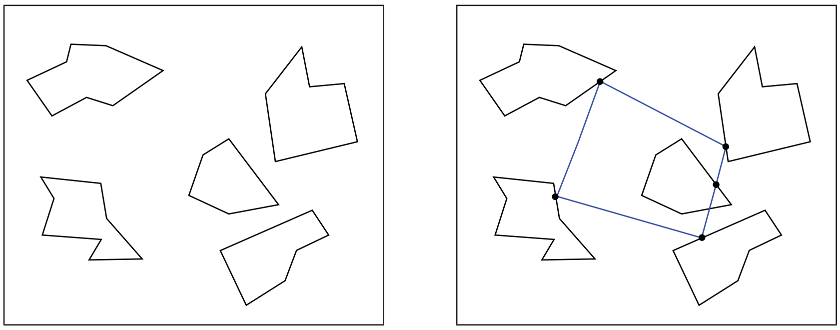
More recently, two planning algorithms related to the DTSP[41] were presented. The first one models as a minimum-time control problem, while the second one is an adaptation of the 2-Opt heuristic for the classical TSP. In this case, the combinatorial (sequence of visit) and motion planning aspects of the DTSP are also tackled in a combined manner. The use of GAs has also been verified in this context.[106]

As shown in ref. [54], the DTSP is also NP-hard. Importantly, unlike other generalizations to the TSP, it is not possible to cast the DTSP as a problem in a finite graph without losing the quality of the result. Therefore, this fact prevents the application of well-established techniques akin to the area of combinatorial optimization.

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

Table I. Comparison of the main works regarding non-holonomic vehicle routing.

Reference	Methodology	Approach
Ref. [51]	Exact	Coupled
Ref. [106]	Evolutionary	Coupled
Ref. [85]	Heuristic	Decoupled
Ref. [94]	Heuristic	Decoupled
Ref. [64]	Heuristic	Decoupled



(a) Input: set of regions to be visited. (b) Output: minimized visit path.

Fig. 7. Example of an instance of the TSPN and its solution (blue line). (a) Input: set of regions to be visited. (b) Output: minimized visit path.

Table I presents a comparison of the main works regarding non-holonomic vehicle routing, highlighting the specific features addressed by each one of them.

5. Vehicle Routing for Visiting Regions

Another generalization to the TSP is known as TSP with Neighborhoods (TSPN) which was initially presented in ref. [5]. In that scenario, the salesman still has a tour to execute, however, in this formulation of the problem, a neighborhood is associated with each point. Therefore, both the salesman and the prospective buyer to be visited may meet anywhere within this neighborhood. The problem can then be defined as how to generate the shortest path that intersects every neighborhood at least once. Figure 7 is an illustration of this problem.

The TSPN is an NP-Hard problem, specifically APX-hard (accepts a Polynomial-Time Approximation Algorithm with approximation ratio limited by a constant), i.e. cannot be approximated by a factor $2 - \varepsilon$, with $\varepsilon > 0$, unless P = NP.[83]

Among the first papers that studied the general case for this problem are refs. [38, 67]. In these works, the authors presented algorithms with an approximation factor of $O(\log n)$, where n is the number of regions.

In ref. [27], an algorithm for the TSPN is presented where the regions are uniform unity disks and initially without intersection. For the general case, where the regions may intersect with each other, initially, it is calculated as a set composed of the maximum number of regions that do not intersect (called independent set). Then, a circuit that moves toward the centers of the regions of this new set is built, however passing through the perimeter of the disks that are not part of the set. The approximation ratio of the algorithm is 11.15 for unit disks. Figure 8 illustrates the tour given the maximal independent set.

A Polynomial-Time Approximation Scheme to the problem is presented in refs. [27,69], specifically for the case of convex regions which have no intersection. The solution presented in ref. [24] considers these same restrictions, however, the work has dealt with the case where the regions have a varying size (non-uniform).

The algorithm presented in ref. [28] considers the use of circular regions of different sizes, however, it is required that these have the same (or comparable) diameters. The circuit is restricted to traverse each region only through a finite set of predefined points. It is shown that the algorithm also has

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

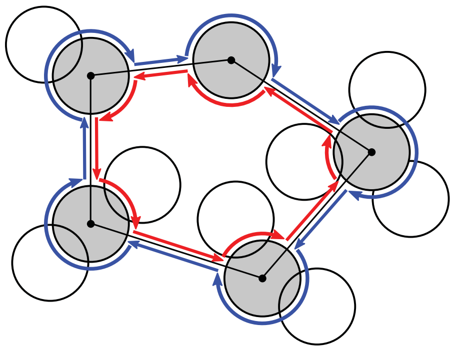
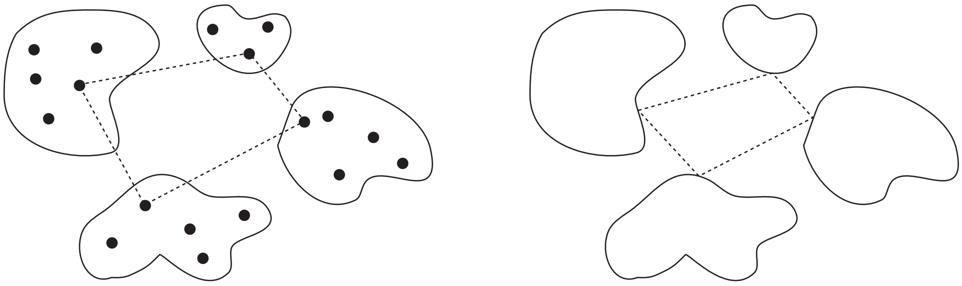


Fig. 8. Solution to the TSPN proposed in ref. [27]. Example of a tour given the maximal independent set (gray circles). Initially, the blue line tour is followed clockwise, and next, the red line tour is followed counterclockwise, guaranteeing that all regions will be visited.



(a) EGTSP (b) TSPN

Fig. 9. Comparison of the circuits obtained considering the (a) EGTSP and (b) TSPN.

constant $O(1)$-factor approximation. For the more general form, and lifting the predefined points, a $O(\log n)$-algorithm is presented.

The Euclidean Group TSP (EGTSP),[29] a similar problem related to the TSPN, was found in the literature. In this specific problem, the points of interest are grouped into regions, and the problem was posed as to generate the shortest path passing through at least one of the points pertaining to each region. Recalling that in the TSPN, the region must be visited (continuous case), but not the exact position of a given point (discrete case). This problem can be modeled as the previously mentioned GATSP.

Figure 9 exemplifies two resulting circuits generated considering the EGTSP and TSPN formulations.

The routing problem considering a mobile sink, such as a robot, for data collection in a WSN is one of the most known problems that use the formulation proposed by the TSPN. The sensor communication radius is the neighborhood of the point, and the mobile agent should be in that region to be able to communicate with the sensor. Different methodologies have been proposed to tackle this problem, from exact optimization approaches to evolutionary algorithms.

The EGTSP also serves as inspiration for the modeling of the data collection problem. In ref. [105], sensor nodes are grouped into different subgroups, and the data collection path should be minimized in such a way that at least the exact position of one sensor node in each subgroup is visited. In ref. [98], the authors also used a model that refers to the EGTSP however, after the sensor nodes were grouped, the path generated did not necessarily pass through the exact position of a node, but on virtual points calculated in the center of the previously defined groups.

Considering the TSPN model, in ref. [108], the sequence of the visit was obtained by solving a TSP instance based on the center of the regions (collecting points), and then three evolutionary algorithms were used to optimize the path by moving these points within the boundaries of the regions.

In ref. [22], it is presented a method based on the Ant Colony Optimization technique, where the main difference to the work of ref. [108] is the fact that the permutation (visit sequence) is also

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

Table II. Comparison of the main works regarding region visiting routing.

Reference	Methodology	Neighborhoods
Ref. [27]	Heuristic	Convex
Ref. [108]	Evolutionary	Convex
Ref. [22]	Evolutionary	Convex
Ref. [37]	Exact	Non-convex
Ref. [2]	Heuristic	Non-convex

embedded in the optimization process. Importantly, in both studies, there is a major assumption, i.e. the regions should not intersect.

