



UNIVERSITY OF LEEDS

This is a repository copy of *Optimal Smooth Paths Based on Clothoids for Car-like Vehicles in the Presence of Obstacles*.

White Rose Research Online URL for this paper:
<https://eprints.whiterose.ac.uk/165251/>

Version: Accepted Version

Article:

Lambert, ED orcid.org/0000-0002-2297-0441, Romano, R orcid.org/0000-0002-2132-4077 and Watling, D orcid.org/0000-0002-6193-9121 (2021) Optimal Smooth Paths Based on Clothoids for Car-like Vehicles in the Presence of Obstacles. *International Journal of Control, Automation and Systems*, 19. pp. 2163-2182. ISSN 1598-6446

<https://doi.org/10.1007/s12555-020-0179-1>

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.



eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Optimal Smooth Paths based on Clothoids for Car-like Vehicles in the Presence of Obstacles

Edward Derek Lambert, Richard Romano and David Watling

Abstract: Automated Guided Vehicles are increasingly used for material transfer in factory and warehouse environments amongst humans and human operated vehicles. Safe and efficient operation is challenging when there is a mix of human and automated traffic as fixed guide paths can become blocked more frequently. In this work we aim to show smooth and efficient paths based on clothoid curves can be used to automatically plan diversions which can be traversed at high speed by automated fork-lift vehicles, which are car-like in the sense they have a limited turning radius and angular acceleration. The approach, based on numerical optimisation within convex region constraints is described in detail, and numerical results for curvature and sharpness are compared to a cubic spline on a small number of simulated environments. The clothoid spline is less affected, in terms of its objective function, by a shift in the obstacle boundaries than a cubic spline, for obstacle shifts below 0.5m. The clothoid spline takes longer to converge for but the output path has attractive qualities like lower peak sharpness, enabling high speed operation. The method is therefore most useful for applications where path quality is important and updates are required less frequently. Changing the objective function by increasing weighting parameter b allowed the path shape to be tuned to reduce the peak sharpness, at the cost of increasing the total length. With $b > 100$, convergence was poor because parts of the spline were pushed outside the assigned region, an artefact arising from the constraints only being enforced at the start and end of each segment. The analytical Jacobian of the constraints was effective at reducing the number of function evaluations to reach convergence.

Keywords: Continuous curvature path planning, Non-linear, Optimal, Nonholonomic car-like vehicle, Obstacle Avoidance, Convex Regions

1. INTRODUCTION

Consider a fleet of Autonomous Guided Vehicles (AGV) moving material in an automated manufacturing plant. A lattice roadmap made up of virtual guide paths is a widely used solution for planning the motion of each AGV [1]. This is designed by engineers installing the AGV system, and may remain in use for many years. In environments which are shared with humans and human operated vehicles there is a greater probability of unexpected obstructions blocking the guide paths. Automatic replanning to avoid these obstructions has the potential to increase performance and robustness of shared environment AGV systems. Although numerous techniques have been developed for planning paths of car-like vehicles around obstacles [2], none has achieved wide acceptance in industry [3] [4]. The problem addressed in this paper is finding a smooth path around obstacles which can be followed exactly by a vehicle with car-like dynamics.

It is important to consider the variety of solutions which have been developed already in this area. A range of techniques for motion planning are well described with examples in [5] including graph search methods such as Prob-

abilistic Roadmaps with Dijkstra in addition to incremental search methods like Rapidly Exploring Random Trees. [6] goes into more detail for only those techniques suitable for on-road autonomous vehicles and [2] gives optimality and completeness results for many in a handy comparison table. Recent developments have used gradient descent to modify Bezier curves based on obstacle keypoints [7], evaluated alternative clothoid tentacles [8] and found the parameters of interpolating clothoids as an optimisation [9]. The benefits of clothoids for controlling lateral acceleration identified in [10] can be exploited by unoccupied vehicles, which can travel faster on smooth paths without lateral instability [11][12].

Path planning techniques can be divided into spatial sampling based and continuous methods. Sampling based planning algorithms operate on a discretization of the state space. Into this category fall roadmap planners such as [13] where the graph resulting from discretization is reused multiple times. The graph for adaptive paths is likely to be used only once as the obstacle field is likely to change based on recent sensor data. Roadmap planners are still extremely useful at the global scale where local sensor updates are less relevant. One result is the split ar-

chitecture described by [2] where a roadmap planner is used for strategic planning (planning over a longer time scale like a few minutes, extending from one part of the site to the other) and a different approach is used for tactical planning (creation of a detailed trajectory for the next few seconds). [14] describe this approach, using different techniques for the two time-scales as integration planning.

There are other sampling based methods which can also be useful for path adaptation where local sensor updates are important such as Dense Random Trees as described in [15] and [16] where the discretization is performed as the search proceeds. Sampling is frequently used to make problems of high dimensionality feasible, but can only offer resolution completeness. This is the guarantee that if a solution exists at the sampling resolution, then it will be found. Sampling from configuration space can result in paths which must be smoothed before they are traversable, so the most relevant techniques are based on sampling from the control space, or using parametrized curves so that every sample is feasible as in [17]. A frequent issue with dense random tree sampling methods is the introduction of artefacts in the solution which are difficult to remove by post-processing.

By contrast, the family of solutions based on numerical optimisation which operate directly on the continuous state space offer improved path quality and guarantees. These methods can be divided into parametric formulations which describe the path as some type of curve such as a polynomial [18] and those where the path is represented by a series of time samples which satisfy the differential constraints such as Timed Elastic Bands [19]. Compared to parametric methods, those optimising over a series of samples must search a much greater number of variables and also account for more constraints. This leads to additional computational burden, so they are often limited to a short time horizon and make use of a reference path to linearise obstacle constraints as the tactical planner in [20] which also uses output constraints to turn overtaking into a convex problem. For longer paths which can be stored in limited memory and reused parametric methods may be preferable, provided they are able to represent the dynamic limitations of the AGV.

Path representations which are suitable for the dynamic constraints of car-like vehicles can be based on different types of spline. Splines which are Cartesian can be calculated conveniently but they are only traversable if polynomial terms up to 5th order are included [21]. Polar splines have a smoothly varying curvature at first order and above but they are unable to represent a straight line they must be mixed with other curve types to form a complete path. Other curve types such as Bezier curves exist but one representation which is particularly suited to industrial AGV roadmaps is the clothoid curve or Euler spiral [22]. Using this parametrization, and constraining peak curvature and sharpness, the resulting path will be

feasible for a car-like vehicle at non-zero speed. The importance of sharpness limitation is sometimes overlooked but this is a real physical limit on the motion of a vehicle. This is because the sharpness is proportional to the angular acceleration at a constant traversal speed. Previous work on finding clothoid based paths around obstacles has mostly used spatial discretization to generate a series of points between the origin and destination, followed by curve fitting to find the clothoid segments which best fit to the points. This is a practical solution and variants of it are used by [23] who generates the key-points from a sequence of position samples from a manual drive, and [24] who fits to a series of predefined manoeuvres: u-turn, lane change and so on. As with other sampling based methods, the choice of sample points affects the final solution, leading to suboptimal solutions. This was made clear in [25] where sampling and curve fitting was compared to a direct optimisation method.

An early method for creating continuous curvature paths based on clothoids, arcs and straight lines was the CC-Steer algorithm [26]. This local planner could be used to connect samples from configuration space to create a probabilistic road map. A similar algorithm from around the same time from [27] was also able to create CC-paths without using clothoids by considering curvature continuity while approximating a holonomic path with a heuristic exploiting the differential flatness of car pulling trailers. Differential flatness of a dynamical system indicates the prior states can be determined from the current state with no exogenous variables [28]. Fraichard et al [26] showed CC-Steer approximated Reeds-Shepp [29] paths which are provably the shortest for connecting points with heading continuity. Using the maximum sharpness and maximum curvature to produce the shortest path is fundamental to the operation of CC-Steer, but this is not the only important objective. Often it is preferable to minimize the sharpness of turns in order to reduce lateral forces and maintain high speed.

The contribution of Henrie and Wilde [24] was to describe an algorithm to join two configurations with the least maximum curvature and least sharpness to create comfortable paths similar to those a human driver would follow. This used symmetric clothoid pairs to assemble paths with the same structure as [26]. One limitation of this work is that the clothoid pairs are always arranged symmetrically which limits the range of manoeuvres. This was addressed by the bisection method proposed by [23] which performs a numerical search to find either two or four clothoid segments, which are not required to be symmetrical, only matched so that the curvature at the end is zero. Another contribution of [23] is to test one approach for creating smooth global plans by fitting arcs and lines and clothoids to a series of samples from a GPS trace of a test vehicle driven by remote control. [30] goes into further detail in an Appendix B regarding the reachability of

clothoid pairs, but does not examine the geometric limits discussed in Section 2.2. The search procedure is fast at finding the parameters which meet the constraints but no algorithm is given for the correct choice of points to interpolate given a set of obstacles.

Recent works searching for paths with limited curvature rate such as [31] who used a superset of clothoids to find the best approximation to a holonomic path with limited curvature and provide a tuning parameter that could be tweaked to avoid obstacles. [32] also approximated holonomic paths but trained a neural network to speed up generation of the initial parameter guess. A double continuous curvature (DCC) path planner is a component in the path tracker described in [12]. The shortest DCC path from the current pose to the global reference path is found every control cycle by Nelder-Mead without considering obstacles. Silva et al [33] found a compact representation of a smooth road centre line consisting of arcs, lines and clothoids using existing methods to join certain key poses e.g.[34], depending whether they would track a roundabout or a straight road. Others such as [35] used a smooth road centre line as a reference to linearise the obstacle constraints and then sequentially minimized maximum curvature and sharpness in addition to deviation from the reference path.

