Approximation of Curvature-Constrained Shortest Paths through a Sequence of Points

Jae-Ha Lee[1], Otfried Cheong[2], Woo-Cheol Kwon[3], Sung Yong Shin[3], and Kyung-Yong Chwa[3]

[1] Max-Planck-Institut für Informatik, Germany. Email: lee@mpi-sb.mpg.de
[2] Dept. of Computer Science, HKUST, Hong Kong. Email: otfried@cs.ust.hk
[3] Dept. of Computer Science, KAIST, Korea
E-mail: {famor,syshin,kychwa}@jupiter.kaist.ac.kr

Abstract. Let B be a point robot moving in the plane, whose path is constrained to forward motions with curvature at most 1, and let $\mathcal{X}$ denote a sequence of n points. Let s be the length of the shortest curvature-constrained path for B that visits the points of $\mathcal{X}$ in the given order. We show that if the points of $\mathcal{X}$ are given *on-line* and the robot has to respond to each point immediately, there is no strategy that guarantees a path whose length is at most $f(n)s$, for any finite function $f(n)$. On the other hand, if all points are given at once, a path with length at most $5.03s$ can be computed in linear time. In the *semi-online* case, where the robot not only knows the next input point but is able to "see" the future input points included in the disk with radius R around the robot, a path of length $(5.03 + O(1/R))s$ can be computed.

1 Introduction

Consider a point robot in the plane whose turning radius is constrained to be at least 1. More formally, given a continuous differentiable path $\mathcal{P} : I \longrightarrow R^2$ parameterized by an arc length $s \in I$, the *average curvature* of $\mathcal{P}$ is defined by $|\mathcal{P}'(s_1) - \mathcal{P}'(s_2)|/|s_1 - s_2|$. We require that the robot's path has an average curvature of at most 1 in every interval. This restriction corresponds naturally to constraints imposed by the steering mechanism found in car-like robots, a *non-holonomic constraint*.

In spite of considerable interest and recent work on non-holonomic motion planning problems in the robotics literature, they have received relatively little attention from a theoretical point of view, and in fact appear to be considerably harder than holonomic motion-planning problems.

In this paper, we study the path planning problem for a steering-constrained robot moving through a prespecified sequence of points. We want to minimize the length of the path. Any solution must decide the orientation with which the robot passes each point, and this decision will affect the length of the resulting path.

The problem of finding the shortest path through a sequence of points has not attracted much attention so far, since, in general, the shortest path is simply the union of the shortest paths connecting each pair of consecutive points. In our curvature-constrained case, however, the shortest path through a sequence of points must satisfy the curvature constraint at each "joint," and is not the union of the shortest paths between consecutive points (which are simply line segments).

Previous results. Most of the previous work considers the curvature-constrained path planning problem between two given configurations (a configuration specifies both a point and a direction of travel). Dubins [7] was perhaps the first to study curvature-constrained shortest paths. He proved that, in the absence of obstacles, a curvature-constrained shortest path from any start configuration to any final configuration consists of at most three segments, each of which is either a straight line or an arc of a unit (radius) circle, assuming that the curvature of the path is upper bounded by 1. Since Dubins' characterization implies a constant-time algorithm to find a shortest path between two configurations, it is easy to obtain a linear-time algorithm for computing a shortest path through a sequence of *configurations*. Reeds and Shepp [11] extended this obstacle-free characterization to robots that are allowed to make reversals, that is, to back up. Using ideas from control theory, Boissonnat, Cerezo, and Leblond [4] gave an alternative proof for both cases.

There has also been work on computing curvature-constrained paths between two configurations in the presence of obstacles [8, 9, 13]. Agarwal, Raghavan, and Tamaki [2] considered the restricted case of pairwise disjoint *moderate obstacles* (convex obstacles whose boundary has curvature bounded by 1) and gave efficient approximation algorithms. For the same problem, Boissonnat and Lazard [5] gave an $O(n^2 \log n)$-time exact algorithm. Reif and Wang [12] confirmed that the problem of deciding whether there exists a curvature-constrained shortest path amidst general obstacles is NP-hard. Agarwal et al. [1] presented an $O(n^2 \log n)$-time algorithm to compute a curvature-constrained shortest path between two configurations inside a convex polygon. Recently, Ahn et al. [3] characterized the set of all points (without orientations) reachable from a given configuration inside a convex polygon.

