An Experimental Analysis on the Necessary and Sufficient Conditions for the RRT* Applied to Dynamical Systems

Israel Becerra(✉), Heikel Yervilla-Herrera, and Rafael Murrieta-Cid

Centro de Investigación en Matemáticas (CIMAT), Guanajuato, Mexico
{israelb,yervilla.herrera,murrieta}@cimat.mx

Abstract. In this paper, we study some properties of several local planners for nonholonomic dynamical systems to achieve asymptotic global optimality through the RRT*. More specifically, we study the conditions that a local steering method must have to produce global optimal trajectories in an environment with obstacles. The main properties we analyse in the steering methods are the following: (1) Whether or not the steering method produces local optimal motion primitives (optimal letters). (2) Whether or not the steering method concatenates the local optimal primitives in such a way that the resulting concatenation is also optimal (optimal words). (3) Whether or not the steering method produces trajectories that respect the topological property. Experimentally, it is studied how those properties affect the speed of convergence to the globally optimal solution, moreover, their sufficiency and necessity is also validated, all making use of the problem of finding the time-optimal trajectories for a differential drive robot in the presence of obstacles. We also discard conditions that show not to be necessary and we give some insight on the necessary and sufficient conditions for the RRT* to asymptotically converge to optimal trajectories, which is indeed the sough research target.

Keywords: RRT* · Dynamical systems · Optimality · Nonholonomy · Steering methods · Motion planning

1 Introduction

Motion Planning has presented great advances since the early proposed algorithms where the generation of exact solutions was the preferred approach [3], passing through cellular decompositions, graphs searches, potential fields, and sampling-based methods [10]. In particular, the sampling-based methods have been successful in finding collision-free trajectories for high dimensional spaces and have seen a constant evolution from the early PRM [9], which is one of the first tools to construct road-maps using a sampling-based approach, to the

The authors would like to acknowledge the financial support of Intel Corporation.

M. Morales et al. (Eds.): WAFR 2018, SPAR 14, pp. 835–851, 2020.
https://doi.org/10.1007/978-3-030-44051-0_48

RRT [13], which introduces interesting concepts such as sampling the controls space to address the kinodynamic planning problem. A lot of work has been done regarding Kinodynamic and nonholonomic problems [2,5,11–13], particularly in sampling-based planning methods. In the present work we study this particular type of planning methods.

Since the introduction of the PRM and the RRT, one of the major breakthroughs was the introduction of the PRM*, RRT* and related approaches [7], which provided tools that are capable of achieving asymptotic optimality. Specially, the RRT based algorithms naturally extend to deal with nonholonomic constraints, which has given them a large amount of attention in the last years. In [15] a recent survey of such algorithms can be found. Nonetheless, the RRT based algorithms rely on the availability of a local steering method that corresponds to solve a two point boundary value problem (BVP), which by no means is an easy task for many interesting dynamical systems. Some efforts have been done to avoid this issue, such as in [14], where the authors propose an incremental sampling-based planner, the Stable Sparse RRT (SST), which does not rely on the availability of a local steering method. However, when a local steering method is available, for certain scenarios, the RRT* can produce higher convergence rates than the SST, as it is shown in [14] experiments. Hence, the interest in RRT* based algorithms is still present.

Effort has been put into extending the RRT* to deal with a variety of kinodynamic motion planning problems. For instance, in [18] the authors extend the RRT* to consider linear differential constrains by using a fixed-final-state-free-final-time controller that connects any pair of states, optimizing a cost function that includes a trade-off between the duration of a trajectory and the expended control effort. The non-linear dynamics are considered using first-order Taylor approximations. Similarly, in [16], a linearisation of the system dynamics is performed to later apply a linear-quadratic regulator (LQR) controller within the RRT*, finding in that manner optimal plans for several complex or underactuated systems. Nonetheless, a richer generalization to extend the RRT* method to deal with a broader range of problems with differential constraints is still of interest, moreover, a crucial question that remains unanswered is related to the properties that the local planners must posses.

An attempt to achieve such generalization is presented in [6,8]. In contrast to other works, the authors of [6,8] tried to present a set of conditions to be able to apply the RRT* to nonholonomic planning problems. The conditions presented in [6] are proved to be sufficient, while the conditions presented in [8] are merely suggested without a formal proof of whether they are necessary or sufficient. The main requirement presented in [6] is the availability of a local optimal planner that will induce an optimal distance function, and which directly impacts the steering and rewire procedures. Apart from the required local planner, the authors proposed sufficient conditions in the form of a local controllability property, referred as weak local controllability, and a second sufficient condition related to the existence of a solution to the problem. According to the authors, fulfilling the aforementioned conditions guarantees the RRT* asymptotic optimality. In [8], the authors relax some requirements such as the optimality of the local planner, but introduce some new conditions, namely, the topological property [17] related to the local planner, and the system to be small time locally attainable.

The main contributions in the present work can be summarized as follows:

- We present a concise summary of the conditions presented in [6,8] to achieve asymptotic optimality while using the RRT* in the context of kinodynamic planning problems, and experimentally evaluate the validity of the conditions that were merely suggested without formal proof.
- A compilation of different methodologies based on the summarized conditions, which combine local steering methods with the RRT*, is presented.
- An experimental analysis to compare the performance of the considered methods. For such comparison the case of time-optimal trajectories for a differential drive robot (DDR) in the presence of obstacles is addressed.
- We exhibit the existence of cases where a local planner that does not generate optimal motion primitives (optimal letters) does not converge to the optimal cost, indeed, it does not even converge to a finite cost.
- Experimentally we found that the *topological property* is not a necessary condition in the current context, noting that the problem of achieving asymptotically global optimality with the RRT* is a different problem from approximating any geometric path through paths computed with a local planner that respects the nonholonomic constraints [12].
- Making use of a counterexample approach, we give some insight on the necessary and sufficient conditions for the RRT* to asymptotically converge to optimal trajectories, which sets possible directions for future research.