The TSPN was also addressed considering the distribution of sensors in the three-dimensional space (applicable especially when the use of UAV or Unmanned Underwater Vehicles are considered). In ref. [107], a solution based on Estimation of Distribution Algorithm for the generation of efficient paths passing through several spherical regions arranged in space is presented.

In addition to being used by mobile robots, the TSPN has also been applied in the context of robotic manipulators. In ref. [37], it presented a scenario where a camera is placed at the end-effector of a manipulator arm to take pictures of objects from different positions. The problem was formulated as a non-convex Mixed-Integer Non-linear Program. In a similar way, ref. [2] addresses the problem of optimizing the sequence of task execution for industrial robots also as the TSPN. It is proposed a tour construction heuristic based upon the ones from the TSP domain, called Constricting 3-Opt.

Table II presents a comparison of the main works regarding vehicle routing for visiting regions, highlighting which specific features each one addresses.

6. Non-Holonomic Vehicle Routing for Visiting Regions

In the following recent works, the combination of restrictions imposed by both the DTSP (motion constraints) and the TSPN (region visit) began to be considered as a single problem, called Dubins TSP with Neighborhoods (DTSPN). In this context, we will further extend the categories of approaches mentioned in Section 4. In the DTSPN, not only the orientations and visiting sequence must be selected, but the actual position of the waypoint with respect to the region (in general is allocated in the boundaries) must also be determined in a coupled or decoupled manner.

The first work to address the DTSPN was ref. [73], where the nomenclature Polygon Visiting DTSP was used. The formulation was developed to capture a scenario where an UAV on a reconnaissance mission should fly over certain regions and photograph all the targets in the shortest time possible. As the targets are static and the vehicle travels at constant speed, the cost function used is composed only of the factor related to the minimization of the path length. The paper presents a solution using a GA where the chromosome is composed of a position and an orientation within each region to be visited. The algorithm is evaluated from a series of numerical simulations and the results are compared with those obtained by a random search algorithm.

In a subsequent work,[74] a sampling-based approach (sampling-based roadmap) was presented. Several configurations (position and orientation) are sampled within the limits of the regions. These samples are called entry positions in the region, once all the samples are oriented toward the interior of the region or at most parallel to the edge. After the sampling is performed, the paths connecting all samples of each region to all other samples in the other regions are calculated, resulting in a connected graph. Then a solution to a GATSP reduced to an instance of the ATSP is calculated. The average case runtime of the entire method is $O(n^{2.2} + \frac{n_{\text{samples}}^2}{n} + n_{\text{samples}})$. The method is resolution complete, which means it provably converges to a non-isolated global optimum as the number of samples grows.

Another method found in the literature presented in ref. [40] consists of an extension of ref. [74] and also made use of a sample-based technique. The main difference is that the samples need not only to be *entry positions*, but may assume any orientation. Furthermore, the methodology can take advantage of the samples which are allocated at intersecting regions, thus generating paths that are shorter than those produced by the aforementioned approach. This was accomplished by using a more general version of the Noon and Bean transformation,[72] which deals with overlapping node sets.

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

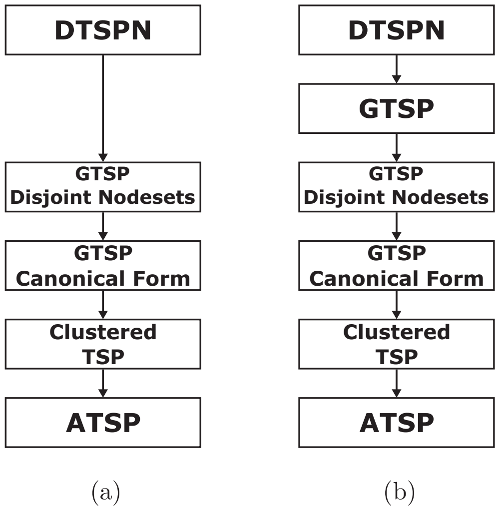


Fig. 10. Comparison of the transformation steps of the algorithms for the DTSPN presented in (a) ref. [74] and (b) ref. [40].

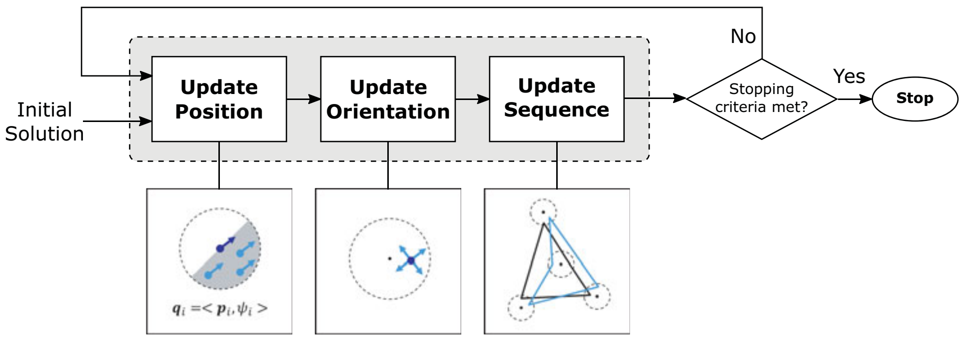


Fig. 11. Illustration of the three basic steps that compose the evolutionary algorithm presented in ref. [60].

Figure 10 presents the transformation steps of the algorithm presented in ref. [74] and the improvement proposed in ref. [40].

In ref. [60], a simple three-stage evolutionary algorithm was proposed to solve both the combinatorial and the continuous steps of the problem in a concerted manner. In the first phase, the method varies the position of the waypoints within the boundaries of each region, next it optimizes the path orientation at each waypoint, and finally it chooses the best actual sequence of visit by finding a solution to an instance of the ATSP. Although it does not consider the intersection among the regions, it provided results which were comparable to ref. [40]. Figure 11 illustrates the steps of the algorithm.

In ref. [103], the use of a Local Iterative Optimization strategy to independently adjust the waypoints' orientations and positions on the regions was proposed, considering that the sequence of visit is already given. Its main advantage is the low computational requirements comparing to other techniques, such as GAs. However, the DTSPN instance must respect the D_4 constraint, which means the Euclidean distance between the regions must be larger than 4ρ.

More recently, the DTSPN was also considered as the basis of the formulation in the context of data collection in WSNs. In ref. [62], a bi-objective evolutionary algorithm was used in order to obtain a minimum length path while maximizing the collecting time (path intersection) at each region.

Table III presents a comparison of the main works regarding non-holonomic vehicle routing for visiting regions, highlighting which specific features were addressed by each one of them.

Table III. Comparison of the main works regarding non-holonomic region visiting routing.

Reference	Methodology	Approach	Neighborhoods
Ref. [73]	Evolutionary	Coupled	Non-convex
Ref. [74]	Sampling-based	Coupled	Non-convex
Ref. [40]	Sampling-based	Coupled	Non-convex
Ref. [60]	Evolutionary	Decoupled	Convex
Ref. [103]	Heuristic*	Decoupled	Non-convex

*The DTSPN instance must respect the D_4 constraint.

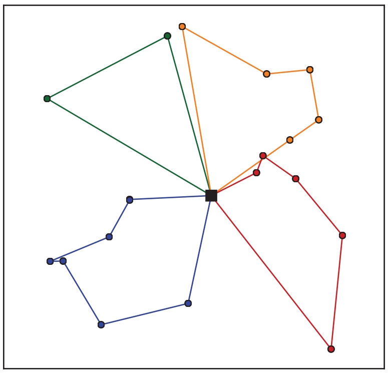


Fig. 12. Example of a solution to the k-TSP considering four vehicles and a single deposit (black square). In this case, the objective function is the minimization of the longest circuit.

