Hardness Results for Two-Dimensional Curvature-Constrained Motion Planning

David Kirkpatrick* Irina Kostitsyna† Valentin Polishchuk‡

Abstract

We revisit the problem of finding curvature-constrained paths in a polygonal domain with holes. We give a new proof that finding a *shortest* curvature-constrained path is NP-hard; our proof is substantially simpler, and makes fewer assumptions about the polygonal domain, than the earlier proof of [Reif and Wang, 1998]. We also prove that it is NP-hard to decide existence of a *simple* (i.e., non-self-intersecting) path.

1 Introduction

Understanding the feasibility and optimality of the motion of car-like robots in the presence of obstacles entails, among many things, an understanding of curvature-constrained paths between specified configurations in the plane, that avoid a given set of obstacles. The study of curvature-constrained path planning has a rich history that long predates and goes well beyond robot motion planning, for example the work of Markov [29] on the construction of railway segments.

1.1 Definitions

Let P be a polygonal domain with holes (forbidden regions, or obstacles) in $\mathbb{R}^2$. Let $\pi : [0, L] \mapsto P$ be a continuous differentiable path, parameterized by arc length, and denote by $\pi'(t)$ the derivative of π at t. Path π is said to be *curvature constrained* if, for some constant c, the *average curvature* of π on every interval $[t_1, t_2] \subseteq [0, L]$, namely $||\pi'(t_1) - \pi'(t_2)||/|t_1 - t_2|$, is bounded above by c. Intuitively, every point on such a path can be sandwiched between two tangent circles of radius $1/c$. We assume that $\pi(0) = s, \pi(L) = t$ are two given points in P, and $\pi'(0) = S, \pi'(L) = T$ are two given vectors; the pair (s, S) (resp., (t, T)) is called the *initial* (resp., *final*) *configuration* of π. The path π is *simple* if it has no self-intersections: $\pi(t_1) \neq \pi(t_2)$ for $t_1 \neq t_2$.

*Department of Computer Science, University of British Columbia, kirk@cs.ubc.ca. Supported by the Natural Sciences and Engineering Research Council of Canada.

†Department of Computer Science, State University of New York at Stony Brook, ikost@cs.stonybrook.edu

‡Helsinki Institute for Information Technology, Department of Computer Science, University of Helsinki, polishch@helsinki.fi. Supported by the Academy of Finland grant 138520.

Under suitable scaling we can assume that $c = 1$. Hereafter we will assume that all paths avoid the interior of obstacles, that is they remain within P; all such paths with average curvature bounded by 1 are referred to as *admissible* paths.

1.2 Background

A fundamental result in curvature-constrained motion planning, due to Dubins [15], states that in the absence of obstacles shortest admissible paths are one of two types: a (unit) circular arc followed by a line segment followed by another arc (CLC), or a sequence of three circular arcs (CCC)[1]. Variations and generalizations of the problem were studied in [8, 9, 11, 13, 14, 18, 28, 30, 31, 33, 34, 36, 37].

Dubins' characterization plays a fundamental role in establishing the existence as well as the optimality of curvature-constrained paths. Jacobs and Canny [22] showed that even in the presence of obstacles it suffices to restrict attention to paths of Dubins form between obstacle contacts and that if such a path exists then the shortest such path is well-defined. Fortune and Wilfong [19] give a super-exponential time algorithm for determining the existence of, but not actually constructing, such a path. Characterizing the intrinsic complexity of the existence problem for curvature-constrained paths is hampered by the fact that there are no known bounds on the minimum length or *intricacy* (number of elementary segments), expressed as a function of the description of the polygonal domain, of obstacle-avoiding paths in Dubins form. In a variety of restricted domains polynomial-time algorithms exist that construct shortest admissible paths [1, 2, 6].

The NP-hardness of computing a shortest admissible path amid polygonal obstacles was established by Reif and Wang [32]. This motivated a variety of approaches to approximating shortest admissible paths including [4, 5, 22, 35, 38–40]. It is known, for example, that shortest robust paths, shortest paths of bounded intricacy and minimum intricacy paths of bounded length all have polynomial-time approximations. The books [25, 26] are general references; for some very recent work on Dubins paths see [7, 12, 16, 17, 20, 21].