In this paper we present a numerical optimisation formulation which can be used to find a clothoid spline which reaches an arbitrary goal through a series of convex obstacle-free regions. This is distinct from earlier works such as [9] as in this paper only the start and the goal poses are fixed, providing freedom to improve the objective within the natural constraints of obstacle polygons rather than being tied to heuristically selected waypoints. It can also be trivially adapted for point-to-point curve fitting similar to that of [23] but with the resulting path minimizing an arbitrary cost function over its length as described in Section 4.4. The new method is able to take into account constraints arising from obstacles directly, and finds a result using highly optimised off-the-shelf non-linear optimisation algorithms: Interior-Point Method and Sequential-Quadratic-Programming were tested. Rather than simulate a path following controller with a specific dynamic model, we reproduce results from an existing root finding method for the smoothest (least sharp) path [23] where the clothoid curve outperformed other primitives. The solution presented improves on current methods with a way to find an optimal path directly from the obstacle representation without introducing sampling bias by selecting waypoints. Constraints and soft objectives are separated and a weighting parameter introduced which allows control over the trade off between minimum sharpness and path length. This makes adaptive clothoid paths useful for a multi-AGV site which must have predictable behaviour and high efficiency.

2. MATHEMATICAL REPRESENTATION AND PROBLEM DEFINITION

Clothoid curves are widely used and appreciated for creating smooth drivable paths with limited angular acceleration. They are infrequently within a numerical optimisation framework, instead many heuristic methods for calculating their parameters have been developed. In what follows we look at the properties of clothoids which are relevant to numerical optimisation.

2.1. One Clothoid Segment

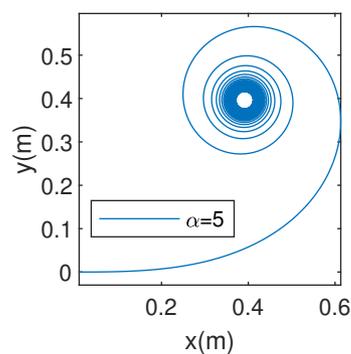


Fig. 1. Clothoid spiral with $\alpha = 5$ for $s > 0$

The clothoid is defined as a curve whose curvature κ increases linearly along its length. The rate of increase is called the sharpness $\alpha = d\kappa/ds$. Points on such a curve are defined by two parameters; the arc length s and the sharpness α . These two parameters describe a curve spiralling out of the origin along the x axis towards one asymptote with infinite positive curvature (turning anti-clockwise) in the positive $x - y$ quadrant for $s > 0, \alpha > 0$, and towards a negative asymptote with infinite negative curvature in the negative $x - y$ quadrant where heading is decreasing (turning clockwise) for $s < 0$ or $\alpha < 0$. The negative distance curve is just a reflection of the positive one and all arcs can be joined by appropriate rotation and translation so we may fix $s \geq 0$. The positive part of the curve with sharpness $\alpha = 5$ is shown in Figure 1. It can be evaluated in Cartesian coordinates with the Fresnel integrals which are reproduced in Equations 3-4. The change in angle over one segment is the deflection δ .

$$\kappa = \kappa_0 + \alpha s \quad (1)$$

$$\delta = \int_0^s (\kappa_0 + \alpha u) du = \kappa_0 s + \frac{\alpha s^2}{2} \quad (2)$$

$$x = C(\alpha, s, \kappa_0) = \int_0^s \cos(\kappa_0 + \alpha u) du \quad (3)$$

$$y = S(\alpha, s, \kappa_0) = \int_0^s \sin(\kappa_0 + \alpha u) du \quad (4)$$

The symbol ψ will be used for resultant heading angle after a number of segments. Whereas each δ increases unbounded, ψ is an angle measured clockwise from positive x direction and may be wrapped in the range $[0, 2\pi]$ without consequence. The configuration of a rigid body in a 2D plane with components $[x, y, \psi]$ will be referred to as a pose. See Appendix 2 for more details. The range of poses reachable by varying the parameters of a single segment are limited. Any x and y position can be reached by choice of parameters α and s - this makes intuitive sense as the parameter space is two dimensional as are the constraints. In order to meet heading ψ and curvature κ constraints as well, a spline composed of multiple clothoid segments is needed. Note that a clothoid segment with $\alpha = 0$ will form either a straight line of length s if the initial curvature is zero or an arc of length s otherwise.

2.2. Required number of Segments for Interpolation

First we consider G^2 interpolation with clothoids of fitting the minimum number of clothoid segments to a series of points with fixed $[x, y, \psi, \kappa]$ as addressed in [9][36]. This is helpful to understand the way the required number of clothoid segments varies depending on the constraints applied. As we intend to optimise some objective function of the curve it is required that the solution is underdetermined, that numerous feasible solutions to the interpolation problem exist allowing us to search over them to find the best.

G^2 continuity is needed for a smooth path which is traversable for a car-like vehicle [23] [24]. Its importance for fork lift operation is detailed in [12]. As explained in Section 2.1 there are two additional degrees of freedom available for each clothoid segment included in a G^2 -continuous spline. The requirement to end on a specified point with $[x, y, \psi, \kappa]$ provides four constraints. This implies that there is a unique solution for two segments per point as this gives rise to four parameters and four constraints.

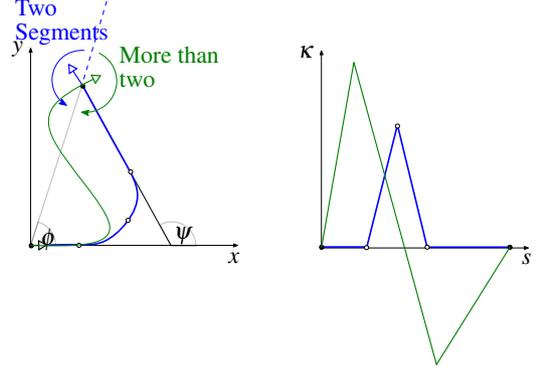


Fig. 2. ϕ is the lower limit on heading ψ at position x, y reachable from the origin with the blue curve without adding a third clothoid segment to create an S shape as shown in green. Corresponding curvature κ arc length s trace is plotted alongside.

In order to connect a clothoid segment with a straight line keeping G^2 continuity, the clothoid must have zero curvature at the point of connection. A clothoid pair with zero curvature at the end can be described as 'matched': The curvature change $\kappa = \alpha \cdot L$ over each segment is equal and opposite. Such a pair is shown in Figure 2 alongside a curvature-arc-length plot for illustration. Gim et al[23] show that the minimum reachable heading with two matched clothoids is given by Equation 5.

$$\psi > \phi = \arctan\left(\frac{y}{x}\right) \quad (5)$$

This is because an s-shaped curve cannot be formed with only two segments as illustrated in Figure 2. In [23] smaller angles are addressed with a second algorithm which computes two matched clothoid pairs (four segments total) instead. The advantage of the method is that the clothoids do not have to be symmetrical, so the final position can be reached with a lower sharpness and peak curvature as well as a shorter path length in some cases compared to using symmetric clothoid pairs.

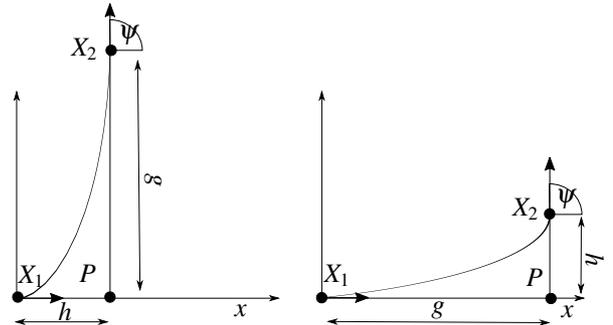


Fig. 3. The two limiting positions for an unsymmetrical clothoid pair without line segments reaching an angle ψ according to the g/h condition in Equation 6. For $\psi = 90$, $\frac{I_C(\psi)}{I_S(\psi)} = 1.7749$

The condition for the existence of an unsymmetrical clothoid pair to fit three non-collinear points is given by Equation 6 reproduced from [37].

$$\frac{g}{h} + \frac{\cos(\psi)}{\sin(\psi)} < \frac{I_C(\psi)}{I_S(\psi)} \quad (6)$$

The Fresnel Integrals defining the Cartesian position of a clothoid given in Equations 3 and 4 in Section 2.1 can equivalently be expressed as integrals over deflection δ as defined in Equation 2.

$$x = a \cdot I_C(\delta) = a \cdot \int_0^\delta \frac{\cos u}{\sqrt{u}} du \quad (7)$$

$$y = a \cdot I_S(\delta) = a \cdot \int_0^\delta \frac{\sin u}{\sqrt{u}} du \quad (8)$$

Equation 6 gives a maximum ratio of g/h in terms of the final angle ψ . The length g is always the distance to the furthest point from the intersection P and h is the distance to the closer point, as shown in Figure 3. As the condition is given in terms of points rather than poses, the requirement that the points not be collinear is equivalent to Equation 5 because $\psi = \phi$ would indicate collinearity and $\psi < \phi$ would result in the opposite initial heading given by vector $X_1 - P$. In order to avoid very short, high sharpness segments, straight line segments can be introduced before or after the clothoid pair as suggested in [37]. This allows unequal cases beyond the limits shown in Figure 3 to be fitted.

Using six parameters to describe two matched clothoids with zero curvature at the beginning and end permits the inclusion of a straight line at either end. The parameters would be two for each clothoid segment (total four) and two more for the length of the straight line at either end. The paths created for automated driving by [23] use a matched clothoid pair (two segments) for any corner. With this number of segments, there are certain unreachable poses as an s-shape cannot be formed within one region. The limited flexibility of these paths should be sufficient for executing a turn and returning to zero curvature in each region.