7. Multiple Vehicle Routing

The results to the problems discussed in the previous sections consist of a single circuit. An extension of the TSP for the case where paths must be created for more than one agent is referred to as k-TSP. The k-TSP can be described as the problem of generating paths for k salesmen that start in a given city and return to the same city after each one of the other cities has been visited by exactly one of the salesmen. The objective, in this case, may be the minimization of the sum of the length of all paths generated (reduction in energy expenditure) or minimization of the longest circuit (reduction of travel time). A very similar problem to the k-TSP is usually named Multiple TSP. However, unlike the previous problem, the requirement that all circuits have a common base city is not considered.[92]

Figure 12 depicts an example for better visualization and understanding of an instance used as input and the expected result after finding a solution to the k-TSP.

The k-Traveling Salesman Problem (k-TSP) is an instance of the more general problem known as VRP.[23,97] As for the VRP, it is considered for vehicles with a certain capacity constraint to fulfill the demands, while the k-TSP is a formulation where the vehicles have unlimited capacity. This problem is also NP-hard, and a summary of possible formulations and approaches used are presented in ref. [11].

However, since the k-TSP/VRP are classical combinatorial optimization problems, their use in robotics applications are limited, and they serve as the basis for diverse generalizations considering aspects such as motion constraints and inexact visiting positions (neighborhoods).

In ref. [80], the authors presented a path planning algorithm for multiple Dubins vehicles. It was assumed that all points to be visited have a minimum separation from each other of at least twice the size of the minimum radius of curvature of the vehicle. This is one of the main weaknesses of the method since the Dubins metric exert greater influence on the length of the path exactly when the points are close to each other. In addition, the circuits do not necessarily have the same start and arrival points. A constant factor approximation algorithm is presented and the objective function is to minimize the sum of the distances traveled by all vehicles. The technique is divided into two steps: (i) construction of a complete graph considering all vehicles and targets followed by the Minimum Spanning Tree (MST) calculation; (ii) obtaining of an Eulerian graph and calculation of a feasible path through the targets.

In ref. [63], the classical k-TSP problem with motion constraints is formalized, referring to the k-DTSP. The authors proposed a non-linear mathematical formulation to the problem, which was later

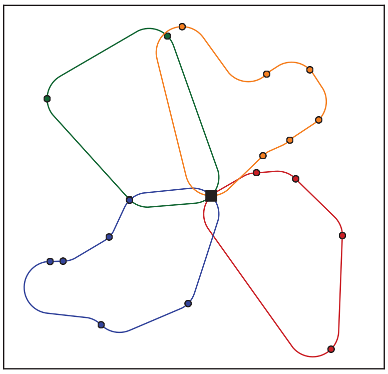


Fig. 13. Solution to the *k*-DTSP considering four vehicles and a single deposit (black square).

solved assuming a combinatorial approach based on the discretization of the vehicle's orientation at each point. Since it is an exact formulation approach, its use is restricted to a reduced number of points. Figure 13 illustrates the solution of a *k*-Dubins TSP (*k*-DTSP) instance.

The route generation for more than one vehicle was also related to a task allocation problem. For example, in refs. [7,70], distributed algorithms are presented for previously known demand allocation (static case) considering different vehicles. Although it was considered a limited communication between vehicles, the vehicle movement restrictions were not treated. In ref. [70], a distributed auction algorithm to spatially distributed tasks were presented, while in ref. [7], a game-theoretical formulation was proposed.

Despite the vehicle's curvature constraint being considered in ref. [86] (the vehicles are fixed-wing UAVs), the problem of target allocation and path planning were treated separately. The allocation is made from the ordering of the targets according to the Euclidean distance, and the route generation made using the Dubins metric. The Dubins path for each segment was optimized given an initial heading condition and an open final heading condition. In order to avoid collision among the vehicles, tours with different altitude were used.

The *k*-TSP has also been used as the basis to model the problem of data collection in WSNs using multiple mobile robots.[14,95] Although the communication radius (neighborhood) was initially mentioned in ref. [95], however, in ref. [14], it was actually considered during the generation of the final path. In addition, as seen in both works, the motion constraints of the vehicles were not considered by the methodology. These works consist of a generalization to the TSPN, being the first works to introduce the *k*-TSPN. The objective function in this specific case was to minimize the time to download the data from all sensors. The methodology solves the TSPN and *k*-TSP separately by combining two classical techniques. First, the method presented in ref. [27] was used to solve the TSPN part (Fig. 8). Next, the tour was separated into multiple paths by applying the *k*-SPLITOUR.[35] Figure 14 shows a solution to an example instance considering the methodology proposed in ref. [14].

The authors of ref. [46] have also studied the *k*-TSPN. Their work proposed a constant factor approximation algorithm, considering that the neighborhoods consist of a uniform circular area that must be visited by some tour and that the length of the longest tour should be minimized (data collection latency). However, the kinematics of possible real vehicles were not taken into account, and the mobile agents do not have a common initial/end point (i.e. *k*-rooted paths). They introduced the General Minimum Spanning Tree with Neighborhood (GMSTN) problem, whose goal was stated as follows: Given a set of circular neighborhoods, which may touch or overlap each other, find an MST of this set. Finally, an algorithm to the GMSTN which was later transformed to the *k*-TSPN (with *k* bases) was proposed.

Ref. [61] introduced the *k*-DTSPN, which considers the problem of planning efficient paths among target regions for multiple robots with a minimum turning radius constraint. As seen, the paper presented two approaches for the problem. Initially, it presented a heuristic that solved the problem in two distinct steps: (i) a TSP instance is solved considering Euclidean costs and then separated in *k* tours (Algorithm 1); (ii) determine the Dubins curves between the positions in order to make the whole path attainable by non-holonomic vehicles (apply the AA[85]). The use of a Memetic Algorithm[71] was proposed next to solve both the combinatorial and continuous parts of the problem in a combined

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

Table IV. Comparison of the main works regarding multiple vehicle routing.

Reference	Methodology	Motion constraints	Single base	Regions
Ref. [80]	Exact	•		
Ref. [86]	Heuristic	•	•	
Ref. [14]	Heuristic		•	Convex
Ref. [61]	Heuristic	•	•	Convex
Ref. [61]	Evolutionary	•	•	Convex
Ref. [46]	Heuristic			Convex

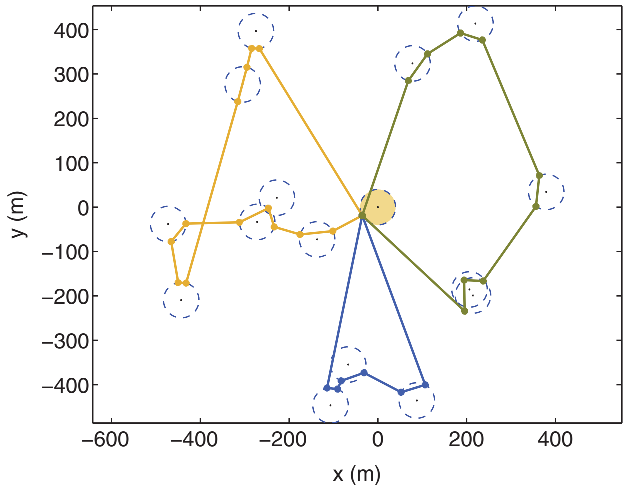


Fig. 14. Data gathering tour in wireless sensor networks using multiple vehicles. The goal is to minimize the total collecting time, as proposed in ref. [14].

manner. Figure 15 presents the resulting tours obtained by both algorithms. The upper bound for the length of the longest route obtained using Algorithm 1 can be obtained based on the bound of both techniques used, which is given by

$$\mathcal{T}_{\max} \leq \frac{1}{k}(L - 2c_{\max}) + 2c_{\max} + \left\lceil \frac{N}{2} \right\rceil \tau\rho\pi, \tag{6}$$

where $\tau \in [2.657, 2.658]$.

Algorithm 1 k-SPLITOUR($\mathcal{Q}$, k) ref. [35]

1: Find a 1-tour (TSP) $\mathcal{T}_0 = \langle \mathbf{q}_0, \mathbf{q}_1, \ \ldots, \mathbf{q}_n, \mathbf{q}_0 \rangle$, where $\mathbf{q}_0$ is the initial vertex (base).