[1] In general, any of the C or L segments could have zero length.

1.3 Results

In Section 2 we present a new proof that finding a shortest admissible path from (s, S) to (t, T) is NP-hard. Our proof is considerably simpler than the one given in [32]. In addition, our construction is more "robust" in that it applies even in non-degenerate polygonal domains, specifically domains that have no "pinhole" gaps between obstacles.

Our hardness proof in Section 2 depends critically on the fact that admissible paths can self-intersect. This leaves open the possibility that the problem of finding a shortest simple admissible path could be solved in polynomial time. (Note that the simplicity constraint is relevant in many applications; e.g. laying out a conveyor belt or designing a pipeline.) In Section 3 we show that this is not the case: even deciding the *existence* of an admissible path is NP-hard, if we restrict attention to simple paths (which, of course, implies that finding any approximation to the shortest such path is NP-hard).

2 NP-hardness of determining shortest curvature-constrained paths

Our NP-hardness proof involves a reduction from 4CNF-satisfiability. Specifically, suppose that Φ is a formula in 4CNF involving m clauses and k variables $X_0, \ldots, X_{k-1}$. We show how to construct a polygonal environment E, whose description is bounded in size by some polynomial in k, together with configurations (s, S) and (t, T) and a distance D, such that there exists an admissible path from (s, S) to (t, T) whose length is at most D if and only if Φ is satisfiable.

Our proof, like that of Reif and Wang, uses the idea of path-encoding, introduced by Canny and Reif [10] in their proof that determining the shortest obstacle-avoiding path, with no constraint on curvature, joining specified points in $\mathbb{R}^3$, is NP-hard. The fact that our problem is set in $\mathbb{R}^2$ makes it difficult to adapt the Canny-Reif approach directly (further evidenced by the fact that shortest obstacle-avoiding paths in $\mathbb{R}^2$ can be constructed in polynomial time, at least in the familiar algebraic model of computation).

In general, the path encoding approach involves first constructing a basic environment that admits exactly 2^k distinct shortest paths (referred to as *canonical paths*) between the two specified placements. These canonical paths all have essentially the same length D_Φ that can be distinguished, using a number of bits that is polynomial in k, from all non-canonical paths. Canonical paths are associated with the distinct truth assignments to the variables $X_0, \ldots, X_{k-1}$. Next, the environment is augmented with additional obstacles that serve to block (filter) every canonical path whose associated truth assignment does not satisfy the formula Φ.

In Reif and Wang's proof canonical paths pass through a sequence of checkpoints, at distinguishable angles and unequal–but essentially indistinguishable–lengths. The complexity of their construction arises from the rather sensitive analysis needed to show that as paths continue and errors propagate these properties are preserved. This exploits, among other things, the existence of pinhole gaps between obstacles, at which the checkpoints are located.

2.1 Overview of the proof

We avoid the complexity of the Reif-Wang construction by mimicking the proof, due to Asano *et al.* [3], of the NP-hardness of minimum-length motion planning for a rod (measuring the trace length of any fixed point), the first construction to employ the path-encoding approach in a planar setting. In this variant, canonical paths all have *exactly* the same length D_ϕ. In fact, canonical paths all have exactly the same length as they pass a sequence of checkpoints, vertical lines in our construction. Between these checkpoints the environment consists of elementary modules, each of which performs some basic manipulation of the canonical paths that enter the module. In fact, as we will argue, the properties of our modular construction are *decomposable* in the sense that they assume paths respect curvature constraints only*within* modules; in the transitions *between* modules only continuity is assumed. As a consequence, local analysis alone supports a global conclusion: if Φ is not satisfiable then any admissible path from (s, S) to (t, T) has length that exceeds D_Φ by an amount that can be expressed in a number of bits that is polynomial in k. Hence, even though D_Φ itself may not be exactly expressible in a polynomial (in k) number of bits, there exists a distance $D \geq D_\Phi$ that can be so expressed, such that there exists a (relaxed) admissible path of length at most D if and only of Φ is satisfiable.