The remainder of this paper is organized as follows. Section 2 presents the problem definition. Section 3 summarizes the conditions in [6,8]. Section 4 shows the compilation of methodologies based on the summarized conditions. Section 5 describes the methodologies in the context of the problem of time-optimal trajectories for a DDR. Section 6 shows the performance experimental analysis. Section 7 presents a discussion with our analysis on the sufficiency and necessity of the studied conditions.

2 Problem Definition

Let X and U be smooth manifolds that represent the state and control spaces, respectively, and consider the following dynamical system:

$$\dot{x}(t) = f(x(t), u(t)), x(0) = x_0, \tag{1}$$

where $x(t) \in X$, $u(t) \in U$, for all t, f is Lipschitz continuous and $x_0 \in X$. Let $X_{free} \subset X$ be the set of collision free states, $X_{goal} \subset X$ the goal set, and $c : X \to \mathbb{R}_{\geq 0}$ the cost function. In the present work we consider the *optimal kinodynamic motion planning problem* that is stated as follows:

Definition 1 (Optimal Kinodynamic Motion Planning Problem). *Find a dynamically-feasible trajectory* $x : [0,T] \rightarrow X$*, with* $x(0) = x_0$*, that (i) is collision-free, i.e.* $x(t) \in X_{free}$*,* $\forall t$*, (ii) reaches the goal region, i.e.* $x(T) \in X_{goal}$*, (iii) minimizes the cost functional* $J(x) = \int_0^T c(x(t))dx$*.*

We will also consider that to solve the *optimal kinodynamic motion planning problem*, the RRT* methodology is being used. See [7] for further details.

3 Conditions for Asymptotic Convergence of the RRT*

In this section we present the conditions introduced in [6,8]. We summarize them keeping only the conditions that are not redundant (nor dominated by any other condition), unless it is convenient to preserve them to gain flexibility on the available catalogue of conditions that will be chosen to be fulfilled. The first condition in [6] is the availability of an optimal local planner (local steering method), which returns the optimal trajectory connecting any two states, z_1, $z_2 \in X$, in the absence of obstacles, and which is formally defined as follows:

Definition 2. *A local planner* ℓ *is an* Optimal Local Planner*, if there exists an* $\epsilon > 0$*, and* ℓ *returns a trajectory* $x^* : [0,T] \rightarrow X$*, with* $x^*(0) = z_1$*, and* $x^*(T) = z_2$*, driven by the input* $u^* : [0,T] \rightarrow U$ *fulfilling* $\dot{x}^* = f(x^*(t), u^*(t))$ $\forall t \in [0,T]$*, s.t.* $J(x^*) = \min_{T \in \mathbb{R}_{\geq 0}, u} J(x)$*,* $\forall \parallel z_1 - z_2 \parallel \leq \epsilon$*.*

The second sufficient condition in [6] is the that the considered dynamical system respects the so called *weak local controllability* (WLC) property. For stating that condition consider the next concepts. Denote $\mathcal{B}_\epsilon(z)$ the closed ball centred at state z, and let $\mathcal{X}_{z,z'}$ denote the set of all trajectories that start from state z and reach state z' respecting the state transition equation of the dynamical system. Given a state $z \in X$ and a constant $\epsilon > 0$, let $\mathcal{R}_\epsilon(z) = \{z' \in X | \exists x \in \mathcal{X}_{z,z'},\ s.t.\ x(t) \in \mathcal{B}_\epsilon(z)\ \forall t \in [0,T]\}$, see [6] for more details. Then, the *weak local controllability* (WLC) is given by the next definition:

Definition 3 (Weak Local Controllability (WLC) [6]). *There exist constants* α*,* $\bar{\epsilon} \in \mathbb{R}_{>0}$*,* $p \in \mathbb{N}$*, such that for any* $\epsilon \in (0, \bar{\epsilon})$*, and any state* $z \in X$*, the set* $\mathcal{R}_\epsilon(z)$ *of all states that can be reached from* z *with a path that lies entirely inside the* ϵ*-ball centred at* z*, contains a ball of radius* $\alpha\epsilon^p$*.*

The third sufficient condition in [6] is an assumption on the obstacle region to ensure that there exists an optimal trajectory with enough free space around it to allow almost-sure convergence. For the sake of brevity we refer the reader to [6] for details.

In [8], the authors present an enhancement of the computational effectiveness of the RRT* algorithm. This is achieved by modifying the NearVertices procedure such that the neighbours are computed considering a weighted Euclidean box, which resembles the shape of the sub-Riemannian ball that respects the nonholonomic constrains of the dynamical system. This modification requires the dynamical system to be *small time locally attainable* (STLA), which gives rise to the first condition presented in [8]. Consider a given state $z \in X$ and a real number $t > 0$, the *small-time attainable set* $\mathcal{A}(z,t)$, is the set of all reachable states within time t by the considered dynamical system, starting from state z. Then, the *small time locally attainable* (STLA) property is given by next definition:

Definition 4. *A system is said to be* Small Time Locally Attainable (STLA) *at state $z \in X$, if $\mathcal{A}(z,t)$ has a non empty interior for all $t > 0$.*

The second condition shown in [8] is related to the local planner used in the RRT*. Such condition is referred as the *topological property*, which was first introduced in [17]; broadly speaking, it states that the trajectory that joins any two states that are within a ball of radius η, will not leave a ball of radius ϵ. The *topological property* is formally stated as follows:

Definition 5. *A local planner ℓ that drives the system from a state z to a state z' with a trajectory $x \in \mathcal{X}_{z,z'}$, with $z = x(0)$ and $z' = x(T)$, respects the* Topological Property (TP) *if*

$$\forall \epsilon > 0, \exists \eta > 0 \ s.t. \ \forall z \in X, \ \forall z' \in \mathcal{B}_\eta(z), \forall t \in [0,T], \ x(t) \in \mathcal{B}_\epsilon(z).$$

In [17], the *topological property (TP)* was introduced under the assumption that the dynamical system is *small time locally controllable* (STLC). This implicitly suggests that property as a third condition assumed in [8], which would override the STLA property, since it is well known that a system that is STLC is STLA, but not the contrary. Such property is given by the next definition:

Definition 6. *A system is said to be* Small Time Locally Controllable (STLC) *at state $z \in X$, if $\mathcal{A}(z,t)$ has a neighbourhood of z for all $t > 0$.*

As stated by [6], a system that is locally controllable in the sense of [4], which is referred in [17] as *small-space locally controllable*[1] (SSLC), also fulfils the WLC property. However, since a SSLC can be interpreted as the STLC condition, a system that is STLC would also respect the WLC condition. Therefore, joining the facts that a system that is STLC is WLC, and that the STLA property is not sufficient to guarantee the local planner *topological property*, we will keep the STLC condition over the WLC and STLA properties.

Summarizing, the conditions presented in [6,8] are:

I The considered dynamical system is STLC.
II The local planner used in the RRT* is an optimal local planner.
III The local planner used in the RRT* respects the topological property.
IV There exist an optimal path which has enough obstacle-free space around it to allow almost-sure convergence.

[1] A system is *small-space locally controllable at $z \in X$*, if for any neighbourhood Ω of z, there exists a neighbourhood $A_\Omega(x)$ whose points are all accessible by the system without departing from Ω. The system is *small-space locally controllable* if it is SSLC at any $z \in X$.

Notice that an optimal local planner satisfies the *topological property* [17], but a local planner that satisfies the *topological property* is not necessarily optimal, indeed, in Sect. 5.3 we present such a planner. As stated in [17], it is not an easy task to obtain an optimal local planner, nor to design a local planner that respects the topological property, hence, we will keep both properties to preserve the alternative of either satisfying one condition or the other one depending on the available local planner.

4 Methods Overview for Using the RRT* for Nonholonomic Dynamical Systems

The different methodologies we present in this section are derived based on the characteristics of the local planer ℓ used in the RRT*. We will generate different combinations depending on the optimal letters or optimal words used by the planner, or whether ℓ fulfils or not the *topological property*. We refer as the optimal letters to the motion primitives returned by an optimal planner, and the optimal words, to the concatenation of motion primitives that compose any optimal trajectory. For instance, for the time-optimal trajectories for a DDR in the absence of obstacles [1], the optimal letters–motion primitives–, are either rotations in site (clockwise rotation $\curvearrowright$ or counter-clockwise rotations $\curvearrowleft$) or straight line motions (forward motion $\Uparrow$ or backward motion $\Downarrow$), and the structure of the optimal words are one of the shown in Table 1, or a subsection of them.

Table 1. Structure of optimal words

Tangent	$\curvearrowright\Uparrow\curvearrowright$	$\curvearrowright\Downarrow\curvearrowright$	$\curvearrowleft\Uparrow\curvearrowleft$	$\curvearrowleft\Downarrow\curvearrowleft$
Tangent$_\pi$	$\Uparrow\curvearrowright_\pi\Downarrow$	$\Downarrow\curvearrowright_\pi\Uparrow$	$\Uparrow\curvearrowleft_\pi\Downarrow$	$\Downarrow\curvearrowleft_\pi\Uparrow$
ZigZag	$\Uparrow\curvearrowleft\Downarrow\curvearrowright\Uparrow$	$\Downarrow\curvearrowleft\Uparrow\curvearrowright\Downarrow$	$\Uparrow\curvearrowright\Downarrow\curvearrowleft\Uparrow$	$\Downarrow\curvearrowright\Uparrow\curvearrowleft\Downarrow$

Table 2 shows the different local planners to be considered in the rest of the paper. We use the superscript (+) to denote that a planner respects the respective property, or (-) if it does not. The superscript (*) means that the planner partially respects the mentioned property.

Table 2. Local planners

	Optimal **L**etters	Optimal **W**ords	**T**opological Property
$\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$	YES	YES	YES
$\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$	YES	Not all	YES
$\mathbf{L}^+\mathbf{W}^-\mathbf{T}^+$	YES	NO	YES
$\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$	YES	NO	NO
$\mathbf{L}^-\mathbf{W}^-\mathbf{T}^-$	NO	NO	NO

5 Description of Local Planners: The Case of Time-Optimal Planning for a DDR

In this section, we further describe the local planners to be used in the context of the problem of obtaining time-optimal trajectories for a DDR in the presence of obstacles. We describe each of the planners and prove their claimed properties. These planners are the ones that will be used in the experimental analysis shown in Sect. 6. The kinematics of a DDR are given by the next set of equations:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} w, \tag{2}$$

where $v = \frac{1}{2}(u_1 + u_2)$ is the robot's linear speed, $w = \frac{1}{2b}(u_1 - u_2)$ is the robot's angular speed, and $2b$ is the width of the DDR. The left and right wheels angular speeds are u_1 and u_2, respectively, which are the system controls. In the following sections a DDR will be used as our dynamical system, which is STLC [12], fulfilling in that way Condition I from Sect. 3. We will also assume in the next sections that Condition IV is respected.

5.1 Local Planner $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$

This planner corresponds to the planner from [1] that solves the time-optimal planning problem for a DDR in the absence of obstacles. The authors of [1] present an algorithm that returns an optimal trajectory (and its cost) within nine possible symmetry classes of optimal trajectories (indexed as A, B, C, ..., H, I), connecting in that way any two states $z_1 \in X$ and $z_2 \in X$. Note that this planner is not local, conversely, it contains the whole dictionary of optimal words to cover the whole state space X. Such planner does not only respects optimality but also respects the *topological property* as it is stated in Proposition 1.