2: For each j, $1 \leq j < k$, find the last vertex $\mathbf{q}_{i(j)}$ such that the cost of the path from $\mathbf{q}_0$ to $\mathbf{q}_{i(j)}$ along $\mathcal{T}_0$ is no greater than $(j/k)(L - 2c_{\max}) + c_{\max}$, where L is the Euclidean length of the circuit found in Step 1 and

$$c_{\max} = \max_n \ \|\mathbf{q}_0 - \mathbf{q}_n\|. \tag{7}$$

3: Build the k tours as $\mathcal{T}_1 = \langle \mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_{i(1)}, \mathbf{q}_0 \rangle$, $\mathcal{T}_2 = \langle \mathbf{q}_0, \mathbf{q}_{i(1)+1}, \ldots, \mathbf{q}_{i(2)}, \mathbf{q}_0 \rangle$, $\ldots$, $\mathcal{T}_k = \langle \mathbf{q}_0, \mathbf{q}_{i(k-1)}, \ldots, \mathbf{q}_n, \mathbf{q}_0 \rangle$.

Table IV presents a comparison of the main works regarding multiple vehicle routing, highlighting which specific features each one addresses.

In general, most of the studies in the literature that addressed the vehicle routing problem were focused primarily on static environments, i.e. they do not consider the case where new demands are inserted over time, which is a limitation in many different scenarios and applications such as surveillance, search and rescue, and data collection in WSNs.

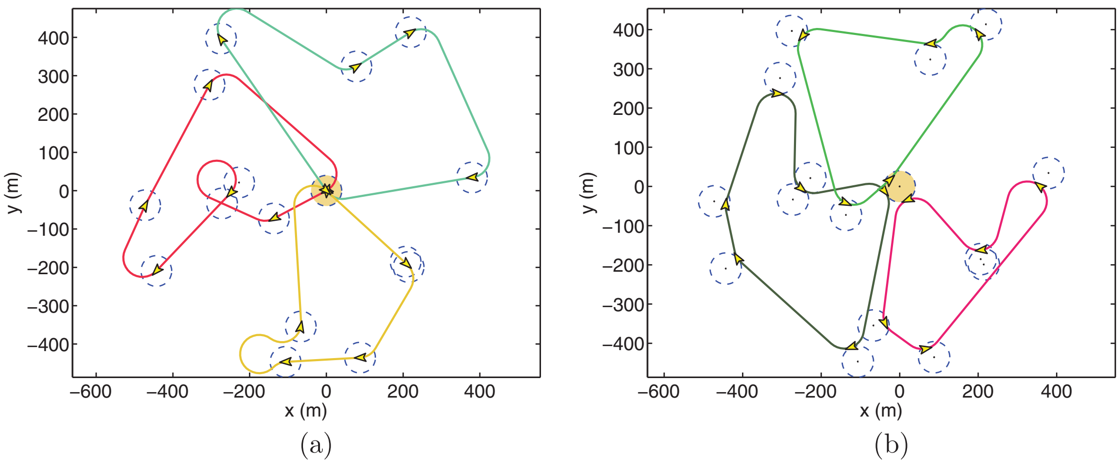


Fig. 15. Example solutions for the k-Dubins TSP with Neighborhoods (k-DTSPN) obtained by (a) Heuristic and (b) Memetic Algorithm, both proposed in ref. [61].

8. Dynamic Vehicle Routing

In this section, the dynamic vehicle routing problem will be addressed and discussed. Unlike the cases presented in the previous sections, the dynamic case considers the scenario where one does not have all the information about the demands of the environment (specifically the points or regions to be visited) prior to the generation of the route. Therefore, it is necessary to replan whenever new information becomes available.

This problem was initially introduced by ref. [79], where demands were arranged in a graph, and it was termed Dynamic Traveling Repairman Problem (DTRP). The first work that dealt with the problem of demands placed on the Euclidean plane was ref. [12], where only one vehicle was used to fulfill the demands. In a following work,[13] the technique was extended to the case of multiple vehicles with limited capacity, being called mDTRP.

The first work that addressed the online version of the TSP for a general metric space was ref. [9], where an optimal algorithm was presented. The online version for the asymmetric case (ATSP) has also been the focus of the study.[8] In ref. [43], a brief review of the literature related to the online routing problem was presented.

The dynamic routing problem has different aspects to it which should be considered when developing new techniques, for example, the metric used by the cost function, type of architecture (centralized/decentralized), vehicle characteristics (motion constraints, number of vehicles), among others.

An approach widely used to deal with dynamic problems is called *Online Algorithms*. The term *online* here refers to the characteristic of the methods to deal with problems where the solution needs to be calculated incrementally since it has no knowledge of the entire instance *a priori*. The main focus of this approach is the development of efficient algorithms (mostly heuristics), i.e. techniques to generate good solutions quickly, however, without performance guarantees.[43]

Figure 16 exemplifies the dynamic nature of the problem with the inclusion of new demands on the environment, while the vehicle is already executing the path initially planned. As it can be seen, the path (untraveled portion) must be efficiently reshaped to cover new points (or regions) of interest that may appear over time, after the vehicle has departed the base.

Regarding the metric being used for the cost function, it is possible to mention as examples of possible objectives: minimizing the energy used by the vehicle; minimizing the waiting time to attend to a demand; minimizing the path length and minimizing the number of vehicles being used.

The work of ref. [12] introduced the use of queue-based methods. In this sense, the most intuitive and simple manner to solve a DTRP instance is to attend new demands in the order in which they arrive. This first-come, first-served (FCFS) policy may be defined as (i) when there are unserved demands in the environment, the agent travels directly from one demand location to the next following a FCFS order and (ii) when no unserved demands are present, the agent remains stationary until a new demand arrives. Considering the use of multiple vehicles, this policy can be generalized as follows:

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

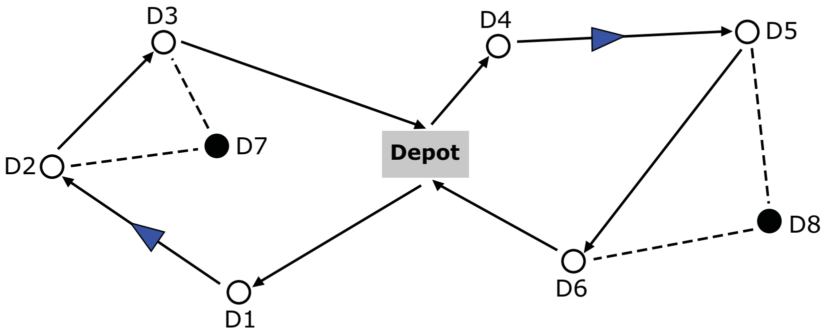


Fig. 16. Illustration of the dynamic vehicle routing problem. The initial paths (solid lines) for two vehicles (blue triangles) and the necessary modifications (dashed lines) to visit two new demands (black circles) placed after the start of the navigation.

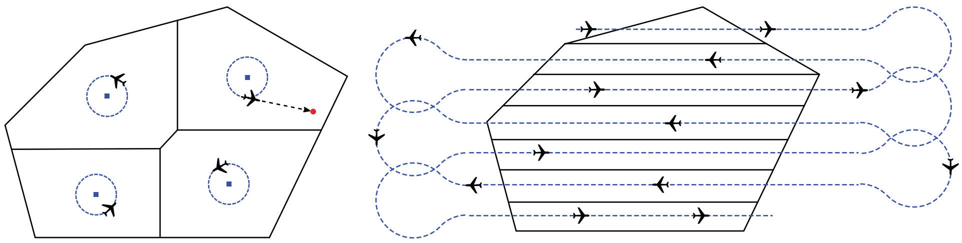


Fig. 17. Example strategies for the DTRP based on environment partition, as proposed in ref. [31]. (a) Median Circling Policy and (b) Strip Loitering Policy.