Figure 1 illustrates the full reduction in schematic form. As it suggests, the construction is based on a sequence of three top-level modules. The first module, what we call a *Compound Beam Splitter*, splits a single in-coming canonical path into 2^k parallel canonical paths, indexed from 0 on the topmost path to $2^k - 1$ on the bottommost path, with fixed separation σ. We interpret the b-th bit of the binary representation of the index of a canonical path as a truth assignment to the variable X_b. Each of these paths has exactly the same length, measured from their common start point S to the vertical line L_1. The second module is a *Formula Filter* module that obstructs exactly those canonical paths whose associated truth assignment does not satisfy the formula Φ. The third module, a *Compound Beam Combiner*, is just the mirror image of the first, except that the 2^k in-coming paths have a smaller separation σ', the result of having passed through the Formula Filter.

It turns out that all three of these modules can be con-

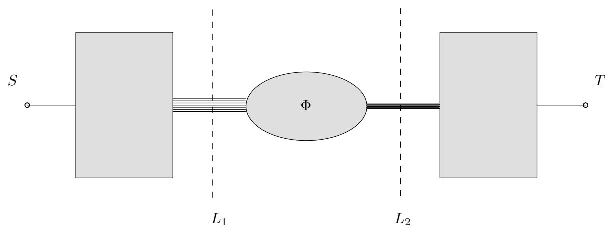

Figure 1: Full Reduction schematic.

structed by appending copies of one elementary (parameterized) module that we call a *Wide Beam Splitter* $WBS(\Delta)$ or its mirror image a *Wide Beam Combiner* $WBS^{-1}(\Delta)$ (Fig. 2). The details are analogous to those in the proof of Asano *et al.* [3].

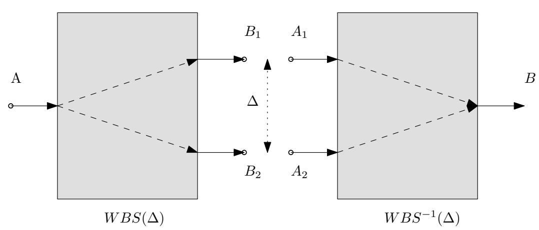

Figure 2: Wide Beam Splitter (WBS) and Combiner(WBS^{-1}) schematics.

2.2 Details of the wide beam-splitter module

The detailed construction of our Wide Beam Splitter is shown in Figure 3. It is described in terms of two parameters w, the width of the module, and Δ_w, the separation of the two canonical paths that emerge from the module. Since $w < 4$, it is straightforward to see that as w decreases Δ_w decreases. More precisely, since the horizontal (resp., vertical) separation of the centers of the left and right turning circle pairs is $w - 2$ (resp., $\sqrt{4w - w^2}$), we have $\Delta_w = 4 - 2\sqrt{4w - w^2}$.

As illustrated there are two canonical traversals of the Wide Beam Splitter. Both share a horizontal segment starting at the left terminal. Thereafter one (shown in solid red) traces a C^+C^-L path [2], emerging at the lower terminal while the other (shown in dashed green) traces a C^-C^+L path, emerging at the upper terminal. It is not hard to confirm that all admissible traversals must make a turn of length at least π on a circle tangent

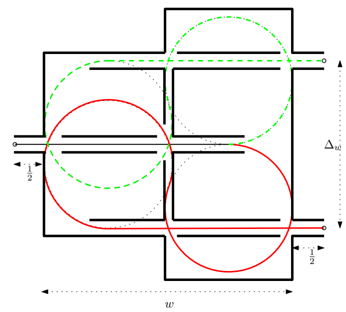

Figure 3: Wide Beam Splitter detail.

to the right boundary followed, not necessarily immediately, by a turn, of similar length but opposite direction, on a circle tangent to the left boundary.[3] Since any (even locally) shortest admissible path must have Dubins form between obstacle contacts, it follows directly that every shortest admissible traversal must have form $LCLCL$, where the C-segments are doubly supported by obstacles and the middle L-segment may have zero length.