Proposition 1. *Local planner* $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ *respects the topological property.*

Proof. The local planner $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ returns optimal trajectories for the whole state space X according to [1]. Local planners that are based on families of optimal trajectories satisfy the topological property [17].

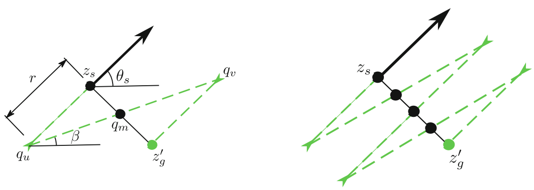
Corollary 1. *A local planner that considers a subset of the optimal words of the dictionary from [1], for a neighbourhood of any* $z \in X$*, respects the topological property.*

5.2 Local Planner $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$

Planner $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ is also based on the optimal planner presented in [1]. From the nine possible families of optimal trajectories, planner $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ sticks to only trajectories from symmetry classes D and G, which are able to reach a neighbourhood of any state $z \in X$. Class D corresponds to a structure of the form $\curvearrowright\Uparrow\curvearrowright$, which encloses seven different symmetric trajectories. Class G is of the structure $\Downarrow\curvearrowright\Uparrow$, containing its respective seven symmetric trajectories. Planner $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ is a local version of planner $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$, it contains some of the optimal words of the whole dictionary, moreover, it respects the topological property by Corollary 1. If planner $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ is used in the RRT* construction, only samples that can be optimally connected to the tree are included in the tree, that is, if the optimal trajectory to connect a given sample to the tree does not correspond to class D or class G, then the sample is discarded.

5.3 Local Planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$

This local planner alternates straight line motions and rotations in site yielding sort of a 'zigzag' behaviour, hence, note that it makes use of optimal letters. Given two states $z_s = (x_s, y_s, \theta_s) \in X$ and $z_g = (x_g, y_g, \theta_g) \in X$, the planner proceed as follows (see Fig. 1). First, we generate a intermediate goal $z'_g = (x_g, y_g, \theta_s)$, that is, we build z'_g such that it has the x and y coordinates of z_g, but z'_g has the orientation of z_s. Second, consider the line l between states z_s and z'_g, projected on $\mathbb{R}^2$. The DDR will first move backward (motion primitive $\Downarrow$) a distance r, then rotates in site to align with a direction β, then moves forward a distance $d = \frac{2r}{sin\beta}$, then rotates in site, moves backward, etc., and keeps repeating this. Following such behaviour the robot will visit m intermediate points over l, until z'_g is reached. Third, the planner makes the robot to rotate in site from z'_g to finally align with z_g.



(a) Trajectory with one intermediate point ($m = 1$).

(b) Trajectory with three intermediate points ($m = 3$).

Fig. 1. Trajectories (dashed lines) computed by $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ to reach a state z'_g from state z_s.

To compute the r, β and m parameters, we make use of Algorithm 1. This algorithm makes sure that planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ respects the *topological property*, keeping the trajectories inside a ball $\mathcal{B}_{2r}(z_s)$ by iteratively increasing the number of reference intermediate points over l, until the containment is achieved. Through Proposition 2 we formally prove that the local planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ respects the *topological property*.

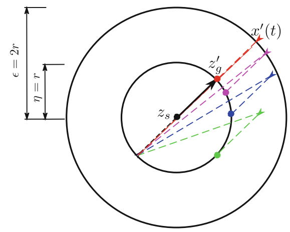


Fig. 2. The starting state z_s is shown at the center of balls $\mathcal{B}_\epsilon(z_s)$ and $\mathcal{B}_\eta(z_s)$. Different trajectories are shown as dashed lines, corresponding to trajectories that planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ might produce to take the robot from z_s to the respective goals shown as coloured points over the inner circle. The goal z'_g is the state goal whose trajectory contains the farthest point $x'(t)$.

Proposition 2. *Local planner* $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ *respects the topological property.*

Proof. Given a start state $z_s = (x_s, y_s, \theta_s) \in X$ and a goal state $z_g = (x_g, y_g, \theta_g) \in X$, Algorithm 1 computes the parameter $r = \sqrt{(x_s - x_g)^2 + (y_s - y_g)^2 + (\theta_s - \theta_g)^2}$, which can be used to define a ball $\mathcal{B}_r(z_s)$ around the start state z_s. Later, algorithm 1 computes the parameters β and m, such that by construction, the trajectory x delivered by the local planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$, with $x(0) = z_s$ and $x(T) = z_g$, does not leave a ball $\mathcal{B}_{2r}(z_s)$, that is, $x(t) \in \mathcal{B}_{2r}(z_s)$, $\forall t \in [0, T]$. Moreover, given any potential goal state $z'_g \in \mathcal{B}_r(z_s)$, the trajectory x' generated by $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$, with $x'(0) = z_s$ and $x'(T') = z'_g$, respects $x'(t) \in \mathcal{B}_{2r}(z_s)$, $\forall t \in [0, T']$. The goal state z'_g whose trajectory contains the farthest state $x'(t)$, is $z'_g = (x_s + r\cos(\theta_s), y_s + r\sin(\theta_s), \theta_s)$, which actually touches the periphery of $\mathcal{B}_{2r}(z_s)$ at $(x_s + 2r\cos(\theta_s), y_s + 2r\sin(\theta_s), \theta_s)$ (see Fig. 2). Setting $\epsilon = 2r$, and $\eta = r$, yields $\epsilon = 2\eta$, meaning that to any ϵ corresponds a unique η, independently of z_s, hence, the local planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ respects the *topological property*.