> **The *m* stochastic queue median (SQM) policy**[13]: Given an environment $\mathcal{E}$, determine the m median locations and locate a vehicle in each one of these locations. Every time a new demand arrives, assign it to the vehicle in the nearest median location. The vehicles must service all the demands in a FCFS order, returning to the median position after completion.

Many methodologies in the literature addressing the DTRP consist of centralized policies, i.e. a central entity is responsible for calculating all routes and informing each vehicle about its respective route.[13] However, when using multiple vehicles, it is interesting to use distributed techniques, thus increasing scalability as well as making the system more robust to the possible failure of this single controlling entity. Among the works that proposed distributed solutions are refs. [6,34,78]. The overall strategy is basically to partition the environment into k openly disjoint subregions, and each vehicle will be responsible to serve demands on its own region given a local policy, e.g. FCFS.

The literature in the Optimization Research area for the DTRP usually deals with the problem in a more abstract way, not directly considering issues that can arise in a real instantiation, for example, the physical limitations of the vehicle among others. The first work to consider the curvature restriction of the vehicle was ref. [84], where the vehicle was driven in a zig-zag pattern motion across the environment. This zig-zag occurs because the environment is divided into small subregions following a format that resembles a bead, due to the characteristic shape of the motion made by vehicles with curvature constraints, consequently, the algorithm was named *Bead-Tiling*.

In ref. [31], two different solutions were presented. The first was to divide the environment into subregions according to a Voronoi diagram, where the vehicles must remain in the centroid of each region until the insertion of a new demand when it is seen to be moving up toward it (called *Median Circling Policy*). The following solution divides the environment into horizontal bands of movement (corridors), therefore having the vehicles cover the entire environment by following these corridors (called *Strip Loitering Policy)*. Figure 17 illustrates the previously mentioned strategies.

Despite the good results presented by the aforementioned solutions when the objective is to minimize the waiting time of the demands, the energy budget is large, since vehicles are in motion

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

Table V. Comparison of the main works regarding dynamic vehicle routing.

Reference	Methodology	Motion constraints	Environment
Ref. [13]	Heuristic		Sub-regions
Ref. [78]	Heuristic		Partitioned
Ref. [84]	Pattern	•	Full
Ref. [31]	Heuristic	•	Partitioned
Ref. [31]	Pattern	•	Partitioned

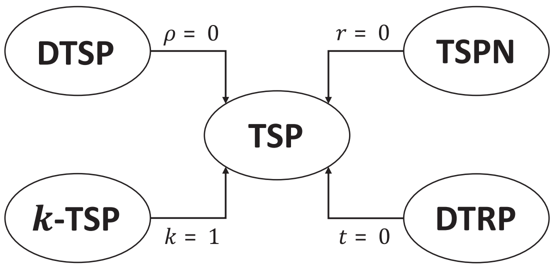


Fig. 18. Illustration of the generalizations of the TSP considering different variables.

during the entire time (even when there are no new demands to attend). This problem could be minimized if the vehicles could return to a common base.

Ref. [10] proposed dispatching rules and loitering policies for multiple UAVs to respond to fixed-location, multiple-priority demands that are dynamically inserted into the environment. The main objective was to design a set of fast-responding methods that can be used in real time. They considered three basic decision instances: (i) where to send idle UAVs; (ii) which UAV should service a demand and (iii) which target a UAV should attend. However, the policies do not directly take into account real characteristics of the system when planning the navigation of the vehicles.

Several other characteristics can also be analyzed in the context of dynamic routing, for example, restrictions on the visit time (time window) to a particular demand,[76,77] demands with different service priorities,[90] moving demands[15,89] and vehicles with a limited knowledge of the environment.[30] In ref. [17], a general overview of the challenges and main existing techniques to the problem of dynamic vehicle routing regarding robotic systems was presented.

Table V presents a comparison of the main works regarding dynamic vehicle routing, highlighting which specific features each one addresses.

9. Conclusion

The problem of planning paths for mobile agents based on length or time optimization is essential and of great importance, and has been the goal of several research fields, especially robotics. In this work, a broad and comprehensive review of the literature is presented, discussing the fundamental generalizations especially regarding the use of TSP-like problems in robotic systems. We have mainly considered the following characteristics: (i) motion constraints; (ii) inexact visit position (neighborhoods); (iii) multiple vehicles and (iv) dynamic scenarios.

Figure 18 illustrates how these generalizations converge to the original TSP accordingly to the minimum turning radius (ρ), region radius (r), number of vehicles (k) and demand time of arrival (t).

However, several challenges requiring further research effort are foreseen in this problem domain. Following, we present and discuss possible future research directions:

- **Dubins TSP with Neighborhoods:** The DTSPN deals with the combination of restrictions imposed by both the DTSP (motion constraints) and the TSPN (region visit) and a good algorithmic solution remains an open problem. The few techniques found in the literature are mostly based on evolutionary strategies[60,73] or sampling-based approaches.[40,74] Therefore, the development of simpler and more efficient methods, for example, a closed-form heuristic,[102] is still an open research topic.

https://doi.org/10.1017/S0263574718000735 Published online by Cambridge University Press

Table VI. Main works in the literature and characteristics addressed by them.

Reference	Motion constraints	Neighborhoods	Multiple vehicles	Dynamic routing
Ref. [85]	•			
Ref. [57]	•			
Ref. [54]	•			
Ref. [59]	•			
Ref. [41]	•			
Ref. [28]		•		
Ref. [108]		•		
Ref. [22]		•		
Ref. [74]	•	•		
Ref. [40]	•	•		
Ref. [60]	•	•		
Ref. [80]	•		•	
Ref. [86]	•		•	
Ref. [95]		•	•	
Ref. [14]		•	•	
Ref. [46]		•	•	
Ref. [61]	•	•	•	
Ref. [9]				•
Ref. [84]	•			•
Ref. [6]			•	•
Ref. [78]			•	•
Ref. [31]	•		•	•
Ref. [10]	•		•	•

- **Multiple vehicles:** Although few works have dealt with the multiple agents considering Dubins vehicles[80] and neighborhoods,[14,61] the proposition of more general methods for groups of heterogeneous robots (i.e. with different curvature constraints and velocities) is also a possible research field.
- **Dynamic demands:** Most of the current literature about routing problems usually considers static instances, i.e. the vehicle is dispatched to follow an immutable, previously planned path. Therefore, methodologies do not take into account the intrinsically dynamic nature of many different tasks. For example, in a WSN, new nodes (or those awakening from sleep states) must be attended by the vehicle as soon as possible. Most of the approaches in the literature are based on environment segmentation[31] or loitering strategies according to a predefined pattern,[84] in both cases, there is a high-energy consumption since it is not considered a closed circuit with a central base. Since the demands arrive in an online fashion, it is fundamental to develop heuristics capable of efficiently adapting to the current vehicle path since exact optimization approaches are not able to deal with large instances.
- **Three-dimensional space:** Three-dimensional path planning is a fundamental task concerning different types of robots, like fixed-wing UAV or underwater Remotely Operated Vehicles. Even though it might be possible for such robots to traverse the environment solely in a reactive way, the competence of planning efficient paths in advance is an important feature. However, most of the works in the literature have considered only point-to-point three-dimensional paths[39,75] and not a closed circuit of visit through multiple points. In this case, the consideration of additional constraints such as flight-path angle and torsion is also an important topic for future investigation.
- **Environments with obstacles:** In order to deal with more realistic environments containing obstacles, most of the solutions for routing problems are based on probabilistic approaches such as RRTs, PRM or Evolutionary Algorithms. The main drawback with such strategies is that they usually have as their main focus path feasibility (vehicle and environment constraints), usually neglecting the length of the generated path. Moreover, very few works in the literature have considered TSP-like problems in environments with obstacles. Furthermore, in the case of multiple vehicles, a time-dependent trajectory should be considered instead of a simple path dwelling more on collision avoidance among the agents.