Now, if we focus on admissible paths of form $LCLCL$ joining a point a on vertical line L_1 to a point b on a vertical line L_2 (see Figure 4(i)), it is easily confirmed that in any shortest such path the middle L-segment must have length zero, provided that w, the separation of L_1 and L_2, is less than 2 and the vertical separation of a and b is at least $\Delta_w/2$ (otherwise, fixing one of the turning circles while moving the other so that the middle L segment degenerates to zero, shortens the path). Among all such $LCCL$-transitions joining points a and b we can show that the shortest transition has the symmetric form illustrated in Figure 4(ii), independent of the directions at a and b. Furthermore, the shortest such transition, over all pairs of points a and b with vertical separation at least $\Delta_w/2$ (again independent of the directions at a and b), is the one shown in Figure 4(iii) in which a and b have vertical separation exactly $\Delta_w/2$.

These minimality results are proved by reference to Figure 5. The LC-transition from p to q has length $\lambda = d_1 + d_2 + (\pi/2 - \theta) = (w - 1 + \sin\theta)/\cos\theta + (\pi/2 - \theta)$ and its y-projection has length $\lambda_y = y_1 + y_2 = ((w-1)\sin\theta + 1)/\cos\theta$. It is straightforward to confirm that (i) the derivative, with respect to θ, of λ is $((w-1)\sin\theta)/\cos^2\theta$ and (ii) the derivative, with respect to θ, of λ_y is $((w-1) + \sin\theta)/\cos^2\theta$. It follows that not only is λ minimized, over configurations with $\theta \geq 0$, when $\theta = 0$, but so also is the derivative of λ

[2] We denote by C^+, resp., C^- a clockwise (resp., counter-clockwise) oriented unit circular arc.

[3] A skeptical reader may in fact see alternative traversals of our Wide Beam Splitter, as illustrated. We note that such alternatives can be eliminated, at a sacrifice in clarity of the figure, by narrowing all of the internal corridors sufficiently.

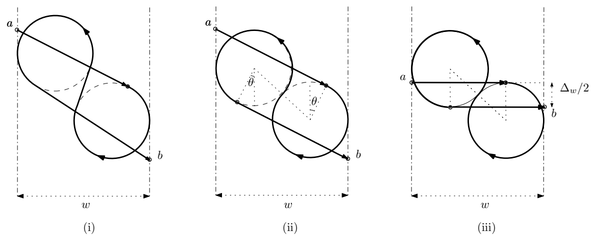

Figure 4: (i) Generic $LCLCL$ transition, (ii) minimum length transition between two specified points, and (iii) minimum length transition between two parallel lines.

with respect to λ_y.

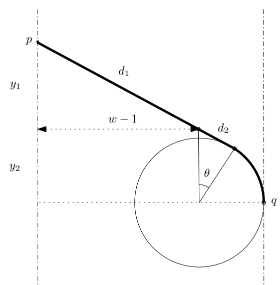

Figure 5: Generic LC transition joining points p and q.

2.3 Other remarks on the proof

Despite the comparative simplicity of our NP-hardness construction, the reader may object that we have traded one type of degeneracy in the construction (namely pinhole gaps between obstacles) for another (vertical obstacles spaced at distance exactly two, giving no horizontal freedom for turns in their midst). In fact, this property of our construction is imposed solely to simplify the argument; a very similar splitter construction is possible even if such degeneracies are forbidden.

Although this observation is not a distinguishing feature of our construction, it is worth noting that the hardness result remains intact even if we restrict our attention to shortest admissible paths of bounded intricacy.

2.4 Extensions

One of the advantages of introducing a simpler proof of the NP-hardness of finding shortest curvature-constrained paths in $\mathbb{R}^2$ is the potential this raises for establishing the hardness of other related path-planning problems. As an example, we point out that our results for standard curvature-constrained paths are easily modified to apply to polygonal (piecewise linear) paths that satisfy a novel parameterized notion of *discrete bounded curvature* [23] that coincides with the standard notion in the limit. A wide beam splitter designed for discrete bounded curvature paths is illustrated in Figure 6.