Our model and results. Let B be a point robot and $\mathcal{X} = \langle X_1, X_2, \cdots, X_n \rangle$ denote a sequence of points in the plane. We aim to find a curvature-constrained path passing through the points in $\mathcal{X}$ in the given order; there is no restriction on the orientation with which it arrives at (and departs from) each point. The lack of such a restriction, which is the main difference between our problem and the previous work, is a double-bladed sword—on the one hand, the freedom to choose a direction at each point allows for drastically shorter paths, on the other hand, finding such a path is considerably more involved.

A configuration σ for B is a pair $\sigma = (X, \phi)$, where X is a point in the plane representing the location of the robot and ϕ is a unit vector representing its orientation. We emphasize again that a shortest path between two configurations can be computed in constant time, due to Dubins [7]. In order to find a shortest path through $\mathcal{X}$, it suffices to determine the orientation ϕ_i at each point X_i so that

$$\sum_i ||\text{shortest path from } (X_i, \phi_i) \text{ to } (X_{i+1}, \phi_{i+1})||$$

is minimized, where $|| \cdot ||$ denotes the length of a path.

An algorithm for this problem is called $f(n)$-*competitive* (or $f(n)$-*approximate)* if, given a sequence $\mathcal{X}$ of length n, it computes a path of length at most $f(n)s + c_0$, where s is the length of the shortest path visiting the points of $\mathcal{X}$ and $c_0 \geq 0$ is a constant (independent of $\mathcal{X}$ and n).

We first consider the on-line version of the problem, where the points X_i are given one by one, and the robot has to respond to each new point by moving to it. We show that there is no $f(n)$-competitive algorithm for any finite function $f(n)$, settling an open problem by Agarwal et al. [2].

We then look at the case where the sequence $\mathcal{X}$ is given in advance. An intuitive approach for computing an approximate shortest path would be to choose a discrete set of orientations for each point, and to try all combinations of possible orientations. Somewhat surprisingly, it turns out that this approach cannot guarantee a bounded approximation factor, as we will see in Section 3.

Our main result is a linear time algorithm that, given a sequence $\mathcal{X}$, computes a path whose length is at most 5.03 times the length of the shortest path.

Finally, we mention a $(5.03 + O(1/R))$-competitive algorithm for a *semi-online* version where the robot can "see" the whole prefix of the remaining sequence contained in a disk with radius R around it.

2 Geometric preliminaries

Given a configuration $\sigma = (X, \phi)$, the oriented line passing through X with orientation ϕ is denoted L_σ. A configuration $\sigma = (X, \phi)$ belongs to an oriented path (or curve) $\mathcal{C}$ if $X \in \mathcal{C}$ and L_σ is the oriented tangent line to $\mathcal{C}$ at X. Note that a configuration σ belongs to two oriented unit-radius circles. We denote these two circles by C_σ^+ and C_σ^-, where C_σ^+ is oriented counterclockwise, C_σ^- is oriented clockwise. For a circle $\mathcal{C}$, $closed(\mathcal{C})$ (resp. $open(\mathcal{C})$) denotes the closed (resp. open) disc bounded by $\mathcal{C}$. If X and Y are two points on a simple oriented curve $\mathcal{C}$, then $\mathcal{C}[X, Y]$ denotes the portion of $\mathcal{C}$ from X to Y, including X and Y. For two orientations ϕ and ϕ', let $[\phi, \phi']$ denote the counterclockwise interval of orientations between ϕ and ϕ' (that is, the orientations met when we rotate ϕ counterclockwise to ϕ'). For convenience, let $[0, 2\pi]$ denote the interval of all orientations.

The following fact is a direct by-product of a proof by Pestov and Ionin [10], see also Ahn et al. [3].

Fact 1 *Given a configuration σ and a point Y in the interior of C_σ^+ or C_σ^-. Then the length of any path from σ to Y is at least π.*

Segments and Dubins paths. Let $\mathcal{C}$ be a path. We call a nonempty subpath of $\mathcal{C}$ a C-*segment* (resp. S-*segment*) if it is a circular arc of unit radius (resp. a line segment) and maximal. A *segment* is either a C-segment or an S-segment. Suppose $\mathcal{C}$ consists of a C-segment, S-segment, and a C-segment, in this order; then we will say that $\mathcal{C}$ is of type CSC. Dubins [7] proved the following result.

Fact 2 *Any shortest path between two configurations is of type CCC or CSC, or a substring thereof.*

We will refer to paths of type CCC or CSC or a substring thereof as *Dubins paths*. We emphasize that an optimal path between two given configurations can be computed in constant time.