2.3. Obstacle Field Representation

Any field of polygonal obstacles can be equivalently represented as a set of possibly overlapping convex regions of free space [5]. The steps of a path planning algorithm utilising convex regions can be divided into the following steps:

1. Spanning. Convert obstacle representation into a small number of possibly overlapping convex regions which span the free space.
2. Assignment. Assign Path segments to a sequence of connected regions between the region containing the start to the region containing the goal

3. Curve Fitting. Solve for the best path from the start to the goal which remains within this sequence over its entire length

The ‘Spanning’ problem involves calculating a minimum number of spanning regions and is an NP hard problem in itself. A common representation of an obstacle field constructed from range data is an occupancy grid [38]. This consists of a 2D array of cells and can be created from uncertain range measurements from vehicles which are able to estimate their own position [39]. Each cell represents an area of the floor, with a number $p \in [0, 1]$ indicating the probability it contains an obstacle. A threshold can be used to create a binary map of occupied and unoccupied cells. The coarse obstacle-free region sets used in the numerical examples can be generated using the vertical decomposition method [40]. This begins with piecewise linear polygons, which can be created by connecting the cell corners of a binary occupancy grid. More complex environments and cluttered obstacle fields could be addressed using Iterative Region Inflation by Semi-definite Programming as described in [41].

The advantage of the regions being convex is that they can be solved guaranteeing optimality (that no other parameter set can minimize the objective further) and completeness (that if a path exists it will certainly be found). Given that the polygonal representation is only an approximation to the real obstacle field, it can only offer completeness within the limits of the resolution used to represent the obstacles but this provides more freedom than a series of pose samples chosen heuristically and may permit lower cost solutions in some cases.

Deits and Tedrake [18] use a mixed integer formulation which addresses ‘Assignment’ and ‘Curve Fitting’ simultaneously to achieve optimal results. ‘Assignment’ can also be approached as a graph problem, where there is a node for each region. Edges are inserted between regions which are connected. Connected regions can be found easily in configuration space as the vehicle is reduced to a point and therefore if any corner of a region is contained within any other, it is certain one is reachable from the other and they should be connected in the graph. Using the straight line distance between region corners to give an edge weighting, results in an approximation of the shortest global path. In this way the topology can be solved separately, compensating for one weakness of numerical optimisation, the ‘topological blindness’ identified in [42].

2.4. Test Environment 1

The first test environment was created for a hypothetical item fetch AGV with a payload of 50kg and a top speed of 1m/s.

A simple region shape for exposition is an axis aligned rectangle extending from x_{min} to x_{max} and from y_{min} to y_{max} . Regions for the local avoidance problem are shown in Fig-

ure 4. We assume the assignment is available between N regions and N clothoid pairs. Referring to Figure 4, the first pair will be assigned to the blue region, the second pair to the red region and the last pair to the amber region.

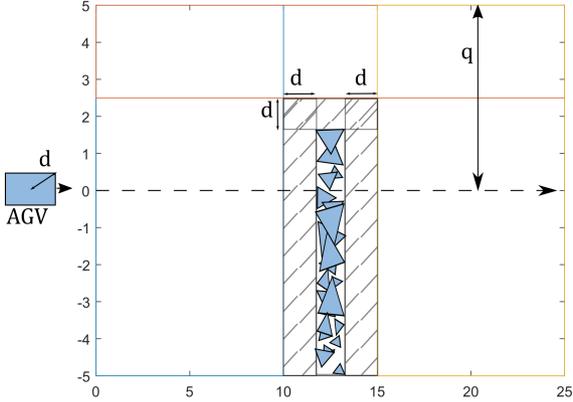


Fig. 4. Example of an obstacle field represented as a set of possibly overlapping empty convex regions. The reference path is a straight dashed line along the x-axis (to the right). The largest dimension of the vehicle body used to expand the obstacles d and the maximum deviation from the path is q .

2.5. Test Environment 2

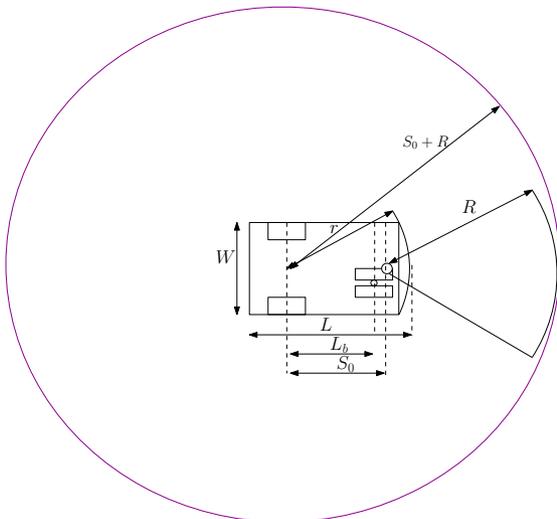


Fig. 5. Dimensions used to expand the obstacles

Table 1. Dimensions from datasheet [43]. *Stopping distance R based on top speed 3.22m/s and hypothetical braking deceleration of 4m/s^2

Parameter	Dimension (mm)
W	1067
L	1583
r	1289
L_s	1001
S_0^*	1200
R^*	1300

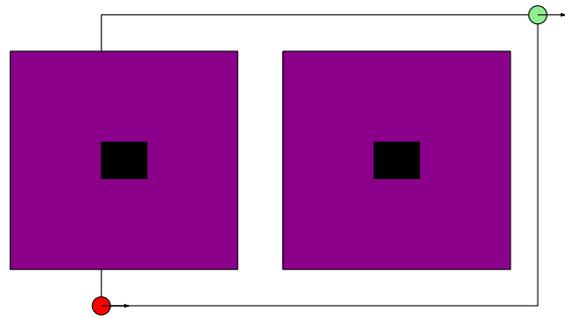


Fig. 6. Pallet environment. Obstacles are black with expansion by the vehicle disk shown in purple.

The second test environment was created for an automated fork lift AGV research platform which is taken to be representative [44]. The obstacles are based on $(1.0 \times 1.2)\text{m}$ pallets which are commonly used in the UK and Netherlands [45]. The datasheet for the manually operated vehicle on which the AGV is based, a Hyster E30-40HSD [43] gives the dimensions in Table 1. The plan of the vehicle is given in Figure 5.

The datasheet gives the maximum speed as 7.2 miles per hour (3.22m/s). Traction is provided by dual 4.8kW motors. The unloaded weight of the vehicle is 3059kg and the battery 1043kg for a total of 4201kg [43].

For correct operation it is important to consider the exclusion zone of the safety rated range sensor fitted to the front of the vehicle. If an obstacle breaches the exclusion zone the AGV must perform an emergence stop, or in some cases slow down significantly. To avoid slowing down the path planning must account for not only the shape of the vehicle but also the shape of this zone. Often this is a cuboid slightly wider than the vehicle, sufficiently long that the AGV can come to a complete stop from full speed before the front makes contact with a static obstacle. More details are available in the NIST Safety Standards [46].

In the two simulated environments the obstacle field is represented in 2D. The bounding circle dimension is strongly influenced by the stopping distance R . Starting

from the constant acceleration equation $v^2 = u^2 + 2as$ and setting $v = 0$ gives the stopping distance $s = \frac{u^2}{-2a}$.

2.6. Problem Definition: Clothoid Fitting to Convex Regions

The core problem is G^2 continuous path planning around obstacles for a point robot. Using a convex spanning region approach described in Section 2.3 this can be divided into a small number of sub-problems. This paper addresses the ‘Curve Fitting’ sub-problem in detail using the path representation set out in Section 2.2. The approach is described in Section 3.

The requirement for G^2 (curvature) continuity provides four constraints on each segment similar to the posture interpolation problem [9]. In each region we permit one path piece consisting of a line segment length $s_{0,i}$, a matched pair of clothoid segments defined by $\alpha_{1,i}, \alpha_{2,i}, L_{1,i}, L_{2,i}$ and the other line segment length $s_{F,i}$, as explained in Section 2.2. This provides a total of six free parameters per path piece. As there are only four constraints per region the problem is under-determined and the length of the straight lines can be balanced against the length of the clothoids to find the ideal combination. We consider the start and end region to always be separate even if they overlap, so there will be a minimum of four clothoid segments across the whole spline and s-shapes are feasible.

3. METHOD: CLOTHOID FITTING TO CONVEX REGIONS

The core routine is a method to find the parameters of a sequence of G^2 -continuous (continuous in curvature) clothoid segments which is contained entirely within a chain of obstacle-free regions. We minimize the weighted squared sum of path length and sharpness, over a sequence of N path pieces assigned to N convex regions. The objective function is designed to trade off smoothness with path length such that the best path is one that minimizes the weighted squared sum of sharpness and path length to reach the x , y , ψ and κ of the reference path. This is captured in Equation 9 below:

$$\begin{aligned} \min_{\alpha, \mathbf{L}, s_0, s_F} J_T &= b \cdot \alpha^T \alpha + \mathbf{L}^T \mathbf{L} + s_0^T s_0 + s_F^T s_F \\ \text{such that } \begin{cases} c_{eq}(\alpha, \mathbf{L}, s_0, s_F) = \mathbf{0} \\ c(\alpha, \mathbf{L}, s_0, s_F) \leq \mathbf{0} \end{cases} & \quad (9) \\ \text{with bounds } \mathbf{L} \geq \mathbf{0}, s_0 \geq \mathbf{0}, s_F \geq \mathbf{0} & \end{aligned}$$

Bold variables indicate vectors of the parameters defined in Section 2.1 for every path piece. Scaling parameter b is discussed in Section 3.1. The equality constraints arise from enforcing continuity between each piece and of the first and last segments with the origin and destination respectively. The inequality constraints relate to the requirement to remain outside every obstacle. The first deriva-

tives of the objective and constraints are given in Appendix 1.