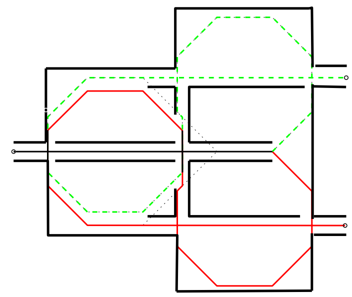

Figure 6: Beam Splitter gadget for discrete bounded-curvature paths.

3 Staying simple is hard

We prove that deciding existence of a simple admissible path in a polygonal domain is NP-hard by a reduction from planar 3SAT. Recall that the graph of a 3SAT instance has a vertex for each variable and a vertex for each clause. The graph edges connect clause-variable pairs whenever the variable belongs to the clause. In addition, the graph contains the cycle through variables (Fig. 7). 3SAT is hard even when restricted to instances with planar graphs [27]. We identify a 3SAT instance with its graph.

We transform the graph I of a planar 3SAT instance to an instance of the path finding problem, following the steps analogous to those used to show hardness of finding a simple thick wire in a polygonal do-

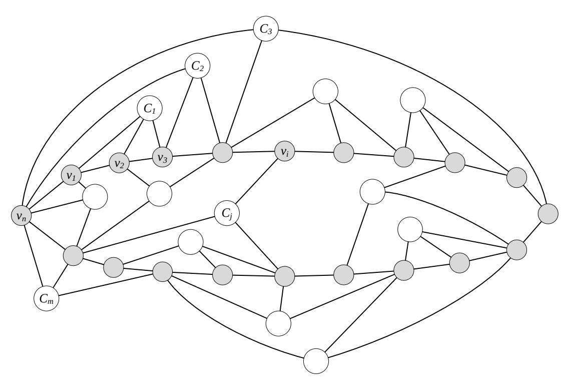

Figure 7: (Graph of) an instance I of 3SAT. Variables are shaded circles, clauses are hollow circles.

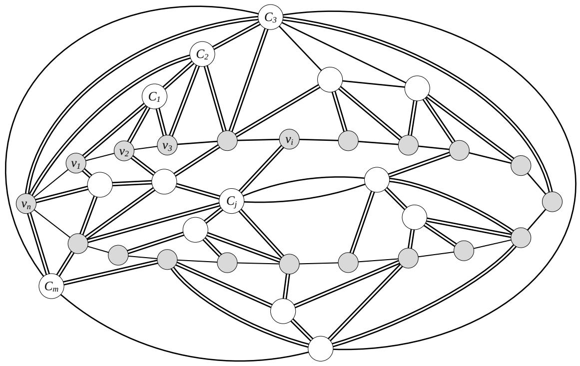

Figure 8: I augmented with parent-child edges, sibling edges, and with variable-clause edges duplicated.

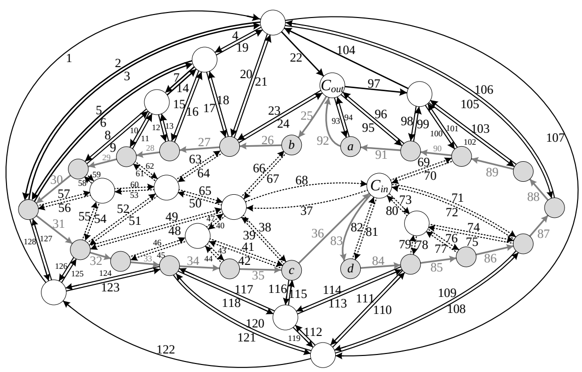

Figure 9: The closed spanning walk $\mathcal{W}$; the numbers indicate the order in which edges are traversed.

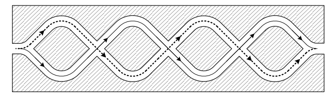

Figure 10: A variable gadget.