Algorithm 1. Local planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ (β, m computation)

Input: $z_s = (x_s, y_s, \theta_s)$ and $z_g = (x_g, y_g, \theta_g)$
Output: β and m

```
r ← √((x_s − x_g)² + (y_s − y_g)² + (θ_s − θ_g)²)
û ← (cos(θ_s), sin(θ_s), 0) //Unit vector in direction θ_s
q_u ← z_s − r ∗ û //Intermediate state (see Fig. 1a), q_u = (x_u, y_u, θ_u)
q_m ← ((x_s+x_g)/2, (y_s+y_g)/2, θ_s) //Intermediate state (see Fig. 1a), q_m = (x_m, y_m, θ_m)
m ← 1        //Total number of intermediate points
q_v ← z_g + r ∗ û //Intermediate state (see Fig. 1a), q_v = (x_v, y_v, θ_v)
Loop:
β ← angle to align the DDR from q_u toward q_m
if √((x_v − x_s)² + (y_v − y_s)² + (β − θ_s)²) > 2 ∗ r    OR
√((x_u − x_s)² + (y_u − y_s)² + (β − θ_s)²) > 2 ∗ r then
    q_u ← q_m − r ∗ û
    q_m ← ((x_m+x_g)/2, (y_m+y_g)/2, θ_s)
    m ← 2 ∗ m + 1
    goto Loop.
else
    return β, m
end if
```

5.4 Local Planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$

The present local planner consists of a rotation in site ($\curvearrowleft/\curvearrowright$), a straight line motion ($\Uparrow/\Downarrow$), and a rotation in site ($\curvearrowleft/\curvearrowright$). This corresponds to the symmetry classes D and F from [1], hence, this planner makes use of optimal letters. The trajectories that the $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ planner throws are used to connect any pair of given states, z_s and z_g (given that the trajectory is collision free), even if such trajectory is not the optimal one, therefore, the planner will be considered to not be using optimal words. It is worth to mention that despite the fact that this planner is able to connect any pair of states, the planner does not respect the *topological property*, as it is proved in Proposition 3. Such statement is proved by giving a family of pathological goal states such that, no manner how close those states are from a start state z_s, there is no arbitrary small ϵ whose associated ball $\mathcal{B}_\epsilon(z_s)$ will contain their related trajectories.

Proposition 3. *Local planner* $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ *does not respect the topological property.*

Proof. Consider a pair of states $z_s = (x_s, y_s, \frac{\pi}{4})$ and $\bar{z} = (\bar{x}, \bar{y}, \frac{\pi}{4})$, with $\bar{x} > x_s$ and $\bar{y} = y_s$. Also note that both have the same orientation $\theta = \frac{\pi}{4}$. In order to move from z_s towards $\bar{z}$, planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ would return the next sequence of motion primitives: $\curvearrowright\Uparrow\curvearrowleft$. Now consider the ball $\mathcal{B}_\epsilon(z_s)$, and consider the ray ρ that goes from z_s and infinitely extends toward $\bar{z}$. The trajectory delivered by $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ to move from z_s to any state $z' \in \rho$ would require to rotate a total angle of $\frac{\pi}{2}$ radians, no matter how close z' is to z_s. Any ball $\mathcal{B}_\eta(z_s)$ inside $\mathcal{B}_\epsilon(z_s)$ would still contain states over ρ, therefore, as $\epsilon \to 0$, the trajectories corresponding to goal states $z' \in \rho$ inside $\mathcal{B}_\eta(z_s)$ would eventually be out of $\mathcal{B}_\epsilon(z_s)$, due to the always required total rotation of $\frac{\pi}{2}$ radians. Therefore, $\exists$ an ϵ with no η s.t. $\forall z \in X$, $\forall z' \in \mathcal{B}_\eta(z)$, $\forall t \in [0, T]$, $x(t) \in \mathcal{B}_\epsilon(z)$. The result follows.

5.5 Local Planner $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$

The proposed local planner consists of a concatenation of arcs of circles, so it clearly applies controls that do not correspond to optimal words, nor to optimal letters. The planner proceeds as follows. Consider a pair of states $z_s \in X$ and $z_g \in X$. Then consider the line segment l in $\mathbb{R}^2$ that joins z_s and z_g. Planner $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$ will try to connect the extreme points of l with an arc centred at l midpoint. The trajectory will consist in rotating in site until the DDR's orientation is aligned with the tangent to the arc, then the robot follows the arc, and finishes rotating in site to align with θ_g, all applying saturated controls. If by following such trajectory collision happens, then a refinement is attempted, introducing intermediate bias points and connecting them with smaller arcs as it shown in Fig. 3. This is repeated until N refinements are attempted. Notice that for any trajectory x, its corresponding projection $\hat{x}$ in $\mathbb{R}^2$ could be approximated by this procedure, and as the number of intermediate bias points tends to ∞, the arcs approximation resembles more and more the curve $\hat{x}$. Nonetheless, it is important to mention that as the number of bias points increase, the cost related to the arcs-approximation might not tend to $J(x)$. This is proved in Proposition 4.

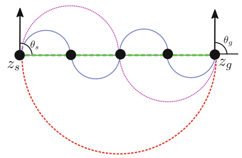


Fig. 3. The green segment is the projection $\hat{x}$ of a trajectory x into $\mathbb{R}^2$, which is the trajectory sought to be approximated by the arc-type trajectories. As the intermediate bias points is increased, the arc-type trajectories tend to $\hat{x}$.