- **Benchmarks and datasets:** Several test instances for different optimization problems (e.g. TSP) can be found in the literature. However, when considering more complex scenarios with restrictions such as motion constraints, neighborhoods and multiple vehicles, there are still no benchmarks available. Hence, the creation of general test instances for the different problems (DTSP, TSPN, DTSPN, k-DTSP, k-TSPN and k-DTSPN) would allow researches to better evaluate and compare new proposed techniques.

Finally, Table VI summarizes and presents a broad comparison of the main works in the literature relating to the vehicle routing problem related to robotic systems, highlighting the specific characteristics addressed by each one them.

Acknowledgments
This work was developed with the support of CNPq, CAPES and FAPEMIG.

References

1. A. Aggarwal, D. Coppersmith, S. Khanna, R. Motwani and B. Schieber, "The Angular-Metric Traveling Salesman Problem," *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms SODA1997*, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics (1997) pp. 221–229.
2. S. Alatartsev, V. Mersheeva, M. Augustine and F. Ortmeier, "On Optimizing A Sequence of Robotic Tasks," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems IROS* (2013).
3. A. Alves Neto, D. G. Macharet and M. F. M. Campos, "Feasible RRT-Based Path Planning Using Seventh Order Bézier Curves," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS2010* (Oct. 2010) pp. 1445–1450.
4. D. L. Applegate, R. E. Bixby, V. Chvatal and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*, Princeton Series in Applied Mathematics (Princeton University Press, Princeton, NJ, USA, 2007).
5. E. M. Arkin and R. Hassin, "Approximation algorithms for the geometric covering salesman problem," *Discrete Appl. Math.* **55**, 197–218 (Dec. 1994).
6. A. Arsie, K. Savla and E. Frazzoli, "Efficient routing algorithms for multiple vehicles with no explicit communications," *IEEE Trans. Autom. Control* **54**(10), 2302–2317 (Oct. 2009).
7. G. Arslan, J. R. Marden and J. S. Shamma, "Autonomous vehicle-target assignment: A game-theoretical formulation," *J. Dyn. Syst. Meas. Control* **129**(5), 584–596 (2007).
8. G. Ausiello, V. Bonifaci and L. Laura, "The on-line asymmetric traveling salesman problem," *J. Discrete Algorithms* **6**(2mo), 290–298 (2008). Selected papers from CompBioNets 2004 - Algorithms and Computational Methods for Biochemical and Evolutionary Networks.
9. G. Ausiello, E. Feuerstein, S. Leonardi, L. Stougie and M. Talamo, "Algorithms for the on-line travelling salesman," *Algorithmica* **29**, 560–581 (2001).
10. N. Bednowitz, R. Batta and R. Nagi, "Dispatching and loitering policies for unmanned aerial vehicles," *J. Simul.* **8**(1), 9–24 (Feb. 2012).
11. T. Bektas, "The multiple traveling salesman problem: An overview of formulations and solution procedures," *Omega* **34**(3), 209–219 (2006).
12. D. J. Bertsimas and G. van Ryzin, "A stochastic and dynamic vehicle routing problem in the Euclidean plane," *Oper. Res.* **39**(4), 601–615 (1991).
13. D. J. Bertsimas and G. van Ryzin, "Stochastic and dynamic vehicle routing in the Euclidean plane with wultiple capacitated vehicles," *Oper. Res.* **41**(1), 60–76 (1993).
14. D. Bhadauria, O. Tekdas and V. Isler, "Robotic data mules for collecting data over sparse sensor fields," *J. Field Robot.* **28**(3), 388–404 (2011).
15. S. Bopardikar, S. Smith, F. Bullo and J. Hespanha, "Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies," *IEEE Trans. Autom. Control* **55**(11), 2554–2569 (Nov. 2010).
16. J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Trans. Robot. Autom.* **7**(3), 278–288 (Jun. 1991).
17. F. Bullo, E. Frazzoli, M. Pavone, K. Savla and S. Smith, "Dynamic vehicle routing for robotic systems," *Proc. IEEE* **99**(9), 1482–1504 (Sep. 2011).
18. A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Trans. Syst. Man, Cybern. - Part A: Syst. Humans* **28**(5), 562–574 (Sep. 1998).
19. H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations* (MIT Press, Cambridge, MA, Jun. 2005).
20. N. Christofides, "Technical note–bounds for the travelling-salesman problem," *Oper. Res.* **20**(5), 1044–1056 (1972).
21. J. Cobano, R. Conde, D. Alejo and A. Ollero, "Path Planning Based on Genetic Algorithms and the Monte-Carlo Method to Avoid Aerial Vehicle Collisions Under Uncertainties," *Proceedings of the IEEE International Conferenceon Robotics and Automation ICRA2011* (May 2011) pp. 4429–4434.

22. G. Comarela, K. Gonçalves, G. L. Pappa, J. Almeida and V. Almeida, "Robot Routing in Sparse Wireless Sensor Networks with Continuous Ant Colony Optimization," *Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation GECCO2011*, New York, NY, USA: ACM (2011).
23. G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manag. Sci.* **6**(1), 80–91 (1959).
24. M. de Berg, J. Gudmundsson, M. J. Katz, C. Levcopoulos, M. H. Overmars and A. F. van der Stappen, "TSP with neighborhoods of varying size," *J. Algorithms* **57**, 22–36 (Sep. 2005).
25. J. P. Desai and V. Kumar, "Motion planning for cooperating mobile manipulators," *J. Robot. Syst.* **16**(10), 557–579 (1999).
26. L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *Am. J. Math.* **79**(3), 497–516 (1957).
27. A. Dumitrescu and J. S. Mitchell, "Approximation algorithms for TSP with neighborhoods in the plane," *J. Algorithms* **48**(1), 135–159 (2003). Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms.
28. K. Elbassioni, A. Fishkin and R. Sitters, "On Approximating the TSP with Intersecting Neighborhoods," **In:** *Algorithms and Computation* (T. Asano, ed.) Lecture Notes in Computer Science, vol. 4288 (Springer, Berlin, Heidelberg, 2006) pp. 213–222.
29. K. M. Elbassioni, A. V. Fishkin, N. H. Mustafa and R. Sitters, "Approximation Algorithms for Euclidean Group TSP," **In**: *Automata, Languages and Programming* (L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi and M. Yung, eds.) Lecture Notes in Computer Science, vol. 3580 (Springer, Berlin, Heidelberg, 2005) pp. 1115–1126.
30. J. Enright and E. Frazzoli, "Cooperative UAV Routing with Limited Sensor Range," *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Keystone, CO, USA (2006).
31. J. J. Enright, K. Savla, E. Frazzoli and F. Bullo, "Stochastic and dynamic routing problems for multiple UAVs," *AIAA J. Guidance, Control, Dyn.* **32**(4), 1152–1166 (2009).
32. P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *Int. J. Robot. Res.* **17**(7), 760–772 (1998).
33. D. Fox, W. Burgard and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.* **4**(1), 23–33 (1997).
34. E. Frazzoli and F. Bullo, "Decentralized Algorithms for Vehicle Routing in a Stochastic Time-Varying Environment," *Proceedings of the 43rd IEEE Conference on Decision and Control CDC2004*, vol. 4 (Dec. 2004) pp. 3357–3363.
35. G. N. Frederickson, M. S. Hecht and C. E. Kim, "Approximation algorithms for some routing problems," *SIAM J. Comput.* **7**(2), 178–193 (1978).
36. O. Gal, Z. Shiller and E. Rimon, "Efficient and Safe on-Line Motion Planning in Dynamic Environments," *Proceedings of the IEEE International Conference on Robotics and Automation* (2009) pp. 88–93.
37. I. Gentilini, F. Margot and K. Shimada, "The travelling salesman problem with neighbourhoods: MINLP solution," *Optim. Methods Softw.* **28**(2), 364–378 (2013).
38. J. Gudmundsson and C. Levcopoulos, "A fast approximation algorithm for TSP with neighborhoods," *Nordic J. Comput.* **6**, 469–488 (Dec. 1999).
39. S. Hota and D. Ghose, "Optimal Path Planning for an Aerial Vehicle in 3D Space," *Proceedings of IEEE Conference on Decision and Control CDC* (Dec. 2010) pp. 4902–4907.
40. J. T. Isaacs, D. J. Klein and J. P. Hespanha, "Algorithms for the Traveling Salesman Problem With Neighborhoods Involving a Dubins Vehicle," *Proceedings of the IEE American Control Conference ACC2011* (2011), pp. 1704–1709.
41. P. Isaiah and T. Shima, "Motion planning algorithms for the Dubins travelling salesperson problem," *Automatica* **53**(0), 247–255 (2015).
42. L. Jaillet, J. Cortés and T. Siméon, "Sampling-based path planning on configuration-space costmaps," *IEEE Trans. Robot.* **26**(4), 635–646 (2010).
43. P. Jaillet and M. R. Wagner, "Online Vehicle Routing Problems: A Survey," **In**: *The Vehicle Routing Problem: Latest Advances and New Challenges* (B. Golden, S. Raghavan, E. Wasil, R. Sharda and S. Voß, eds.) Operations Research/Computer Science Interfaces Series, vol. 43 (Springer, USA, 2008) pp. 221–237.
44. S. Karaman and E. Frazzoli, "Optimal Kinodynamic Motion Planning Using Incremental Sampling-Based Methods," *Proceedings of the IEEE Conference on Decision and Control CDC2010* (Dec. 2010) pp. 7681–7687.
45. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.* **5**(1), 90–98 (1986).
46. D. Kim, R. Uma, B. Abay, W. Wu, W. Wang and A. Tokuta, "Minimum latency multiple data MULE trajectory planning in wireless sensor networks," *IEEE Trans. Mobile Comput.* **13**(4), 838–851 (Apr. 2014).
47. Y. Koren and J. Borenstein, "Potential Field Methods and their Inherent Limitations for Mobile Robot Navigation," *Proceedings of the IEEE International Conference on Robotics and Automation* (1991) pp. 1398–1404.
48. Y. Kuwata, G. Fiore, J. Teo, E. Frazzoli and J. How, "Motion Planning for Urban Driving Using RRT," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS2008* (Sep. 2008) pp. 1681–1686.