3.1. Objective Function

Two important objectives for an alternative path for an AGV are the total length and the peak sharpness. Both of these properties are desirable rather than mission critical so they are useful in resolving the many solutions which are available to reach a given pose smoothly. Other performance measures such as reaching the destination exactly, limiting the peak curvature and avoiding the obstacles are better interpreted as hard constraints, as it is not useful to compromise them in any way for the measures which are only desirable.

The degree to which smoothness is important compared to path length may depend on the application so it is left with a scaling parameter b . The units of b are m^2 per $\text{radian}^2/\text{m}^4 = \text{m}^6 \cdot \text{radian}^{-2}$. If b is set very high, smoother paths can be attained, at the cost of increasing path length. In general there is a trade off between path length and smoothness (measured by the least maximum rate of change of curvature as identified by [24]).

Results will be reported with three alternative parameters settings for b , with and without the straight lines, as described below.

- Equal Weighting with Lines, $b = 1$, $\min_{\alpha_i, L_i, s_0, s_F} J_T = \sum_i^N L_i^2 + \sum_i^N \alpha_i^2$
- Equal Weighting no Lines, $b = 1$, $s_0 = 0$, $s_F = 0$, $\min_{\alpha_i, L_i} J_T = \sum_i^N L_i^2 + \sum_i^N \alpha_i^2$
- Minimum Sharpness, no Lines, $b = \text{inf}$, $s_0 = 0$, $s_F = 0$, $\min_{\alpha_i} J_T = \sum_i^N \alpha_i^2$

3.2. Equality Constraints for Region i

The continuity constraint is to reach the next point given by $[\hat{x}, \hat{y}, \hat{\psi}, 0]^T$. A straight line s_0 before and s_F after the clothoid pair is included while maintaining curvature continuity by the constraint $\kappa = 0$ which forces the clothoid pair to be matched, smoothly returning the curvature to zero at the end.

For multiple regions, the continuity constraints are applied implicitly by integrating each segment starting from the final pose of the last. As the curvature is zero at the end of each pair, this can be done by integrating pairwise from the origin as in Equation 10 followed by a rotation and translation to the final pose of the last segment using the \oplus operator detailed in Appendix 2. This is an example of single shooting trajectory optimisation as described in [47]. Each region gives rise to one additional $\kappa = 0$ constraint, to ensure the straight lines $s_{0,i}$ and $s_{F,i}$ can be added

with curvature continuity.

$$\begin{bmatrix} x_i & s_{0,i} + C(\alpha_{1,i}, L_{1,i}, 0) \\ & + \cos(\delta_{1,i}) \cdot C(\alpha_{2,i}, L_{2,i}, k_{m,i}) \\ & - \sin(\delta_{1,i}) \cdot S(\alpha_{2,i}, L_{2,i}, k_{m,i}) \\ & + s_{F,i} \cdot \cos \hat{\psi}_i \\ y_i & \sin(\delta_{1,i}) \cdot C(\alpha_{2,i}, L_{2,i}, k_{m,i}) \\ & + \cos(\delta_{1,i}) \cdot S(\alpha_{2,i}, L_{2,i}, k_{m,i}) \\ & + s_{F,i} \cdot \sin \hat{\psi}_i \\ \psi_i & \alpha_{1,i} L_{1,i}^2 / 2 + \alpha_{2,i} L_{2,i}^2 / 2 \\ \kappa_i & \alpha_{1,i} L_{1,i} + \alpha_{2,i} L_{2,i} \end{bmatrix} \quad (10)$$

The path in local coordinates for one piece given by Equation 10 can be composed using the \oplus operator to find the final position in global coordinates.

$$[X_i, Y_i, \Psi_i]^T = [X_{i-1}, Y_{i-1}, \Psi_{i-1}]^T \oplus [x_i, y_i, \psi_i]^T, \forall i \in [1, N]$$

where capital letters indicate global coordinates. The first region is given the index $i = 1$. $[X_0, Y_0, \Psi_0]$ indicates the starting position. This is taken to be the origin of the coordinate system.

3.3. Equality Constraints Vector

Subtracting the goal pose from the N th pose in global coordinates gives the equality constraints, where N is the total number of convex regions under consideration.

$$\mathbf{c}_{eq} = \begin{bmatrix} X_N - \hat{X} \\ Y_N - \hat{Y} \\ \Psi_N - \hat{\Psi} \\ \kappa_1 - 0 \\ \vdots \\ \kappa_N - 0 \end{bmatrix} = \mathbf{0} \quad (11)$$

Note that the \mathbf{c}_{eq} array is length $N + 3$ as there is a constraint on curvature at the end of each clothoid pair κ_i . The other quantities are scalar and refer to the pose of the final segment at the end of the spline.

3.4. Inequality Constraints

In order to operate on the obstacle field directly and avoid spatial sampling, inequality constraints are used to fix the segments inside their assigned regions. The assignment of curve sections to regions will not be discussed here, but can be approached as an integer problem to find the boolean assignment matrix $\mathbf{H}^{(N \times R)}$ which results in the lowest cost solution. R is the number of regions in the map, and N is the number of pieces of the spline between the start and the goal. A simple region shape for exposition is an axis aligned rectangle extending from X_{min} to X_{max} and from Y_{min} to Y_{max} . This leads to the following eight inequality constraints on every region, ensuring that both the start and end of each piece are contained.

Based on the assignment $\mathbf{H}(i, j)$, indicating path piece i must be entirely contained in region subscript j . The constraints for one region are given by

$$\mathbf{H}(i, j) = 1 \iff \mathbf{d}_i = \begin{bmatrix} X_i - X_{max,j} \\ -X_i + X_{min,j} \\ Y_i - Y_{max,j} \\ -Y_i + Y_{min,j} \\ X_{i-1} - X_{max,j} \\ -X_{i-1} + X_{min,j} \\ Y_{i-1} - Y_{max,j} \\ -Y_{i-1} + Y_{min,j} \end{bmatrix} \leq \mathbf{0} \quad (12)$$

These inequality constraints ensure the start and end of the curve piece i assigned to region j remain inside the region. The position at the start of curve i is identical to the position at the end of curve $i - 1$ by the application of the continuity constraints. For the first curve the start is fixed at the origin by the choice of coordinate frame. The inequality confirms that the initial position must be in a region of free space for a solution to exist.

The constraints for the entire problem can be constructed by stacking \mathbf{d}_i for each region into a partition vector as follows

$$\mathbf{c}_{ineq} = \begin{bmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_N \end{bmatrix} \leq \mathbf{0} \quad (13)$$

Constraint Equation 12 only applies to the start and end of each clothoid pair, not the entire curve. This works quite well when the curvature remains low but will cause a problem for certain combinations of region shape and path curvature as discussed in Section 4.2.

3.5. Multiple Shooting Formulation

Alternatively the problem can be posed as a multiple shooting trajectory optimisation by the terminology of [47]. The suggestion is that although there are more numerous parameters and more constraints on the multiple-shooting problem, it may counter-intuitively be easier for the solver because each parameter-constraint pair is more independent and closer to linear. This involves extra parameters $X_{0,i}, Y_{0,i}, \Psi_{0,i}$ which provide a pose offset for each clothoid pair and explicit continuity constraints between each clothoid pair and the last. The problem would then be described as

$$\begin{aligned} & \min_{\alpha, \mathbf{L}, \mathbf{s}_0, \mathbf{s}_F, X_0, Y_0, \Psi_0} J_T = b \cdot \alpha^T \alpha + \mathbf{L}^T \mathbf{L} + \mathbf{s}_0^T \mathbf{s}_0 + \mathbf{s}_F^T \mathbf{s}_F \\ & \text{subject to } \begin{cases} \tilde{\mathbf{c}}_{eq}(\alpha, \mathbf{L}, \mathbf{s}_0, \mathbf{s}_F, X_0, Y_0, \Psi_0) = \mathbf{0} \\ \tilde{\mathbf{c}}_{ineq}(\alpha, \mathbf{L}, \mathbf{s}_0, \mathbf{s}_F, X_0, Y_0, \Psi_0) \leq \mathbf{0} \end{cases} \\ & \text{with bounds } \mathbf{L} \geq \mathbf{0}, \mathbf{s}_0 \geq \mathbf{0}, \mathbf{s}_F \geq \mathbf{0} \end{aligned} \quad (14)$$

Now the coordinates are calculated slightly differently, using the new offset parameters rather than a recurrence

relation.

$$[X_i, Y_i, \Psi_i]^T = [X_{0,i-1}, Y_{0,i-1}, \Psi_{0,i-1}]^T \oplus [x_i, y_i, \psi_i]^T, \forall i \in [1, N]$$

where capital letters indicate global coordinates.

Subtracting the final pose and curvature of the previous pair from the initial pose each pair, with zero curvature gives the equality constraints for that region arising from continuity

$$\tilde{\mathbf{q}}_i = \begin{bmatrix} X_{0,i} - X_{i-1} \\ Y_{0,i} - Y_{i-1} \\ \Psi_{0,i} - \Psi_{i-1} \\ \kappa_{0,i} - 0 \end{bmatrix} = \mathbf{0} \quad (15)$$

There is also a constraint with the goal pose similar to the single shooting form, which must be included in the stack.

$$\tilde{\mathbf{q}}_{goal} = \begin{bmatrix} X_N - \hat{X} \\ Y_N - \hat{Y} \\ \Psi_N - \hat{\Psi} \\ \kappa_N - 0 \end{bmatrix} = \mathbf{0} \quad (16)$$

Here we use the first subscript of $X_{0,i}$ to indicate it is the X offset for the region given by the second subscript. The offsets are new search parameters for this formulation. Symbols with a single subscript, the subscript identifies the region and X_i refers to that at the end of the curve assigned to that region. Again the first region is given the index $i = 1$. The start pose is $[X_0, Y_0, \Psi_0]$. This is taken to be the origin of the coordinate system.