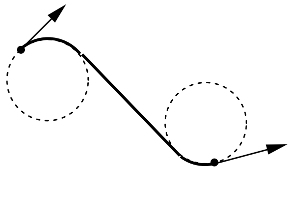

Fig. 1. Different types of Dubins paths.

Direct paths and detour paths. Suppose that X and Y are two consecutive points in $\mathcal{X}$ such that $dist(X,Y) < 2$. We can draw two unit (radius) circles each of which passes both of X and Y, we denote them by C^+_{XY} and C^-_{XY} (Figure 2). Any path from X to Y that is included in $closed(C^+_{XY}) \cap closed(C^-_{XY})$ is called a *direct path*. A path that is not direct is called a *detour path*. A detour path will pass some point outside $closed(C^+_{XY}) \cap closed(C^-_{XY})$ before arriving at Y. Let ϕ^U_{XY} (resp. ϕ^L_{XY}) denote the orientation of the clockwise tangent to C^-_{XY} (counterclockwise tangent to C^+_{XY}) at X. Note that any direct path from X to Y must start with an orientation in $[\phi^L_{XY}, \phi^U_{XY}]$.

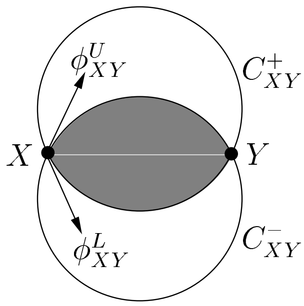

Fig. 2. A path included in $closed(C^+_{XY}) \cap closed(C^-_{XY})$ is called a direct path.

Lemma 1 *The length of any detour path is at least π.*

Proof: Let $\mathcal{P}$ be any detour path from X to Y. Let $\sigma = (X, \phi)$ be its configuration at X. There are three cases depending on the orientation ϕ. See Figure 3.

- Case 1. ϕ is directed to the outside of $open(C^+_{XY}) \cup open(C^-_{XY})$ (Figure 3a). Consider $open(C^+_\sigma) \cup open(C^-_\sigma)$. If the path $\mathcal{P}$ passes a point in its interior, the length is longer than π by Fact 1. Otherwise the resulting path must "wind around" C^+_σ or C^-_σ, so its length is at least π.
- Case 2. ϕ is directed into $open(C^+_{XY}) \cup open(C^-_{XY})$ but not $closed(C^+_{XY}) \cap closed(C^-_{XY})$ (Figure 3b).
  Observe that Y is contained in $open(C^+_\sigma)$ (or $open(C^-_\sigma)$). Therefore the claim follows directly from Fact 1.

- Case 3. ϕ is directed into $closed(C^+_{XY}) \cap closed(C^-_{XY})$ but later leaves it (Figure 3c).
  Suppose that the robot leaves $closed(C^+_{XY}) \cap closed(C^-_{XY})$ at X'. If we recursively apply this lemma to the path from X' to Y, then the orientation at X' is towards the outside of $closed(C^+_{X'Y}) \cap closed(C^-_{X'Y})$. Thus a path from X' to Y falls into the above two cases, so its length is at least π.

Any detour path falls into one of the three categories above. □

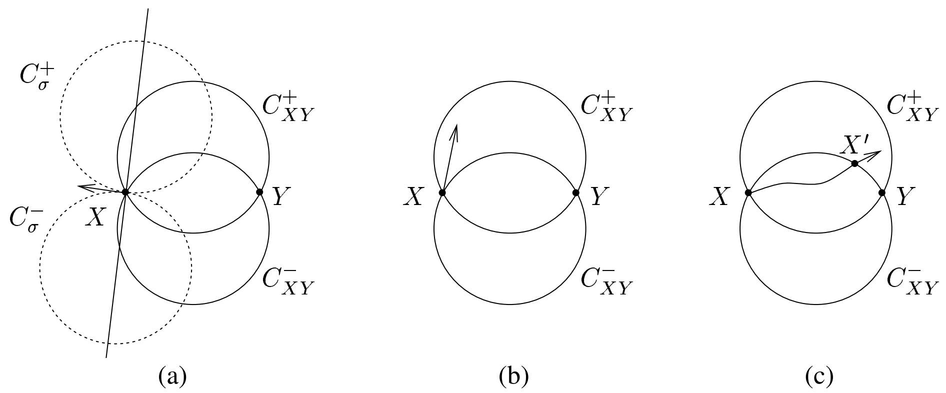

Fig. 3. Proof of Lemma 1

In contrast to the detour path, the length of any direct path from X to Y is bounded by a constant times $dist(X, Y)$. Let us assume that C^+_{XY} is oriented counterclockwise.