49. S. M. LaValle," *Planning Algorithms* (Cambridge University Press, New York, NY, USA, 2006).
50. S. M. LaValle and J. Kuffner, J. J., "Randomized Kinodynamic Planning," *Proceedings of the International Conference on Robotics and Automation* (1999) pp. 473–479.
51. J. Le Ny, Performance Optimization for Unmanned Vehicle Systems *Ph.D. Thesis* (Cambridge, MA: Massachusetts Institute of Technology, 2008).
52. J. Le Ny and E. Feron, "An Approximation Algorithm for the Curvature-Constrained Traveling Salesman Problem," *Proceedings of the 43rd Annual Allerton Conference on Communications, Control and Computing* (2005).
53. J. Le Ny, E. Feron and E. Frazzoli, "On the dubins traveling salesman problem," *IEEE Trans. Autom. Control* **57**(1), 265–270 (Jan. 2012).
54. J. Le Ny, E. Frazzoli and E. Feron, "The Curvature-Constrained Traveling Salesman Problem for High Point Densities," *Proceedings of the 46th IEEE Conference on Decision and Control CDC2007* (Dec. 2007) pp. 5985–5990.
55. S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Oper. Res.* **21**(2), 498–516 (1973).
56. T. Lozano-Pérez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Commun. ACM* **22**(10), 560–570 (1979).
57. X. Ma and D. A. Castañón, "Receding Horizon Planning for Dubins Traveling Salesman Problems," *Proceedings of the 45th IEEE Conference on Decision and Control CDC2006* (Dec. 2006) pp. 5453–5458.
58. D. G. Macharet, A. Alves Neto and M. F. M. Campos, "Feasible UAV Path Planning Using Genetic Algorithms and Bézier Curves," *Proceedings of the 20th Brazilian Conference on Advances in Artificial Intelligence SBIA2010* Berlin, Heidelberg: Springer-Verlag (2010) pp. 223–232.
59. D. G. Macharet, A. Alves Neto, V. F. da Camara Neto and M. F. M. Campos, "Nonholonomic Path Planning Optimization for Dubins' Vehicles," *Proceedings of the IEEE International Conference on Robotics and Automation ICRA2011* (May 2011) pp. 4208–4213.
60. D. G. Macharet, A. Alves Neto, V. F. da Camara Neto and M. F. M. Campos, "An Evolutionary Approach for the Dubins' Traveling Salesman Problem with Neighborhoods," *Proceedings of the 21th Genetic and Evolutionary Computation Conference GECCO* (2012).
61. D. G. Macharet, A. Alves Neto, V. F. da Camara Neto and M. F. M. Campos, "Efficient Target Visiting Path Planning for Multiple Vehicles with Bounded Curvature," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS* (Nov. 2013) pp. 3830–3836.
62. D. G. Macharet, J. W. Monteiro, G. R. Mateus and M. F. Campos, "Bi-objective data gathering path planning for vehicles with bounded curvature," *Comput. Oper. Res.* **84**, 195–204 (2017).
63. D. G. Macharet, J. W. G. Monteiro, G. R. Mateus and M. F. M. Campos, "Time-Optimized Routing Problem for Vehicles with Bounded Curvature," *Proceedings of the XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium LARS/SBR* (Oct. 2016).
64. D. G. Macharet, A. Neto, V. da Camara Neto and M. Campos, "Data Gathering Tour Optimization for Dubins' Vehicles," *Proceedings of the IEEE Congress on Evolutionary Computation CEC* (Jun. 2012) pp. 1–8.
65. M. Mahmoodi, K. Alipour and H. B. Mohammadi, "KidVO: A kinodynamically consistent algorithm for online motion planning in dynamic environments," *Ind. Robot: Int. J.* **43**(1), 33–47 (01 2016).
66. J. D. Marble and K. Bekris, "Towards Small Asymptotically Near-Optimal Roadmaps," *Proceedings of the IEEE International Conference on Robotics and Automation ICRA* (2012) pp. 2557–2562.
67. C. S. Mata and J. S. B. Mitchell, "Approximation Algorithms for Geometric Tour and Network Design Problems," *Proceedings of the 11th Annual Symposium on Computational Geometry SCG1995*, New York, NY, USA: ACM (1995) pp. 360–369.
68. A. Medeiros and S. Urrutia, "Discrete optimization methods to determine trajectories for Dubins' vehicles," *Electron. Notes Discrete Math.* **36**, 17–24 (2010). ISCO 2010 - International Symposium on Combinatorial Optimization.
69. J. S. B. Mitchell, "A PTAS for TSP with Neighborhoods Among Fat Regions in the Plane," *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms SODA2007*, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics (2007) pp. 11–18.
70. B. Moore and K. Passino, "Distributed task assignment for mobile agents," *IEEE Trans. Autom. Control* **52**(4), 749–753 (Apr. 2007).
71. P. Moscato and C. Cotta, "A Modern Introduction to Memetic Algorithms," **In**: *Handbook of Metaheuristics* (M. Gendreau and J.-Y. Potvin, eds.) International Series in Operations Research & Management Science, vol. 146, (Springer, USA, 2010) pp. 141–183.
72. C. E. Noon and J. C. Bean, "An efficient transformation of the generalized traveling salesman problem. Technical Report Technical Report 91-26, University of Michigan (1991).
73. K. J. Obermeyer, "Path planning for a UAV Performing Reconnaissance of Static Ground Targets in Terrain," *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Chicago, IL, USA (Aug. 2009).
74. K. J. Obermeyer, P. Oberlin and S. Darbha, "Sampling-Based Roadmap Methods for a Visual Reconnaissance UAV," *Proceedings of the AIAA Conference on Guidance, Navigation and Control*, Toronto, ON, Canada (Aug. 2010).