Proposition 4. *Given a pair of states $z_s \in X$ and $z_g \in X$, and a trajectory x, with $x(0) = z_s$ and $x(T) = z_g$, let $\Delta_n(x)$ denote an arcs-approximation trajectory that considers n intermediate bias points over trajectory x. There exist cases where $\lim_{n\to\infty} J(\Delta_n(x)) \neq J(x)$.*

Proof. Consider two states $z_s = (0, 0, \frac{\pi}{2}) \in X$ and $z_g = (2, 0, \frac{\pi}{2}) \in X$, and consider a trajectory x, with $z_s = x(0)$ and $z_g = x(T)$, which rotates in site, moves with a straight line, and rotates in site, with saturated wheels speed, just as shown in Fig. 3. Assuming the robot's radius $b = 1$, the cost (elapsed time) related to such trajectory would be $J(x) = 2 + \pi$[2] (assuming a unit valued maximum speed). Considering an arcs-approximation $\Delta_0(x)$, the time that will take the robot to travel the arc would be $\pi + \frac{2\pi * 1}{2} * 2 = 3\pi$, yielding $J(\Delta_0(x)) = 3\pi$. For a trajectory $\Delta_1(x)$, $J(\Delta_1(x)) = (\frac{2\pi * 0.5}{2} + \frac{2\pi * 0.5}{2}) * 3 = 3\pi$, which is

2 The related cost is computed as $t = s(t) + b\sigma(t)$, where $s(t)$ is the rectified path length in $\mathbb{R}^2$, the plane of robot position, and $\sigma(t)$ is the rectified arc length in S^1, the circle of robot orientations, see [1] for details.

computed multiplying the sum of arc lengths by $1/v$. For a trajectory $\Delta_3(x)$, $J(\Delta_3(x)) = (\frac{2\pi*0.25}{2} + \frac{2\pi*0.25}{2} + \frac{2\pi*0.25}{2} + \frac{2\pi*0.25}{2})*5 = 5\pi$, etc. Then, $J(\Delta_n(x)) = (n+2)\pi$ for $n \geq 2^i - 1$, $i = 1, 2, \ldots$, therefore, $\lim_{n\to\infty} J(\Delta_n(x)) = \infty$, which is clearly different from $J(x) = 2 + \pi$. The result follows.

Remark 1. In the scenario described in Proposition's 4 proof, the length of the different trajectories $\Delta_n(x)$ have the same rectified length, but the time that takes the robot to travel them tends to infinity because the DDR linear velocity $v \to 0$ as $n \to \infty$.

Considering the pair of states $z_s = (x_s, y_s, 0)$ and $\bar{z} = (\bar{x}, \bar{y}, 0)$, with $\bar{x} > x_s$ and $\bar{y} = y_s$, using similar arguments to the ones presented in Proposition 3, it can also can be proven that $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$ does not respect the *topological property.* Finally, since local planner $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$ might yield trajectories $\Delta_n(x)$ whose cost does not tend to the cost of the related trajectory x, that is $\lim_{n\to\infty} J(\Delta_n(x)) \neq J(x)$, we will not consider $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$ in the experimental analysis presented in the next section, however, we further developed $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$ in this section because it will be relevant in Sect. 7, where we present our discussion and conclusions.

6 Experimental Analysis of the Convergence: The Case of Time-Optimal Planning for a DDR

In this section, we present two experiments where we compute time optimal trajectories for a DDR in the two environments shown in Fig. 4. The used local planners used in the comparison are $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$ (the global planner proposed in [1]), $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ (a local version of the planner proposed in [1]), $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ (the zigzag planner that respects the *topological property*), and $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ (the planner that rotates in site, moves in straight line motion, and rotates in site).

6.1 Experiment 1

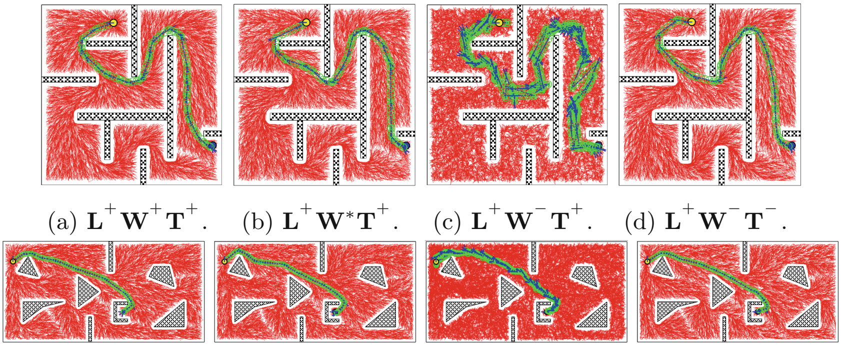
This experiment considers several pairs of star and goal states within the environment. For statistical purposes, the results that we present in this section correspond to the averages over trajectories resulting from 10 pairs of start-goal states. More precisely, for each one of the start-goal pairs, we compute a trajectory using each one of the four local planners, and then we run statistics over the 40 resulting trajectories (4 local planners and 10 trajectories for each). In each case the trees' generation was stopped when 20000 nodes were successfully generated. Table 3 summarizes the results. Planner $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$ achieved the smallest average cost in both environments, followed by $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ in Environment 1, and by $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ in Environment 2. Planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ presented an average cost way above from the other planners. Regarding average planning time, planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ showed the fastest times, followed by $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$, then $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$, and finally $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$.

Table 3. Average total planning time and total cost for Experiment 1.

	Environment 1		Environment 2	
	Plan. Time (sec)	Tot. Cost	Plan. Time (sec)	Tot. Cost
$\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$	324.1677	147.9497	322.3515	153.1534
$\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$	262.3257	151.2736	262.1216	158.8366
$\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$	631.305	819.5697	605.4679	813.8562
$\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$	193.3317	151.98	193.6325	154.4702

6.2 Experiment 2

In this experiment we set a single pair of start-goal states and we compute 10 trajectories using each one of the local planners. The presented results correspond to averages over those sets of 10 trajectories. Again, in each case the trees' generation was stopped when 20000 nodes were successfully generated. Figure 4 shows four sample trajectories that where generated with each one of the local planners. It can be seen that the resulting trajectories are similar, with the exception of the one obtained with planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$, nonetheless, the four trajectories belong to the same homotopy class.