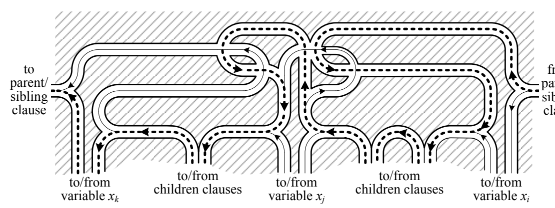

Figure 11: When a clause gadget is traversed, from left to right, one of the channels leading to variables must be used. Otherwise, 3 subpaths go through the top of the gadget leading to a self-intersection.

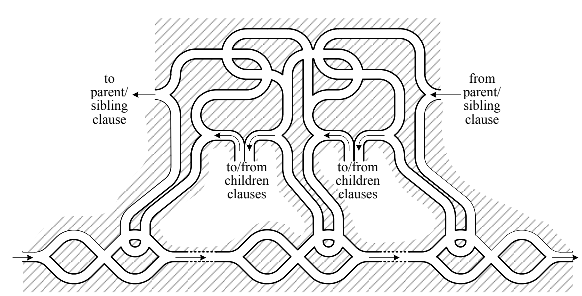

Figure 12: If a variable does not satisfy a clause the channel between them cannot be used by simple path.

main [24]: First, augment I with "parent-child" and "sibling" edges between clauses, and duplicate variable-clause edges (Fig. 8). Define a DFS-walk $\mathcal{W}$ in the augmented I: the walk goes through orphan clauses, recursing to children clauses, then to variables and to sibling clauses (Fig. 9). Replace vertices of I by variable and clause gadgets (Figs. 10, 11). Finally, turn edges of I into corridors connecting clause and variable gadgets.

The crucial ingredients of the proof are the following properties: (1) an admissible path must follow the walk $\mathcal{W}$ (the only flexibility is what clause-variable channel to use in each clause); (2) a variable gadget can be traversed in one of the two ways (setting the truth assignment); (3) for any clause gadget, one of the channels leading to a variable must be used by the path; (4) a variable-clause channel may be used only if the variable satisfies the clause (Fig. 12). Thus, there exists a simple admissible path in the instance iff in each clause there is a channel that can be used by a path to the variable satisfying the clause.

References

[1] P. K. Agarwal, T. Biedl, S. Lazard, S. Robbins, S. Suri, and S. Whitesides. Curvature-constrained shortest paths in a convex polygon. *SoCG'98*.

[2] P. K. Agarwal, P. Raghavan, and H. Tamaki. Motion planning for a steering-constrained robot through moderate obstacles. *SToC'95*.

[3] T. Asano, D. G. Kirkpatrick, and C.-K. Yap. Minimizing the trace length of a rod endpoint in the presence of polygonal obstacles is np-hard. *CCCG'03*.

[4] J. Backer and D. Kirkpatrick. Finding curvature-constrained paths that avoid polygonal obstacles. *SoCG'07*.

[5] J. Backer and D. Kirkpatrick. A complete approximation algorithm for shortest bounded-curvature paths. *ISAAC'08*.

[6] S. Bereg and D. Kirkpatrick. Curvature-bounded traversals of narrow corridors. *SoCG'05*.

[7] S. Bitner, Y. K. Cheung, A. F. Cook, O. Daescu, A. Kurdia, and C. Wenk. Visiting a sequence of points with a bevel-tip needle. *LATIN'10*.

[8] J.-D. Boissonnat and X.-N. Bui. Accessibility region for a car that only moves forwards along optimal paths. Research Report 2181, INRIA Sophia-Antipolis, 1994.

[9] X.-N. Bui, P. Souères, J.-D. Boissonnat, and J.-P. Laumond. Shortest path synthesis for Dubins nonholonomic robot. *ICRA'94*.

[10] J. Canny and J. H. Reif. New lower bound techniques for robot motion planning problems. *FoCS'87*.

[11] H. Chitsaz and S. LaValle. Time-optimal paths for a Dubins airplane. *Conf. Dec. and Cont.*, 2007.