Lemma 2 *The length of any direct path is no greater than* $||C^+_{XY}[X, Y]||$*, which is bounded by* $\frac{\pi}{2} dist(X, Y)$ $(< \pi)$.

Proof: Assume the line XY is parallel to the x-axis. Let $\mathcal{P}$ denote an arbitrary direct path from X to Y. Suppose that $\mathcal{P}$ consists of k segments (each of which is a line segment or a circuclr arc of radius at least 1) such that each segment connects X_i to X_{i+1}, where $X_0 (= X), X_1, \cdots X_k (= Y)$ are points on $\mathcal{P}$ occurring in this order. We prove this lemma by induction on k. The base case that $k = 1$ is trivial since $\mathcal{P}$ is one line segment or circular arc in this case. Without loss of generality, assume X_1 is below the line XY. Let X_1' (resp. X_1'') denote the first point on C^+_{XY} met when we rotate X_1 clockwise centered at X (resp. counterclockwise centered at Y). See Figure 4.

Since $\mathcal{P}[X, X_1]$ is one segment, its length is no greater than $||C^+_{XY}[X, X_1']||$. Moreover, by induction hypothesis, the length of $\mathcal{P}[X_1, Y]$ is no greater than $||C^+_{XY}[X_1'', Y]||$. It is easily seen that x-coordinates of X_1', X_1 and X_1'' are non-decreasing in this order. Therefore, the length of $\mathcal{P}$ is no greater than $||C^+_{XY}[X, Y]||$, which proves this lemma. Since $dist(X, Y)$ is less than 2 from the definition of a direct path, the length of $\mathcal{P}$ is less than π. □

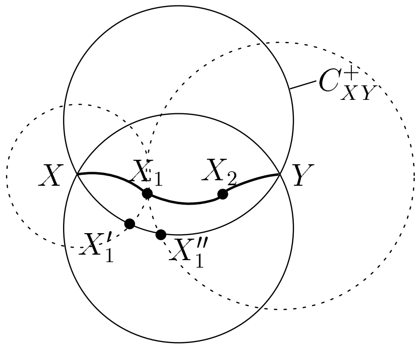

Fig. 4. Proof of Lemma 2.

The proof of the next lemma can be found in the full paper.

Lemma 3 *Given a configuration $\sigma = (X, \phi)$ and a point Y at distance less than 2 from X. The set of orientations with which a direct path from σ to Y can arrive at Y is an interval $[\phi_L, \phi_U]$. The orientation ϕ_L is the direction of the clockwise tangent to $\mathcal{C}$ at Y, where $\mathcal{C}$ is the circle containing Y that is tangent to C^+_σ in closed$(C^+_{XY}) \cap$ closed(C^-_{XY}). The orientation ϕ_U is obtained symmetrically. (See Figure 5)*

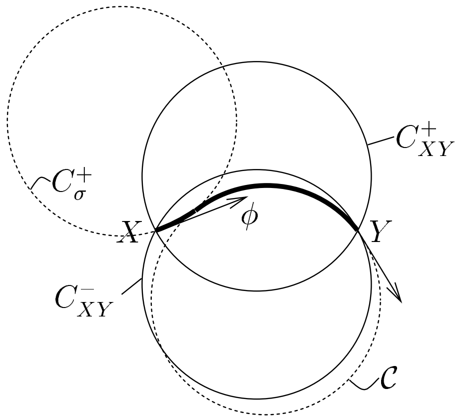

Fig. 5. Lemma 3.

Basic idea. In order to bound the approximation ratio of an algorithm, we need a good lower bound on the value of the optimal solution. The simplest such lower bound is the sum of the Euclidean distances between consecutive points. This bound is good enough when consecutive points are far apart, or if they can be connected with a direct path. The difficulty arises when three consecutive points are close to each other and form a sharp turn—in that case we cannot avoid using a detour path. The length of a detour path is at least π by Lemma 1, which can be arbitrarily longer than the distance of the points. It follows that in the heart of our argument we need a bound on the number of detour paths made by both the optimal path and the path produced by the algorithm. We will first show that if the points are given one-by-one in an on-line fashion, any algorithm can be fooled into producing a large number of detour paths, while the optimal algorithm doesn't need any. Then we will see that if all points are given in advance, we can construct a path using that minimum possible number of detour paths.