The inequality constraints for the entire problem can be constructed by stacking $\tilde{\mathbf{k}}_i$ for each region into a partition vector as follows

$$\tilde{\mathbf{c}}_{ineq} = \begin{bmatrix} \tilde{\mathbf{q}}_1 \\ \vdots \\ \tilde{\mathbf{q}}_N \\ \tilde{\mathbf{q}}_{goal} \end{bmatrix} = \mathbf{0} \quad (17)$$

The inequality constraints for region i are given by

$$\mathbf{H}(i, j) = 1 \iff \tilde{\mathbf{d}}_i = \begin{bmatrix} X_i - X_{max,j} \\ -X_i + X_{min,j} \\ Y_i - Y_{max,j} \\ -Y_i + Y_{min,j} \\ X_{i-1} - X_{max,j} \\ -X_{i-1} + X_{min,j} \\ Y_{i-1} - Y_{max,j} \\ -Y_{i-1} + Y_{min,j} \end{bmatrix} \leq \mathbf{0} \quad (18)$$

The constraints for the entire problem can be constructed by stacking $\tilde{\mathbf{d}}_i$ for each region into a partition vector as follows

$$\tilde{\mathbf{c}}_{ineq} = \begin{bmatrix} \tilde{\mathbf{d}}_1 \\ \vdots \\ \tilde{\mathbf{d}}_N \end{bmatrix} \leq \mathbf{0} \quad (19)$$

4. NUMERICAL RESULTS

First, in Section 4.1 the suitability for finding the smoothest path subject to obstacle constraints is tested on an environment similar to Figure 4. Subsequently, an alternative formulation, the effect of analytical gradients, tuning parameter b and the objective function are evaluated for their potential in speeding up the solution. A number of tests without obstacles are included in Section 4.5 and 4.6 so path parameters can be compared with an existing heuristic method.

4.1. Motivating Problem

The tested problem concerns the avoidance of a small obstacle blocking a straight path while remaining within a set tolerance from it. The allowable distance from the original path is used to generate the outer boundary enclosing $[0, 5]$ and $[11, -5]$. With the obstacle information available from sensors this is broken up into three convex regions. As set out in Section 2.2 this necessitates a path with six clothoid segments, two at the start, two at the goal and two at an intermediate region needed to connect the two.

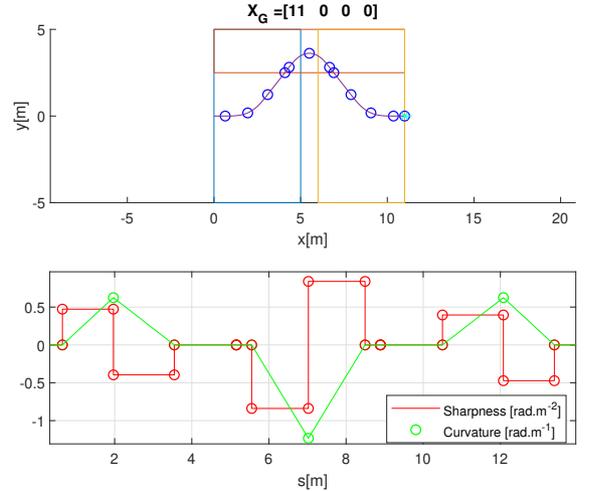


Fig. 7. Single shooting optimal avoidance path for a vehicle traveling along the x -axis which encounters a rectangular obstacle at $[5, -2.5]$, extending to $[6, 2.5]$, rejoining the reference path at $[11, 0]$. Weighting parameter $b = 1$

The single shooting formulation was solved using *interior-point*¹ method in 36 iterations to produce the path and curvature profile shown in Figure 7. The region constraints are satisfied. This can be seen as each of the four segments is marked with an open circle. The same points are marked with open circles in the curvature and sharpness profile shown below the x - y plot in Figure 7. These can be used to evaluate the quality of the path without reference to a specific vehicle model. The largest magnitude

curvature is found at the midpoint with a value of $-1.233 \text{ [m}^{-1}\text{]}$, corresponding to a turning radius of 811mm , suitable for a small vehicle with Ackermann steering, having a wheelbase less than 811mm . The largest magnitude sharpness of $0.8388 \text{ [radm}^{-2}\text{]}$ is seen on the segments before and after the peak curvature. This path can be tracked with high accuracy by different AGVs by reducing forward speed to control the lateral acceleration based on the path curvature, and the angular acceleration based on the sharpness.

The single shooting formulation reduces the number of parameters which might be expected to reduce total execution time. In fact, it took multiple seconds to reach convergence on the small $20\text{m} \times 10\text{m}$ environment made up of three regions tested.

4.2. Weighting Parameter b Effects

Fig. 8. Convergence time as weighting b between sharpness and length is varied

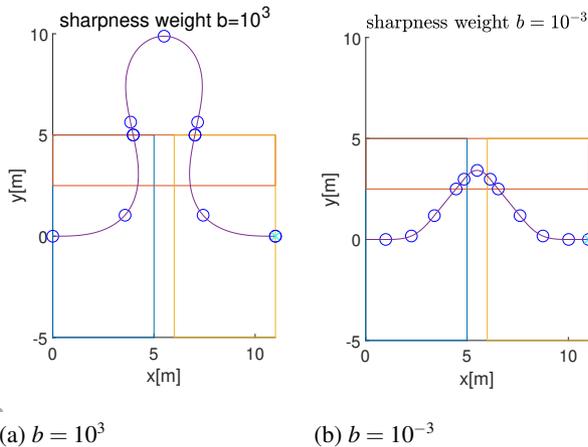
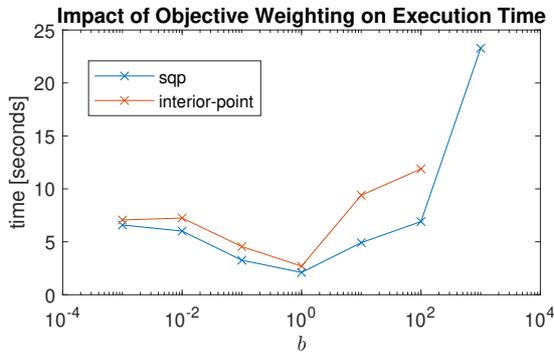


Fig. 9. Path comparison for extremes of b

The weighting parameter b has a significant effect on convergence time as shown in Figure 8. The number of

iterations and function evaluations increased for larger weights, in the same way as time to convergence.

The two path plots cover both extremes of weighting tested for the avoidance problem. Figure 9b shows the effect of a small weighting on sharpness. The path is shorter, the peak sharpness is higher, but it is not the shortest continuous path which would pass through the corner formed by region one and two (the top one). The segments are close to equal length because, for a set of numbers with a fixed sum, the sum of squares will be minimized if the numbers are equal. This provides a bias toward equal length segments and may contribute to the success of an equal weighting of the two components.

Figure 9a shows the effect of a large weighting on sharpness. The path is longer and more meandering but has lower peak sharpness. The constraints are met as the start and end are contained but the curve leaves the convex region at the top. Because the region constraint is not applied to the point of peak curvature, there are a large number of feasible solutions with similar sharpness. This makes an objective based on sharpness ($b \gg 1$) very flat close to the minimum with this set of obstacles. Looking through the text output of *fmincon*, a feasible solution according to the threshold of $1e-6$ is found earlier but the search continues until the threshold is reached. To solve this problem, the region constraints need to be applied to additional samples along the path.

4.3. Obstacle Avoidance Multiple Shooting Formulation

In the multiple shooting formulation by contrast, the curve positions in one region are independent of the parameters of the earlier segments due to the introduction of a new $[X_0, Y_0, \Psi_0]$ offset parameter for each region and a new constraint that this offset matches the final pose of the curve in the preceding region.

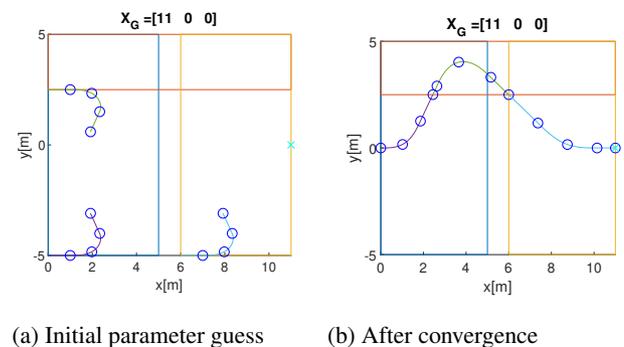


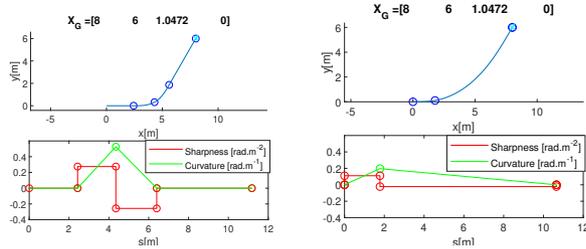
Fig. 10. Multiple shooting initial guess ($\alpha_i = 1.0$, $L_i = 1.0$) showing path continuity constraints are not met until the optimisation has converged

With the initial guess shown in Figure 10a the solver

must find offsets which satisfy both continuity and region containment. This can give better performance than single shooting in some cases involving longer splines [47]. In the problem shown in Figure 10b with three regions the discontinuities have been resolved after convergence, the execution time is comparable with single shooting, slightly improved, with around 700 function calls using interior point method.