[12] H. Chitsaz, S. M. Lavalle, D. J. Balkcom, and M. T. Mason. Minimum wheel-rotation paths for differential-drive mobile robots. *Int. J. Rob. Res.*, 28:66–80, 2009.

[13] H. R. Chitsaz. *Geodesic problems for mobile robots*. PhD thesis, 2008.

[14] K. Djath, A. Siadet, M. Dufaut, and D. Wolf. Navigation of a mobile robot by locally optimal trajectories. *Robotica*, 17:553–562, 1999.

[15] L. E. Dubins. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *Amer. J. Math.*, 79:497–516, 1957.

[16] V. Duindam, X. Jijie, R. Alterovitz, S. Sastry, and K. Goldberg. Three-dimensional motion planning algorithms for steerable needles using inverse kinematics. *Int. J. Rob. Res.*, 29:789–800, June 2010.

[17] E. Edison and T. Shima. Integrated task assignment and path optimization for cooperating uninhabited aerial vehicles using genetic algorithms. *Comput. Oper. Res.*, 38:340–356, January 2011.

[18] S. Foldes. Decomposition of planar motions into reflections and rotations with distance constraints. *CCG'04*.

[19] S. Fortune and G. Wilfong. Planning constrained motion. *Annals of Math. and AI*, 3:21–82, 1991.

[20] A. A. Furtuna and D. J. Balkcom. Generalizing Dubins curves: Minimum-time sequences of body-fixed rotations and translations in the plane. *Int. J. Rob. Res.*, 29:703–726, May 2010.

[21] P. R. Giordano and M. Vendittelli. Shortest paths to obstacles for a polygonal dubins car. *IEEE Transactions on Robotics*, 25(5):1184–1191, 2009.

[22] P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Nonholonomic Motion Planning*, 1992.

[23] D. Kirkpatrick and V. Polishchuk. Polygonal paths of bounded curvature. In preparation, 2011.

[24] I. Kostitsyna and V. Polishchuk. Simple wriggling is hard unless you are a fat hippo. *FUN'10*.

[25] J.-C. Latombe. *Robot Motion Planning*. Kluwer, 1991.

[26] Z. Li and J. F. Canny, editors. *Nonholonomic Motion Planning*. Kluwer, 1992.

[27] D. Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.

[28] X. Ma and D. A. Castacyn. Receding horizon planning for Dubins traveling salesman problems. *Conf. Dec. and Contr.*, 2006.

[29] A. A. Markov. Some examples of the solution of a special kind of problem on greatest and least quantities. *Soobshch, Kharkovsk. Mat Obshch*, 1:250-276, 1887 (in Russian).

[30] F. Morbidi, F. Bullo, and D. Prattichizzo. On visibility maintenance via controlled invariance for leader-follower dubins-like vehicles. *Conf. Dec. and Contr.*, 2008.

[31] J. Reif and H. Wang. Non-uniform discretization for kinodynamic motion planning and its applications. *WAFR'96*.

[32] J. Reif and H. Wang. The complexity of 2d curvature-constrained shortest-path problem. *WAFR'98*.

[33] P. Robuffo Giordano and M. Vendittelli. The minimum-time crashing problem for the Dubins car. *SYROCO'06*.

[34] K. Savla, E. Frazzoli, and F. Bullo. On the Dubins traveling salesperson problems: Novel approximation algorithms. *Robotics: Science and Systems II*, 2006.

[35] J. Sellen. Approximation and decision algorithms for curvature-constrained path planning: A state-space approach. *WAFR'98*.

[36] M. Sigalotti and Y. Chitour. Dubins' problem on surfaces ii: Nonpositive curvature. *SIAM J. Control and Optimization*, 45(2):457–482, 2006.

[37] H. J. Sussman. Shortest 3-d paths with a prescribed curvature bound. *Conf. Dec. and Contr.*, 1995.

[38] H. Wang and P. K. Agarwal. Approximation algorithms for curvature constrained shortest paths. *SoDA'96*.

[39] G. Wilfong. Motion planning for an autonomous vehicle. *ICRA'98*.

[40] G. Wilfong. Shortest paths for autonomous vehicles. *ICRA'89*.