3 On-line algorithm

The following on-line version of our problem was posed by Agarwal et al. [2]: Assume the point sequence $\mathcal{X}$ is given one by one in the form of point requests, and the robot has to service each request by moving to the point. There is no constraint on the orientation with which it arrives at the requested point.

We show that even in the absence of obstacles no $f(n)$-competitive on-line algorithm exists, for any finite function $f(n)$.

Theorem 4 *No on-line algorithm can be $f(n)$-competitive, for any finite function $f(n)$.*

Proof: Let $\varepsilon > 0$ and consider some on-line algorithm. We will incrementally construct a sequence $\mathcal{X}$ of $2n + 1$ point requests, making use of the decisions made by the algorithm to serve the requests in constructing the next point. The distance between consecutive points will be ε. The sequence will be such that there is a path visiting all points using only direct paths, but the on-line algorithm will be forced to make at least n detour paths. By Lemma 1 the path returned by the algorithm has length at least $n\pi$. The shortest path, on the other hand, has length bounded by $\frac{\pi}{2}n\varepsilon$ by Lemma 2. Since ε can be chosen arbitrarily small, the theorem follows.

Let $\mathcal{X} = X_1, \cdots, X_{2n+1}$ be the sequence to be constructed, and let X_1 be an arbitrary point in the plane. Suppose that requests $X_1, \ldots, X_{2i-1}$ have already been created, and so the robot is currently at X_{2i-1}.

Let ϕ_{2i-1} be some direction with which a path consisting of direct paths only can arrive at X_{2i-1}. It will follow from our construction that such an orientation exists (in fact, there will be exactly one such orientation).

We create the next point request X_{2i} at distance ε from X_{2i-1} in direction ϕ_{2i-1}, see Figure 6. By Lemma 3, the set of directions with which a direct path starting at the configuration (X_{2i-1}, ϕ_{2i-1}) can arrive at X_{2i} is an interval $[l_{2i}, u_{2i}]$. Recall that the circle $C^+_{(X_{2i},u_{2i})}$ (resp. $C^-_{(X_{2i},l_{2i})}$) is tangent to the orientation u_{2i} (resp. l_{2i}).

We now observe the direction ϕ in which the robot controlled by the on-line algorithm arrives at X_{2i}. If ϕ points to the left of the oriented line through X_{2i-1} and X_{2i}, we choose X_{2i+1} to be the point at distance ε from X_{2i} on the circle $C^-_{(X_{2i},l_{2i})}$ (the lower one in Figure 6). Otherwise, we choose X_{2i+1} symmetrically on the circle $C^+_{(X_{2i},l_{2i})}$ (the upper one in Figure 6).

The on-line algorithm has no choice but to use at least one detour path while serving the requests X_{2i} and X_{2i+1}. On the other hand, it is clearly possible to visit all points

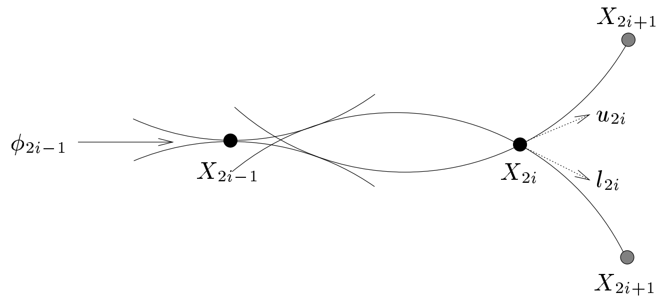

Fig. 6. Proof of Theorem 4. Note that $dist(X_{2i-1}, X_{2i})$ is very small compared to the radius of circles.

using direct paths only. ◻

A close look at the construction in this section reveals that any robot arriving at X_{2i-1} with orientation that is different from ϕ_{2i-1} cannot go through X_{2i} and X_{2i+1} using direct paths only. In other words, there is only one possible path from X_{2i-1} to X_{2i+1} that uses direct paths only. This implies that any approximation technique that tries to discretize the set of possible directions at each point before looking at the geometry will have to make at least n detour paths to visit the sequence we constructed, even if it could use a countably infinite set of orientations! It follows that no bounded approximation ratio can be guaranteed by such an algorithm.

4 Approximation algorithm

We now consider the problem where the whole sequence $\mathcal{X}$ is given in advance. We describe a linear-time approximation algorithm that computes a path whose length is within a constant factor of the optimal. The basic idea is simple:

In case two consecutive input points X_i and X_{i+1} are far away from each other, say $dist(X_i, X_{i+1}) \geq 2$, we can arbitrarily choose orientations ϕ_i and ϕ_{i+1} and connect the configurations (X_i, ϕ_i) and (X_{i+1}, ϕ_{i+1}) using the shortest path. The length of the shortest path—a Dubins path—is bounded by a constant times $dist(X_i, X_{i+1})$ (Lemma 7).