(a) $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$. (b) $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$. (c) $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$. (d) $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$.

(e) $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$. (f) $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$. (g) $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$. (h) $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$.

Fig. 4. Trajectories generated using the four different local planners for given start and goal states.

As expected, see Table 4, for each local planner the planning times are quite similar to the ones presented in Experiment 1. This is because the planning time is mainly affected by the number of nodes, which were set to 20000 per tree in both of the experiments. Regarding the trajectories cost, planners $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$, $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$ and $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ obtained similar resulting costs, conversely to planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ that has a quite larger cost. The best cost comes from $\mathbf{L}^{+}\mathbf{W}^{+}\mathbf{T}^{+}$, followed by $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$, and $\mathbf{L}^{+}\mathbf{W}^{*}\mathbf{T}^{+}$. Figure 5 shows the evolution of the average

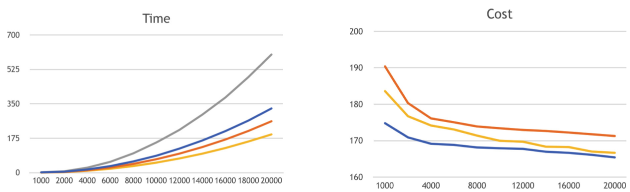
Table 4. Average total planning time and total cost for Experiment 2.

	Environment 1		Environment 2	
	Plan. Time (sec)	Tot. Cost	Plan. Time (sec)	Tot. Cost
$\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$	325.3999	194.4113	326.4784	165.4334
$\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$	265.6703	198.1895	261.3032	171.299
$\mathbf{L}^+\mathbf{W}^-\mathbf{T}^+$	637.5478	990.7503	600.3666	867.1575
$\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$	194.9445	196.5391	194.37	166.6844

accumulated planning times and running costs as a function of number of nodes for Environment 2. Similar tendencies were shown in Environment 1. Costs of planner $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^+$ were not shown as they are huge compared to the other three.

7 Discussion and Conclusions

Based on the statistics obtained in Sect. 6, differences in performance are evident between the different planners. The local planner that achieved the best cost for a fixed number of nodes was the planner $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$, which considered the whole dictionary of optimal words. This is reasonable as that planner delivers optimal trajectories that connect any two states $z \in X$ and $z' \in X$, given there is no obstacles. Nonetheless, note that the difference in performance was not large comparing it against planners $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ and $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$. Regarding the computation time, the best results came from planner $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$. This is because the same concatenation of motions primitives is always executed, without the necessity of applying a procedure that first discerns what type of trajectory is the optimal and then compute the proper parameters for the motion primitives. Surprisingly, $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ outperformed $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ most of the trials in terms of total cost.



(a) Average total time. (b) Average total cost.

Fig. 5. Average planning time and total costs evolutions as a function of the number of nodes for given start and goal states in Environment 2. $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ shown in blue, $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ shown in orange, $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^+$ shown in gray, and $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ shown in yellow.

Apart from the performance comparison, the past results give insight on the necessary and sufficient conditions to achieve asymptotic optimality for the RRT* in the context of the kinodynamic problem. Table 5 summarized a series of inferences that we deduce based on the experimental observations of whether a given local planner ℓ presented a convergence to the minimal cost or not. For such analysis the costs and trajectories yielded by local planner $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ are considered as a baseline. We believe such planner to be the must reliable, since it fulfils all the properties that were presented in [6,8]. In Table 5 each column represents a property to be satisfied by the local planner ℓ, and the rows refer to whether that property is a necessary condition or a sufficient condition. We mark a table entry with * when there is evidence from the previous section suggesting that the shown label (YES or NO) is correct, but further analysis is required to confirm the conjecture. The difference between the *Optimal Words*, *Local Optimal Words* and *Subset Optimal Words* properties (the 3 of them use optimal letters), is that the first one considers that ℓ contemplates a complete dictionary of optimal words that optimally connects (given no obstacles) any pair of states $z \in X$ and $z' \in X$. The second one considers that ℓ has a subset of the optimal words dictionary, such that given a state $z \in X$, there exist a neighbourhood $\xi(z)$ around z, where ℓ yields an optimal trajectory to move from z to *any* $z' \in \xi(z)$. The third one, ℓ has a subset of the optimal words that allows to optimally traverse from a state $z \in X$ to *only some* $z' \in \xi(z)$.

Table 5. Necessary and sufficient conditions

	Optimal **L**etters	Optimal **W**ords	Local Optimal **W**ords	Subset Optimal **W**ords	**T**opological Property
Necessary	YES*	NO	NO	YES*	NO
Sufficient	YES*	YES	YES	YES*	YES*

Starting with the *Optimal Letters property*, the four planners contained optimal letters but since the behaviour of planner $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^+$ can result from a slow convergence or no converge at all, the necessity of this property is set as a conjecture. Using the same reasoning we set the sufficiency of this condition as a conjecture. Considering the *optimal words property*, both planners $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ and $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ where able to converge to the optimal cost, so this property is not necessary. On the other hand, planner $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ tells us that such property is sufficient, moreover, planners $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ and $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ did converge with only a subset of the optimal words. Regarding the *local optimal words property*, planner $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ was able to converge to the optimal cost, so this property is no necessary. Planner $\mathbf{L}^+\mathbf{W}^*\mathbf{T}^+$ confirms that this property is sufficient, furthermore, this planner is applied and equal to $\mathbf{L}^+\mathbf{W}^+\mathbf{T}^+$ when two samples $z \in X$ and $z' \in X$ are close enough. Considering the *subset optimal words property*, planner $\mathbf{L}^+\mathbf{W}^-\mathbf{T}^-$ seems to converge to the optimal cost, however, we did not test the set of all possible planners ℓ, so the necessity of this property is a conjecture. Again, planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ did converge, however, this planner does not match the optimality conditions of a local steering method mentioned in [6], so we leave it as a conjecture that this property is sufficient.