This is despite increasing the number of parameters from six per region to nine per region (18 to 27 with $N = 3$), and the number of constraints from $4 + N$ equality and $8N$ inequality (7 and 24 with $N = 3$) to $4 + 4N$ equality and $8N$ inequality (16 and 24 with $N = 3$). The number of inequality constraints could be reduced slightly for the multiple shooting by fixing the start position at the world origin, to match the features of the single shooting setup.

4.4. Curve fitting With Two Clothoids With Different Objectives



(a) Equal Weighting with Lines (b) Minimum Sharpness, no Lines

Fig. 11. Side by side comparison of a path from the origin to $[8,6,60]$, curvature profile is shown below.

Table 2. Parameters identified for a path from $[0,0,0]$ to $[8,6,60]$. The objectives are explained in Section 3.1

	Equal Weighting, with Lines	Minimum Sharpness, no Lines
α_1 [m^{-2}]	0.1472	0.1094
α_2 [m^{-2}]	-0.1135	-0.0222
L_1 [m]	2.4893	1.7981
L_2 [m]	3.2281	8.8535
s_0 [m]	1.5966	0
s_F [m]	3.7390	0
Total Length [m]	11.0529	10.6516
κ_m [m^{-1}]	0.3633	0.1966
<i>interior-point</i> iterations	12	2000+
<i>interior-point</i> funcCount	101	16220+
<i>sqp</i> iterations	15	12
<i>sqp</i> funcCount	130	89

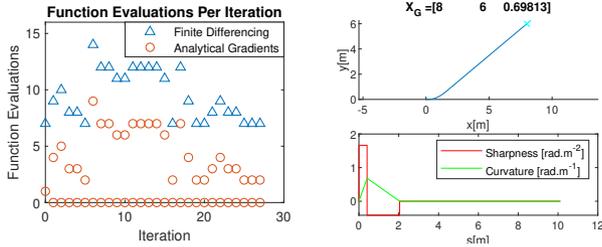
Cartesian path and curvature profile results for a single region containing two clothoid segments and two straight lines are shown in Figure 11a for comparison to those reported by [23] for two matched clothoids. For illustration the parameters identified for the same final pose by our method with two different cost functions are shown in Table 2.

There are two remarks to make here. Firstly, the $[8,6,60]$ point meets the condition in Equation 6, Section 2.2 and can be reached exactly with two clothoids and no line segments. Many nearby points require a line to be included at the end of the clothoid pair for convergence, particularly those with lower final angles (closer to the limit given by Equation 5). Secondly, the initial guess had to be very close to the minimum in order to reach convergence at all with *interior-point* method for this problem. Using the ‘Equal Weighting with Lines’ objective an initial guess of $[1, -1, 1, 1, 1, 1]$ converged in 12 iterations with *interior-point* method. With the same initialisation vector and the ‘Minimum Sharpness, no Lines’ objective, *interior-point* did not converge within 2000 iterations. If the solver was changed to *sqp* convergence took 12 iterations and 89 function evaluations. The *interior-point* method is shown to be slightly faster on some problems but less stable than *sqp* on the limited numerical tests performed. These tests only involved three regions, leading to 18 parameters and 16 constraints. It is a notable advantage of the convex region representation that the number of regions can be strictly limited even in environments with lots of clutter. The documentation lists *sqp* as a medium-scale algorithm, which needs to store and operate on matrices with the dimension of the parameters [48]. If problems are encountered in larger tests, *interior-point* is a large scale algorithm which does not rely on dense matrix operations, and should perform better. Due to the general constrained form of the problem other highly optimized methods for specific performance needs could be used depending on the needs of the application (e.g. limited processing time, limited memory, embedded platform. . .).

The ‘Minimum Sharpness, no Lines’ solution would be expected to be longer with lower peak sharpness than the ‘Equal Weighting, with Lines’ solution, as only reducing sharpness contributes to the objective. As expected Table 2 shows the peak sharpness is reduced from 0.1472 to 0.1094, around 30%. Surprisingly the total length of the ‘Minimum Sharpness, no Lines’ solution is also less than the ‘Equal Weighting, with Lines’. In this case *sqp* is stuck in a local minimum when additional straight lines are included. Line segments must be permitted for convergence on cases outside the boundary of the existence criteria in Section 2.2, but it seems that well within the boundary such as the end point in Figure 11a, considering line segments may lead to the introduction of local minima.

4.5. Curve fitting With Two Clothoids with Analytical Gradient

The impact of analytical gradients was tested on a single region containing two segments to the point $[8, 6, 40]$, close to the lower angle limit, as shown in Figure 12b. With an initial guess which did not meet the regions constraints $\mathbf{p} = [0.1, -0.1, 1, 1, 1, 1]$, *interior-point* method¹ failed to converge. The alternative algorithm *sqp*¹ converged in 27 iterations with or without derivatives as shown in Figure 12a. When using analytical derivatives the number of function evaluations was reduced from 264 total to 123 total but the total execution time increased from 1.33 seconds to 2.28 seconds². Therefore the mean execution time per function evaluation increased from $1.33/264 = 5\text{ms}$ to $2.28/123 = 18\text{ms}$ with the additional of numerical gradients.



(a) Both took 27 iterations so are(b) Both produced exactly the same path and curvature profile shown on the same figure

Fig. 12. Comparison of performance with and without analytical gradients. The number of iterations is high because the heading angle at the end is close to the lower limit given by Equation 5 for this position.

On a similar problem with a different goal heading, providing analytical derivatives reduced the function count reported by *fmincon* from 213 to 55 and the number of iterations to reach the same optimality threshold from 25 to 24. However the computation time increased by about 50% from 2.35 seconds to 3.00 seconds. If the solver was changed to *sqp*², the effect was similar. Slightly reduced count of function evaluations but an increased computation time. This may be due to the numerous integral terms in the expressions for the gradients taking more time to evaluate than the function at multiple locations to allow differencing. There are 22 integrals to evaluate in the Jacobian. Only two integrals are required to evaluate the objective function. Equations 3 - 4 were evaluated with Vectorized Adaptive Quadrature method [49] with a relative

¹*interior-point* and *sqp* are available as options for the *fmincon* function of MATLAB

²Numerical tests ran on a consumer laptop with 16GB RAM and an Intel(R) Core(TM) i5-8250U CPU @1.60GHz

tolerance³ of 10^{-6} . No attempt was made to store and reuse repeated terms in the Jacobian, although there are several.

4.6. Curve fitting With Four Clothoids With Different Objectives

In order to enable further comparison with the bisection method of Gim et al [23], another test was reproduced involving an s-shaped path comprising four clothoids to the pose $[12, 10, \theta]$, where θ varied in increments of 10 degrees. This shows clearly the trade off between path length and peak sharpness.

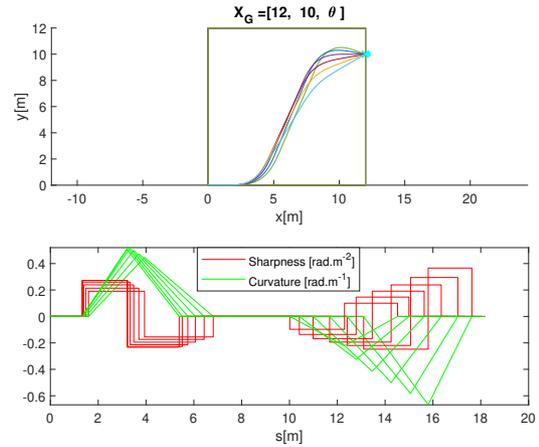


Fig. 13. Six lane change paths with objective 'Equal weighting, with lines', each ending at the same point $[12,10]$ with a heading θ separated by 10 degrees.

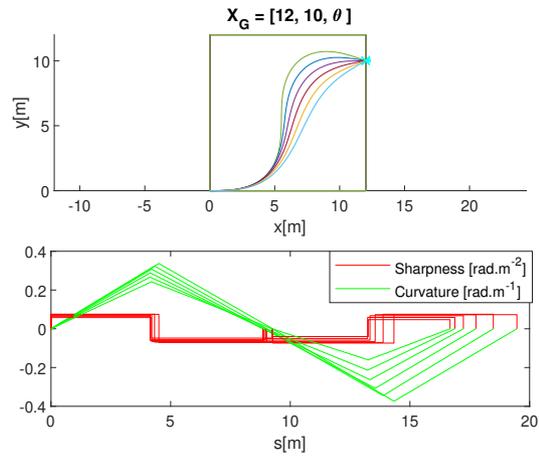


Fig. 14. Six lane change paths with objective 'Minimum sharpness, no lines', each ending at the same point $[12,10]$ with a heading θ separated by 10 degrees

³Vectorized Adaptive Quadrature is available in MATLAB as a built in function $q=integral(fun, xmin, xmax)$, the default upper bound on error is $10^{-6} \times q$

The ‘Equal Weighting, with Lines’ set of paths is shown in Figure 13. For the final angle of -30 degrees the key parameters of the solution are shown in Table 3. The cost is equivalent to penalizing path length with $b = 1[m^6 rad^{-2}]$ but because each segment is squared individually, the cost is lower if each segment is of similar length. The ‘Minimum Sharpness, no Lines’ set of paths for comparison is shown in Figure 14. For every tested heading the path is significantly longer than the ‘Equal Weighting, with Lines’ solution, while the peak sharpness is much lower. This makes intuitive sense because over longer distances the shortest path is a straight line. As the sharpness is increased the path comes to resemble the point to point line more closely.