The remaining case is that X_i and X_{i+1} are close, that is, $dist(X_i, X_{i+1}) < 2$. There are two apparent ways to connect X_i with X_{i+1}: we can either use a direct path, or a detour path. The length of a direct path is again bounded by a constant times $dist(X_i, X_{i+1})$, but its orientations at X_i and X_{i+1} are restricted. A detour path, on the other hand, can be used for arbitrary orientations, but its length can be far longer than the distance of the two points. In other words, if we connect X_i to X_{i+1} using a detour path, there is no guarantee on the approximation ratio for the path between these two consecutive points. We are nevertheless able to obtain a constant approximation ratio for the total sequence by constructing a path that uses the *minimum possible number of detour paths*.

Consider a *dense* subsequence $X_i, \cdots, X_j$ such that $dist(X_k, X_{k+1}) < 2$ for all $i \leq k < j$. Let Φ_k denote the set of orientations with which the robot can arrive at X_k using direct paths only from X_i. We assume that any orientation can be used in X_i, so $\Phi_i = [0, 2\pi]$. Suppose Φ_k is not empty. In order to go to X_{k+1} using a direct path, the robot must start with an orientation in $\Phi_k \cap [\phi^L_{X_k X_{k+1}}, \phi^U_{X_k X_{k+1}}]$. We can compute Φ_{k+1} using the following lemma. We define the size of an interval $[\phi, \phi']$ of orientations to be the angle between ϕ and ϕ' measured from ϕ counterclockwise.

Lemma 5 *If Φ_k is either $[0, 2\pi]$ or an interval of size less than π, then so is Φ_{k+1}. Furthermore, Φ_{k+1} can be computed from Φ_k in constant time.*

Proof: Under the conditions of the lemma, $\Phi_k \cap [\phi^L_{X_k X_{k+1}}, \phi^U_{X_k X_{k+1}}]$ is an interval; Let $[\phi_L, \phi_U]$ denote it. By Lemma 3, the set of orientations Φ_{k+1} is then also an interval $[\phi'_L, \phi'_U]$, where ϕ'_L is obtained from ϕ_U and ϕ'_U is obtained from ϕ_L. ◻

Repeated application of Lemma 5 proves that Φ_k is an interval of length less than π, for $k > i$. The following two constants will determine the quality of our approximation.

Definition 1 *Let*

$$\alpha = \sup\{\ell(\sigma, \sigma')/dist(\sigma, \sigma') \mid \sigma, \sigma' \textit{ configurations with } dist(\sigma, \sigma') \geq 2\},$$
$$\beta = \sup\{\ell(\sigma, \sigma') \mid \sigma, \sigma' \textit{ configurations with } dist(\sigma, \sigma') \leq 2\},$$

where $\ell(\sigma, \sigma')$ denotes the length of a Dubins shortest path between two configurations σ and σ'.

We now describe our main theorem.

Theorem 6 *A $\max(\alpha, \pi/2 + \beta/\pi)$-approximate shortest path subject to the curvature constraint can be computed in linear time.*

Proof: The input sequence $\mathcal{X}$ consists of alternating dense and non-dense subsequences, so that the final point of one subsequence is the first point of the other subsequence. We first determine the orientation at each point in dense subsequences.

Let $X_i, \cdots, X_j$ be a dense subsequence. While scanning it from X_i, we compute Φ_k, for $k = i, \cdots, j$, until it becomes empty. Suppose Φ_{k+1} becomes empty for the first time. Then we construct a path from X_i to X_k using direct paths only. Next, leaving (X_k, X_{k+1}) not connected, we restart from X_{k+1}, as if this is the first point of a dense subsequence, and repeat the same procedure till the point X_j. The remaining dense consecutive points, such as (X_k, X_{k+1}), that have not been connected by a direct path, are then connected by a detour path (the Dubins shortest path respecting the orientations determined by the direct paths). We apply the same procedure to each dense subsequence. Finally, we connect all remaining pairs of consecutive points by Dubins shortest paths, choosing the orientation at each point freely while respecting the predetermined orientations at points in dense subsequences.

We claim that this procedure produces a path using the minimum possible number of detour paths. In fact, consider a maximal sequence of points $X_i, \cdots, X_k$ connected by direct paths by the algorithm. By our construction, any path that uses a direct path to connect X_k with X_{k+1} must use a detour path between X_i and X_k. This implies that no path can use fewer detour paths than the path we constructed. In particular this is also true for the optimal solution.