Concerning the *topological property*, it was shown through planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ than convergence is achievable even if ℓ does not satisfy such property, hence, there is evidence that the *topological property* is not necessary. Two of the three planners that fulfil the topological property are evidently converging to the optimal cost, however, the behaviour of planner $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{+}$ makes us label the sufficiency of this property as a conjecture.

It is important to notice that it was observed that $\mathbf{L}^{+}\mathbf{W}^{-}\mathbf{T}^{-}$ does achieve convergence despite of not fulfilling the *topological property*, nor optimality for a local neighbourhood. However, this planner respects the system dynamics and is able to approximate any path of the DDR up to a resolution. It is then important to recall that planner $\mathbf{L}^{-}\mathbf{W}^{-}\mathbf{T}^{-}$, the fifth planner in Sect. 5, is also able to approximate any path of the DDR subject to its dynamics, but it is not able to converge to the optimal cost as the resolution of the approximation increases. Moreover, such planner does not contain optimal letters, which can be considered as evidence in favour of the necessity of a planner ℓ considering the optimal letters. Thus, the condition that a planner ℓ respects the system dynamics, that it is able to approximate any path of the system, and that the cost associated to the approximation converges to the actual cost of the optimal path x^*, might be relevant to be studied and might be part of the sought necessary and sufficient conditions for the RRT* to achieve asymptotic optimality in the kinodynamic problem. Also keep in mind that to approximate a geometric path that does not respect the system dynamics is different from approximating a path that does respect the system dynamics; the latter is the relevant case for the optimal kinodynamic problem. Let us conclude emphasizing that the main objective of this paper was to analyse local planners in the context of RRT* for dynamical systems. We have narrowed down the conditions that such planners must have and also provided candidate properties to be necessary (such as optimal letters) to achieve asymptotic optimality, yielding insight for further formal mathematical analysis.

References

1. Balkcom, D.J., Mason, M.T.: Time optimal trajectories for bounded velocity differential drive vehicles. Int. J. Rob. Res. **21**(3), 199–217 (2002)
2. Barraquand, J., Latombe, J.: Nonholonomic multibody mobile robots: controllability and motion planning in the presence of obstacles. Algorithmica **10**(2–4), 121–155 (1993)
3. Canny, J.: The Complexity of Robot Motion Planning. ACM Doctoral Dissertation Award. MIT Press, Cambridge (1988)
4. Hermann, R., Krener, A.: Nonlinear controllability and observability. IEEE Trans. Autom. Control **22**(5), 728–740 (1977)

5. Hsu, D., Kindel, R., Latombe, J.C., Rock, S.: Randomized kinodynamic motion planning with moving obstacles. Int. J. Rob. Res. **21**(3), 233–255 (2002). https://doi.org/10.1177/027836402320556421
6. Karaman, S., Frazzoli, E.: Optimal kinodynamic motion planning using incremental sampling-based methods. In: IEEE Conference on Decision and Control (2010)
7. Karaman, S., Frazzoli, E.: Sampling-based algorithms for optimal motion planning. Int. J. Rob. Res. **30**(7), 846–894 (2011)
8. Karaman, S., Frazzoli, E.: Sampling-based optimal motion planning for non-holonomic dynamical systems. In: 2013 IEEE International Conference on Robotics and Automation, pp. 5041–5047 (2013)
9. Kavraki, L.E., Svestka, P., Latombe, J.C., Overmars, M.H.: Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Trans. Robot. Autom. **12**(4), 566–580 (1996). https://doi.org/10.1109/70.508439
10. Latombe, J.C.: Robot Motion Planning. Kluwer Academic Publishers, Norwell (1991)
11. Laumond, J.P., Jacobs, P.E., Taix, M., Murray, R.M.: A motion planner for non-holonomic mobile robots. IEEE Trans. Robot. Autom. **10**(5), 577–593 (1994)
12. Laumond, J.P., Sekhavat, S., Lamiraux, F.: Guidelines in nonholonomic motion planning for mobile robots. In: Laumond, J.P. (ed.) Robot Motion Plannning and Control, pp. 1–53. Springer, Heidelberg (1998)
13. LaValle, S.M., Kuffner, J.J.: Randomized kinodynamic planning. Int. J. Rob. Res. **20**(5), 378–400 (2001)
14. Li, Y., Littlefield, Z., Bekris, K.E.: Asymptotically optimal sampling-based kinodynamic planning. Int. J. Rob. Res. **35**(5), 528–564 (2016)
15. Noreen, I., Khan, A., Habib, Z.: Optimal path planning using RRT* based approaches: a survey and future directions. Int. J. Adv. Comput. Sci. Appl. **7**(11) (2016). https://doi.org/10.14569/IJACSA.2016.071114
16. Perez, A., Platt, R., Konidaris, G., Kaelbling, L., Lozano-Perez, T.: LQR-RRT*: optimal sampling-based motion planning with automatically derived extension heuristics. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2537–2542 (2012)
17. Sekhavat, S., Laumond, J.P.: Topological property for collision-free nonholonomic motion planning: the case of sinusoidal inputs for chained form systems. IEEE Trans. Robot. Autom. **14**(5), 671–680 (1998)
18. Webb, D.J., van den Berg, J.: Kinodynamic RRT*: asymptotically optimal motion planning for robots with linear dynamics. In: 2013 IEEE International Conference on Robotics and Automation, pp. 5054–5061 (2013)