Table 3. Parameters identified for a path from [0,0,0] to [12,10,-30]. This is one of the curves plotted in figure 13. The objectives are explained in Section 3.1 Units α [m⁻²], κ [m⁻¹], L [m]

	Equal Weighting with Lines	Equal Weighting, no Lines	Minimum Sharpness, no Lines
α_1	0.1707	0.0552	0.0727
α_2	-0.1404	-0.0443	-0.0709
α_3	-0.2808	-0.0894	-0.0745
α_4	0.4582	0.1716	0.0665
Peak Sharpness max $ \alpha_i $	0.4582	0.1716	0.0745
L_1	2.4569	4.8616	4.7639
L_2	2.9860	6.0572	4.8878
L_3	2.7117	5.4077	5.2704
L_4	1.6615	2.8180	5.967
s_{01}	1.7170	0	0
s_{F1}	3.2565	0	0
s_{02}	3.2565	0	0
s_{F2}	0.0893	0	0
Total Length	18.14	19.1445	20.8288
κ_{m1}	0.4142	0.2685	0.3464
κ_{m2}	-0.7557	-0.4837	-0.3928

Convergence with the ‘Equal Weighting, with Lines’ cost was particularly strong in the experiments attempted compared to either the more natural absolute sum of the segment lengths, all squared $J = (\sum_j |\alpha_j|)^2 + (\sum_j |L_j|)^2 + (\sum_j |s_{0,j}|)^2 + (\sum_j |s_{F,j}|)^2$ or the minimum sharpness with the line segments fixed at zero so $J = (\sum_j |\alpha_j|)^2$. Again the minimum sharpness approach produced paths almost identical to the bisection method proposed by [23] on the examples they reported.

The final position was taken from [23] so it can be used to compare the curvature profile and path trace to the one produced by the bisection method presented in that paper.

This is a root finding approach which is suitable for meeting the constraint on final position. No objective function is defined so the first solution which meets the constraint tolerance will be accepted. The ‘Minimum sharpness, no Lines’ path trace in Figure 14 looks very similar to the paths produced by the bisection method. The total length of both is around 20.8m. The coordinate system differences make side by side comparison a little challenging, as they used a start position of [0, 0, 90] ending at [10, 12, 120] which is the same path subject to some affine transformations. The curvature profile is comparable side-by-side and shows the sharpness of each section found by bisection to be close to 0.07125, very close to the average of the ‘Minimum Sharpness, no Lines’ column in Table 3.

The optimisation method has a clear advantage as it searches intermediate angles by continuously varying the length and sharpness of the first two clothoid segments. Bisection with four clothoids by contrast only searches very coarsely over intermediate angles (every 10 degrees). Searching more effectively should be a big advantage of using optimisation, but in this particular example the improvement is very small. The main benefit of the new method is the ability to take into account obstacle constraints at the start and the end and find a path purely from a polygonal representation of the obstacles.

A range of curves with θ varying from -30 degrees to 20 degrees is shown in Figure 13. The parameters can be compared to the same range of angles with a cost function which only penalizes the sum of squared sharpness of each segment and forces the straight lines at the start and end of each segment to zero in Table 3. The sharpness is reduced but the total path length is increased until it is almost identical to the curve plotted in [23]. The added value of using four segments over three is questionable as the middle two take almost the same value.

The expected trade off between peak sharpness and total path length can be seen in Table 3.

4.7. Effect of Small Changes to Region Boundaries

When the convex regions are constructed based on an occupancy grid as suggested in Section 2.3 measurement errors may lead to sudden changes in the size of the free space as the probability of one cell being occupied crosses the threshold. In this situation the size of the position shift would be determined by the cell size of the binary occupancy grid. The cell size is typically made larger than the sensor noise by some constant factor. A cell size of 100mm was selected to provide an ample boundary for a common sensor with zero mean error, 10mm standard deviation [50].

The offset error was varied in increments of 100mm, and the resulting change in objective function is shown in Figure 15. Both curve types increase linearly with the position error at first. The cubic objective is the second derivative of position, while the clothoid mini-

mized ‘Equal Weighting With Lines’ from Section 3.1 The clothoid curve objective increases more slowly until the final offset of 0.5m which causes a step change in the objective. This seems to be related to the lack of a constraint on the midpoint of the curve piece. As the obstacle is shifted the path must deflect more and in this case the midpoint leaves its assigned region. The problem of sensor errors pushing the path planner into a suboptimal local minimum could be significant. An additional constraint the midpoint should resolve it in this case, but for real world implementation some kind of safety checks will be needed before attempting to drive a new path for situations like this.

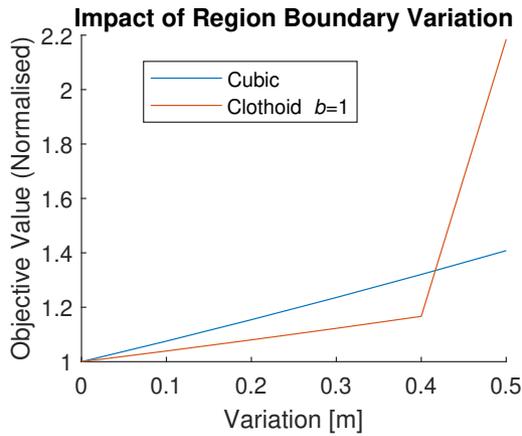


Fig. 15. Normalised objective function against a linear shift of the boundary in x and y

4.8. Curvature Comparison with Cubic Spline on Environment 2

Side by side comparison reveals the clothoid curve in Figure 16 has a higher peak curvature and is much smoother as measured by the peak sharpness. Cubic curves are simple to implement and often used for path planning. The cubic curve only has enough degrees of freedom to meet the heading constraint at the start and end by relaxing the constraint on second derivative to zero at the start and the end. The cubic curve in Figure 17 is continuous and the peak of 1.5m is feasible with a limited steering angle vehicle, however the curvature plot reveals rapid changes in curvature. The sharpness peaks at the start and end and is too large to be shown on the figure. This will result in the steer-drive wheel actuating rapidly and likely greater tracking error by an AGV attempting to follow it. Using a higher order polynomial could improve on this, but not to the extent of the clothoid curve which has piecewise constant sharpness and the magnitude is always less than $2[m^{-2}]$ so it can be tracked by a real AGV such as the one described in Section 2.5.

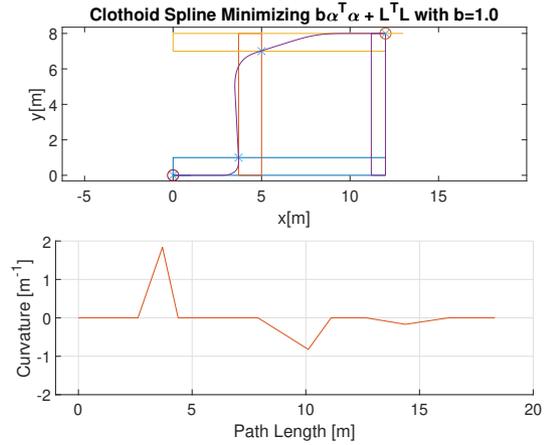


Fig. 16. Path Trace and Curvature for Clothoid Spline

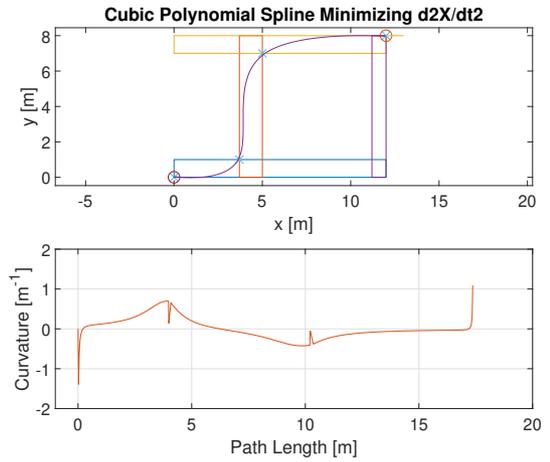


Fig. 17. Path Trace and Curvature for Cubic Polynomial Spline

5. CONCLUSION

The formulation presented in Section 3 is sufficient to solve the path planning with obstacle constraints using clothoid curve segments with G^2 continuity, using a generic constrained non-linear solver. The path with three regions converges reliably and it can be extended to more complex scenes. Execution time of several seconds indicates a need for further optimisation, but the path is equally smooth as the interpolation method of Gim et al [23], without the need for a sequence of points from manual driving or a heuristic for selection of waypoints based on the obstacle field, likely to be suboptimal.

The key concern of industrial automation which it addresses is optimal planning with clothoid curves based on a polygonal obstacle representation which can be constructed from range sensors such as LIDAR. The method is therefore most useful for applications where path quality is important and updates are required less frequently. For example, a centralized adaptive replanning module

which is tightly integrated with a central routing algorithm which collects information from a number of sensors and vehicles and makes safety critical path alteration decisions as discussed in [1]. Interaction with other vehicles may be based on the allocation of regions at different times. Validating the interactions becomes much easier if the information is collected from different sources (around blind corners for example) and a safe alternative path is generated which can be followed by the existing vehicle control system.

Freedom to choose a different objective function or weighting parameter b to generate different shape paths is a strength of this formulation but also reveals the weakness of using an optimisation approach. Different objective functions have different convergence characteristics for the same boundary conditions. One based on an equal weighting converges nicely but as b is increased, emphasising smoothness convergence takes longer as the constraints are only enforced at the start and end of each segment. The search of infeasible solutions leads to wasted iterations, especially when $b > 100$. The appropriate weighting for b might best be addressed using knowledge of the speed controllers to be used for path traversal in order to plan minimum time paths around obstructions and minimize the delay from obstructions in a site.