Our algorithm clearly runs in linear time using Lemma 5. It remains to analyze the approximation ratio.

Let $\mathcal{P}$ denote the oriented path that we generate and let $\mathcal{Q}$ denote an optimal path for the sequence. Let $d_i = 1$ if $dist(X_i, X_{i+1}) < 2$; 0 otherwise. For (X_i, X_{i+1}) with $d_i = 1$, let $D_i = 1$ if $\mathcal{P}[X_i, X_{i+1}]$ is a direct path; 0 otherwise. First, we bound the length of $\mathcal{P}$ restricted to dense subsequences.

$$\begin{aligned}
&\sum_i d_i \cdot ||\mathcal{P}[X_i, X_{i+1}]|| = \\
&= \sum_i d_i \cdot D_i \cdot ||\mathcal{P}[X_i, X_{i+1}]|| + \sum_i d_i \cdot (1 - D_i) \cdot ||\mathcal{P}[X_i, X_{i+1}]|| \\
&\leq \sum_i d_i \cdot D_i \cdot \pi/2 \cdot dist(X_i, X_{i+1}) + \text{(the number of detour paths in } \mathcal{P}) \cdot \beta \\
&\leq \sum_i d_i \cdot \pi/2 \cdot dist(X_i, X_{i+1}) + \text{(the number of detour paths in } \mathcal{Q}) \cdot \beta \\
&\leq \sum_i d_i \cdot \pi/2 \cdot ||\mathcal{Q}[X_i, X_{i+1}]|| + \beta/\pi \cdot \sum_i d_i \cdot ||\mathcal{Q}[X_i, X_{i+1}]|| \\
&\leq (\pi/2 + \beta/\pi) \cdot \sum_i d_i \cdot ||\mathcal{Q}[X_i, X_{i+1}]||
\end{aligned}$$

The first inequality is clear from Lemma 2. The second inequality comes from the fact that our procedure uses the smallest number of detour paths for $\mathcal{X}$. Therefore, we have the following.

$$\begin{aligned}
||\mathcal{P}|| &= \sum_i ||\mathcal{P}[X_i, X_{i+1}]|| \\
&= \sum_i (1 - d_i) \cdot ||\mathcal{P}[X_i, X_{i+1}]|| + \sum_i d_i \cdot ||\mathcal{P}[X_i, X_{i+1}]|| \\
&\leq \alpha \cdot \sum_i (1 - d_i) \cdot ||\mathcal{Q}[X_i, X_{i+1}]|| + (\pi/2 + \beta/\pi) \cdot \sum_i d_i \cdot ||\mathcal{Q}[X_i, X_{i+1}]|| \\
&\leq \max(\alpha, \pi/2 + \beta/\pi) \cdot ||\sum_i \mathcal{Q}[X_i, X_{i+1}]||
\end{aligned}$$

completing the proof. ◻

It remains to prove bounds on the constants α and β.

Lemma 7 $\alpha = 1 + \pi$

Proof: Let $\sigma = (X, \phi)$ and $\sigma' = (Y, \phi')$. Assume that X and Y lie on a horizontal line with X left of Y.

If the distance between X and Y is 2 and $\phi = \phi'$ is the left horizontal direction, there is no path from σ to σ' with length less than $2 + 2\pi$, and so $\alpha \geq 1 + \pi$. This can be seen by considering vertical lines through X and Y and applying Fact 1.

It remains to show $\alpha \leq 1 + \pi$. We prove this by a case analysis according to the orientations ϕ and ϕ'. Details can be found in the full paper. ◻

Lemma 8 $\beta \leq (5/2)\pi + 3$

Proof: Let $\sigma = (X, \phi)$ and $\sigma' = (Y, \phi')$. Assume that X and Y lie on a horizontal line with X left of Y. Consider a circle $\mathcal{C}$ with radius 2 around Y. We will construct a path from σ to σ' by passing through a configuration $\sigma^* = (X^*, \phi^*)$ on $\mathcal{C}$. Since X^* and Y have distance 2, the shortest path from σ^* to σ' has length at most $2 + 2\pi$ by Lemma 7. It remains to show that σ^* can be chosen such that the length of the path from σ to σ^* is at most $1 + \pi/2$. This can be done by following C_σ^+ or C_σ^- until the orientation is vertical, and then following a segment until $\mathcal{C}$ is reached. ◻

Plugging the bounds on α and β into Theorem 6, we obtain the following result.