75. M. Owen, R. Beard and T. McLain, "Implementing Dubins Airplane Paths on Fixed-Wing UAVs," **In**: *Handbook of Unmanned Aerial Vehicles* (K. P. Valavanis and G. J. Vachtsevanos, eds.) (Springer, Netherlands, 2014) pp. 1677–1701.
76. M. Pavone, N. Bisnik, E. Frazzoli and V. Isler, "A stochastic and dynamic vehicle routing problem with time windows and customer impatience," *Mobile Netw. Appl.* **14**, 350–364 (Jun. 2009).
77. M. Pavone and E. Frazzoli, "Dynamic Vehicle Routing with Stochastic Time Constraints," *Proceedings of IEEE International Conference on Robotics and Automation ICRA2010* (May 2010) 1460–1467.
78. M. Pavone, E. Frazzoli and F. Bullo, "Adaptive and distributed algorithms for vehicle routing in a stochastic and dynamic environment," *IEEE Trans. Autom. Control* **56**(6), 1259–1274 (Jun. 2011).
79. H. N. Psaraftis, "Dynamic Vehicle Routing Problems," **In:** *Vehicle Routing: Methods and Studies* (B. L. Golden and A. A. Assad, eds.) Studies in Management Science and Systems, vol. 16 (North-Holland, Amsterdam, 1988) pp. 223–248.
80. S. Rathinam, R. Sengupta and S. Darbha, "A resource allocation algorithm for multivehicle systems with nonholonomic constraints," *IEEE Trans. Autom. Sci. Eng.* **4**(1), 98–104 (Jan. 2007).
81. J. A. Reeds and L. A. Shepp, "Optimal paths for a car that goes both forwards and backwards. *Pac. J. Math.* **145**(2), 367–393 (1990).
82. K. H. Rosen, *Discrete Mathematics and Its Applications*, 7th ed. (McGraw-Hill Higher Education, Boston, 2012).
83. S. Safra and O. Schwartz, "On the complexity of approximating TSP with neighborhoods and related problems," *Comput. Complexity* **14**, 281–307 (Mar. 2006).
84. K. Savla, F. Bullo and E. Frazzoli, "On Traveling Salesperson Problems for Dubins' Vehicle: Stochastic And Dynamic Environments," *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference CDC-ECC2005* (Dec. 2005) pp. 4530–4535.
85. K. Savla, E. Frazzoli and F. Bullo, "On the Point-to-Point and Traveling Salesperson Problems for Dubins' Vehicle," *Proceedings of the IEE American Control Conference ACC2005*, vol. 2 (Jun. 2005) pp. 786–791.
86. T. Shima, S. Rasmussen and D. Gross, "Assigning micro UAVs to task tours in an Urban terrain," *IEEE Trans. Control Systems Technol.* **15**(4), 601–612 (Jul. 2007).
87. A. M. Shkel and V. Lumelsky, "Classification of the Dubins set," *Robot. Auton. Syst.* **34**(4), 179–202 (2001).
88. R. Siegwart, I. R. Nourbakhsh and D. Scaramuzza," *Introduction to Autonomous Mobile Robots*, 2nd ed. (MIT Press, Cambridge, MA, USA, 2011).
89. S. Smith, S. Bopardikar and F. Bullo, "A Dynamic Boundary Guarding Problem with Translating Targets," *Proceedings of the 48th IEEE Conference on Decision and Control held jointly with the 28th Chinese Control Conference CDC/CCC2009* (Dec. 2009) pp. 8543–8548.
90. S. L. Smith, M. Pavone, F. Bullo and E. Frazzoli, "Dynamic Vehicle Routing with Priority Classes of Stochastic Demands," *SIAM J. Control Optim.* **48**(5), 3224–3245 (2010).
91. J. Snape, J. van den Berg, S. J. Guy and D. Manocha, "The hybrid reciprocal velocity obstacle," *IEEE Trans. Robot.* **27**(4), 696–706 (2011).
92. D. Sofge, A. Schultz and K. DeJong, "Evolutionary Computational Approaches to Solving the Multiple Traveling Salesman Problem Using a Neighborhood Attractor Schema," *Proceedings of the Applications of Evolutionary Computing on EvoWorkshops 2002: EvoCOP, EvoIASP, EvoSTIM/EvoPLAN*, London, UK: Springer-Verlag (2002) pp. 153–162.
93. O. Takahashi and R. J. Schilling, "Motion planning in a plane using generalized voronoi diagrams," *IEEE Trans. Robot. Autom.* **5**(2), 143–150 (Apr. 1989).
94. Z. Tang and Ü. Özgüner, "Motion planning for multitarget surveillance with mobile sensor agents," *IEEE Trans. Robot.* **21**(5), 898–908 (Oct. 2005).
95. O. Tekdas, V. Isler, J. Lim and A. Terzis, "Using mobile robots to harvest data from sensor fields," *IEEE Wireless Commun.* **16**(1), 22–28 (Feb. 2009).
96. E. Todorov and W. Li, "A Generalized Iterative LQG Method for Locally-Optimal Feedback Control of Constrained Nonlinear Stochastic Systems," *Proceedings of the American Control Conference ACC*, vol. 1 (Jun. 2005) pp. 300–306.
97. P. Toth and D. Vigo, eds., *The Vehicle Routing Problem* (Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001).
98. C. A. Valle, A. S. da Cunha, W. M. Aioffi and G. R. Mateus, "Algorithms for Improving The Quality of Service in Wireless Sensor Networks with Multiple Mobile Sinks," *Proceedings of the 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems MSWiM2008*, New York, NY, USA: ACM (2008) pp. 239–243.
99. J. Van Den Berg, S. J. Guy, M. Lin and D. Manocha, "Optimal Reciprocal Collision Avoidance for Multi-Agent Navigation," *Proceedings of the IEEE International Conference on Robotics and Automation* (2010).
100. J. van den Berg, S. J. Guy, M. Lin and D. Manocha, "Reciprocal n-Body Collision Avoidance. **In**: *Robotics Research: The 14th International Symposium ISRR* (C. Pradalier, R. Siegwart and G. Hirzinger, eds.) (Springer, Berlin Heidelberg, 2011).
101. J. van den Berg, M. Lin and D. Manocha, "Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation," *Proceedings of the IEEE International Conference on Robotics and Automation* (2008) pp. 1928–1935.

102. P. Váňa, Path Planning for Non-Holonomic Vehicle in Surveillance Missions *Master's Thesis* (Czech Republic: Czech Technical University in Prague, 2015).
103. P. Vána and J. Faigl, "On the Dubins Traveling Salesman Problem with Neighborhoods," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS* (Sep. 2015) pp. 4029–4034.
104. D. Wilkie, J. Van den Berg and D. Manocha, "Generalized Velocity Obstacles," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2009) pp. 5573–5578.
105. F.-J. Wu, C.-F. Huang and Y.-C. Tseng, "Data Gathering by Mobile Mules in a Spatially Separated Wireless Sensor Network," *Proceedings of the 10th International Conference on Mobile Data Management: Systems, Services and Middleware MDM2009*, Washington, DC, USA: IEEE Computer Society (2009) pp. 293–298.
106. X. Yu and J. Y. Hung, "A Genetic Algorithm for the Dubins Traveling Salesman Problem," *Proceedings of the IEEE International Symposium on Industrial Electronics ISIE* (May 2012) pp. 1256–1261.
107. B. Yuan, M. Orlowska and S. Sadiq, "Finding the Optimal Path in 3D Spaces Using EDAs — The Wireless Sensor Networks Scenario," *Proceedings of the 8th International Conference on Adaptive and Natural Computing Algorithms, Part I ICANNGA2007* (Springer-Verlag, Berlin, Heidelberg, 2007) pp. 536–565.
108. B. Yuan, M. Orlowska and S. Sadiq, "On the optimal robot routing problem in wireless sensor networks," *IEEE Trans. Knowl. Data Eng.* **19**(9), 1252–1261 (Sep. 2007).