Corollary 9 *A* 5.03*-approximate shortest path subject to the curvature constraint can be computed in linear time.*

5 Concluding remarks

The on-line and off-line models can be combined into a *semi-online* version. We suppose that the robot not only knows the next request point, but is also able to "see" the following points of the sequence that are included in the disk with radius R around the robot. The semi-online version is a generalization of both the off-line and the on-line problem, since the case $R = 0$ corresponds to the strict on-line problem, while $R = \infty$ is the off-line problem. The following result can be shown using a modification of our off-line approximation algorithm. Details can be found in the full paper.

Theorem 10 *There exists a* $(5.03+O(1/R))$*-competitive algorithm for the semi-online version with parameter R.*

One could have defined a different "semi-online" setting, where the robot is told about the following k points whenever it receives a new request. However, our counter-example in Section 3 can be modified to show that no algorithm can achieve a constant competitive ratio even for $k = o(n)$, as in Figure 7.

Our approximation algorithm guarantees that the minimum possible number of detour paths is used. That means that it has to construct the unique optimal path for the construction of Section 3 (the only path that uses only direct paths). Implementing this correctly will probably only be possible if some form of exact arithmetic is used.

Lemma 8 is not tight. A natural conjecture would be $\beta = 2\alpha$, which would imply that our approximation algorithm computes a 4.21-approximation to the shortest path. It would be nice to compute β exactly.

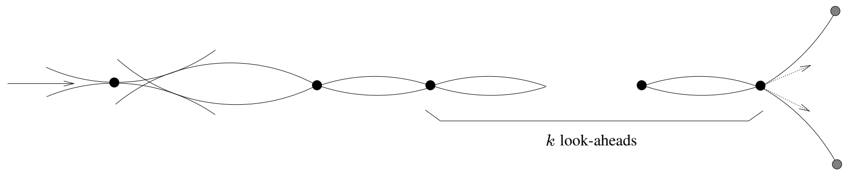

Fig. 7. Extended counter-example

Finally, the shortest path problem through a sequence of points may be of interest in more generalized models that considers the environment with obstacles and/or the restriction on the acceleration [6] instead of the curvature.

Acknowledgement. The part of this research done in KAIST was supported in part by the NRL(National Research Laboratory) of KISTEP (Korea Institute of Science and Technology Evaluation and Planning).

References

1. P. K. Agarwal, T. Biedl, S. Lazard, S. Robbins, S. Suri, and S. Whitesides. Curvature-constrained shortest paths in a convex polygon. In *Proc. 14th SCG*, pages 392–401, 1998.
2. P. K. Agarwal, P. Raghavan, and H. Tamaki. Motion planning of a steering-constrained robot through moderate obstacles. In *Proc. 27th STOC*, pages 343–352, 1995.
3. H.-K. Ahn, O. Cheong, J. Matoušek, and A. Vigneron. Reachability by paths of bounded curvature in convex polygons. Proc. 16th SCG (to appear), 2000.
4. J. D. Boissonnat, A. Cerezo, and J. Leblond. Shortest paths of bounded curvature in the plane. *Interat. J. Intell. Syst.*, 10:1–16, 1994.
5. J. D. Boissonnat and S. Lazard. A polynomial-time algorithm for computing a shortest path of bounded curvature admist moderate obstacles. In *Proc. 12th SCG*, pages 242–251, 1996.
6. B. Donald, P. Xavier, J. Canny, and J. Reif. Kinodynamic motion planning. *JACM*, 40(5):1048–1066, 1993.
7. 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.
8. S. Fortune and G. Wilfong. Planning constrained motion. *Ann. Math. Artif. Intell.*, 3:21–82, 1991.
9. P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Nonholonomic Motion Planning (Z. Li and J Canny, eds.)*, pages 271–342, 1992.
10. G. Pestov and V. Ionin. On the largest possible circle imbedded in a given closed curve. *Dok. Akad. Nauk SSSR*, 127:1170–1172, 1959. (In Russian).
11. J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes forwards and backwards. *Pacific J. Math.*, 145(2), 1990.
12. J. Reif and H. Wang. The complextiy of the two-dimensional curvature-constrained shortest-path problem. In *Proc. of 3rd Workshop on the Algorithmic Foundations of Robotics*, 1998.
13. H. Wang and P. K. Agarwal. Approximation algorithm for curvature constrained shortest paths. In *Proc. 7th SODA*, pages 409–418, 1996.