Real-time performance is important for the method to be useful as part of a AGV control system. The multiple shooting form was not much faster on the small environments tested. The analytical Jacobian of the constraints was effective at reducing the number of function evaluations to reach convergence but not total execution time. An improved implementation with more reuse of terms could approach the factor of two reduction in function evaluations recorded. Even with this improvement, the presented method would be slower than existing heuristic methods for clothoid parameter estimation. Its strength lies in the guarantee of optimality given the obstacle polygons (subject to convergence, which was strong in the limited numerical examples tested). Other avenues to improve performance in further work should explore improving the initial parameter guess based on the point to point solution between the corners of the regions, and the use of alternative numerical methods designed for convex constrained problems such as CVX [51]. Finally, even with execution time of several seconds it is possible to use automatic adaptive paths in some situations, as the clothoid shape allows reuse of the same optimal path by vehicles with differing dynamics by adjustment of the longitudinal speed to control the steering rate.

Expanding the environment with a disc the size of the largest dimension of the vehicle is a gross approximation, which will lead to over cautious paths. It would be better to represent the obstacles in $(2 + 1)$ dimensions, and plan in (x, y, ψ) . Obstacles can then be expanded based on the current heading of the path as described in [18] who use a

sum of squares approach to find $(2+1)$ D convex polygons. Applying these $(2+1)$ D constraints to the optimisation presented here would be straightforward but the number of path pieces needed to make the entire space reachable may be different. This could be investigated in a future study.

Testing a wider range of environments is left to further work. This may motivate an increase in the number of obstacle constraint points on the path to prevent clipping problems seen in Section 4.2. A constraint on the point of peak curvature may be most helpful before sampling the curve at approximately linear intervals. We were not able to prove the statement “provided N connected regions can be found between two poses, there always exists a smooth path between them comprised of line segments and clothoids in the order L-C-C-L”. A wider range of test cases would give confidence that this is true in many cases. Further testing would also be useful to identify cases where a path exists but the peak curvature is so high a vehicle would have to slow down unacceptably to traverse it.

APPENDIX A

1. ANALYTICAL GRADIENTS OF THE SINGLE SHOOTING FORMULATION WITHIN ONE REGION

Stacking the parameters for the optimisation described in Section 3 into a single vector $\mathbf{p}^{1 \times 6} = [\alpha_1, \alpha_2, L_1, L_2, s_0, s_F]^T$ allows the Jacobian of the objective for one region to be expressed as

$$\frac{\partial \mathbf{J}}{\partial \mathbf{p}} = 2 \cdot [b\alpha_1, b\alpha_2, L_1, L_2, s_0, s_F]^T \quad (\text{A.1})$$

The pose at the end of the path piece in the region of interest is $[X_F, Y_F, \Psi_F]$. For the derivatives we only consider one path piece, and drop the i subscript used in Equation 10. The equality constraint vector with goal pose $[\hat{X}, \hat{Y}, \hat{\Psi}]$ is given by

$$\mathbf{c}_{eq} = \begin{bmatrix} X_F - \hat{X} \\ Y_F - \hat{Y} \\ \Psi_F - \hat{\Psi} \\ \kappa_F - 0 \end{bmatrix} = \mathbf{0} \quad (\text{A.2})$$

and the inequality constraints to remain within a square region bounded by $[X_{max}, X_{min}, Y_{max}, Y_{min}]$ is given by

$$\mathbf{c}_{ineq} = \begin{bmatrix} X_F - X_{max} \\ -X_F + X_{min} \\ Y_F - Y_{max} \\ -Y_F + Y_{min} \end{bmatrix} \leq \mathbf{0} \quad (\text{A.3})$$

The matrix derivatives $\frac{\partial \mathbf{c}_{eq}}{\partial \mathbf{p}}^{(6 \times 4)}$ and $\frac{\partial \mathbf{c}_{ineq}}{\partial \mathbf{p}}^{(6 \times 4)}$ were constructed from scalar partial derivatives listed below. Each

element is defined in terms of the parameters \mathbf{p} in the following sections. The symbols are introduced in Section 2.1, the starting pose is taken to be the origin, so the end pose in global coordinates is given by Equation 10. Functions $C(\alpha, s, \kappa)$ and $S(\alpha, s, \kappa)$ are defined by Equation 3 and Equation 4 respectively. The peak curvature where the two clothoids meet is given the symbol $\kappa_m = \alpha_1 L_1$ and the final heading at the end of the second clothoid is given by $\Psi_F = \frac{\alpha_1 L_1^2}{2} + \frac{\alpha_2 L_2^2}{2}$.

A.1. Components with respect to α_1

$$\begin{aligned} \frac{\partial X_F}{\partial \alpha_1} &= \int_0^{L_1} -\frac{u^2}{2} \sin\left(\frac{\alpha_1 u^2}{2}\right) du \\ &+ \cos(\delta_1) \int_0^{L_2} (-L_1 \cdot u) \sin\left(\alpha_1 L_1 \cdot u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (-\sin(\delta_1) L_1^2/2) \int_0^{L_2} \cos(\alpha_1 L_1 + \alpha_2 u) du \\ &- \sin(\delta_1) \int_0^{L_2} (L_1 \cdot u) \cos\left(\alpha_1 L_1 \cdot u + \frac{\alpha_2 u^2}{2}\right) du \\ &- (\cos(\delta_1) L_1^2/2) \int_0^{L_2} \sin(\alpha_1 L_1 + \alpha_2 u) du \\ &\quad + s_F(L_1 L_2 + L_1^2/2)(-\sin\Psi_F) \quad (\text{A.4}) \end{aligned}$$

$$\begin{aligned} \frac{\partial Y_F}{\partial \alpha_1} &= \int_0^{L_1} \frac{u^2}{2} \cos\left(\frac{\alpha_1 u^2}{2}\right) du \\ &+ \sin(\delta_1) \int_0^{L_2} (-L_1 \cdot u) \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (\cos(\delta_1) \frac{L_1^2}{2}) \int_0^{L_2} \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ \cos(\delta_1) \int_0^{L_2} (L_1 \cdot u) \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (-\sin(\delta_1) \frac{L_1^2}{2}) \int_0^{L_2} \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &\quad + s_F(L_1 L_2 + L_1^2/2) \cos(\Psi_F) \quad (\text{A.5}) \end{aligned}$$

$$\frac{\partial \Psi_F}{\partial \alpha_1} = \frac{1}{2} L_1^2 + L_1 L_2 \quad (\text{A.6})$$

$$\frac{\partial \kappa_F}{\partial \alpha_1} = L_1 \quad (\text{A.7})$$

A.2. Components with respect to α_2

$$\begin{aligned} \frac{\partial X_F}{\partial \alpha_2} &= \cos(\delta_1) \int_0^{L_2} \left(\frac{-u^2}{2}\right) \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &- \sin(\delta_1) \int_0^{L_2} \left(\frac{u^2}{2}\right) \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \end{aligned}$$

$$+ s_F(-\sin\Psi_F) \left(\frac{L_2^2}{2}\right) \quad (\text{A.8})$$

$$\begin{aligned} \frac{\partial Y_F}{\partial \alpha_2} &= \\ &\sin(\delta_1) \int_0^{L_2} -\frac{u^2}{2} \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ \cos(\delta_1) \int_0^{L_2} \frac{u^2}{2} \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &\quad + s_F(\cos\Psi_F) \cdot \left(\frac{1}{2} L_2^2\right) \quad (\text{A.9}) \end{aligned}$$

$$\frac{\partial \Psi_F}{\partial \alpha_2} = \frac{1}{2} L_2^2 \quad (\text{A.10})$$

$$\frac{\partial \kappa_F}{\partial \alpha_2} = L_2 \quad (\text{A.11})$$

A.3. Components with respect to L_1

$$\begin{aligned} \frac{\partial X_F}{\partial L_1} &= \cos(\alpha_1 L_1^2/2) \\ &+ \cos(\delta_1) \int_0^{L_2} (-\alpha_1 \cdot u) \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (-\sin(\delta_1) \alpha_1 L_1) \int_0^{L_2} \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &- \sin(\delta_1) \int_0^{L_2} (\alpha_1 \cdot u) \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &- (\cos(\delta_1) \alpha_1 L_1) \int_0^{L_2} \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &\quad - s_F(\alpha_1 L_1 + \alpha_1 L_2) \sin(\Psi_F) \quad (\text{A.12}) \end{aligned}$$

$$\begin{aligned} \frac{\partial Y_F}{\partial L_1} &= \sin(\alpha_1 L_1^2/2) \\ &+ \sin(\delta_1) \int_0^{L_2} (-\alpha_1 \cdot u) \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (\cos(\delta_1) \alpha_1 L_1) \int_0^{L_2} \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ \cos(\delta_1) \int_0^{L_2} (\alpha_1 \cdot u) \cos\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &+ (\sin(\delta_1) \alpha_1 L_1) \int_0^{L_2} \sin\left(\alpha_1 L_1 u + \frac{\alpha_2 u^2}{2}\right) du \\ &\quad + s_F(\alpha_1 L_1 + \alpha_1 L_2) \cos(\Psi_F) \quad (\text{A.13}) \end{aligned}$$

$$\frac{\partial \Psi_F}{\partial L_1} = \alpha_1 L_1 + \alpha_1 L_2 \quad (\text{A.14})$$

$$\frac{\partial \kappa_F}{\partial L_1} = \alpha_1 \quad (\text{A.15})$$

A.4. Components with respect to L_2

$$\begin{aligned} \frac{\partial X_F}{\partial L_2} &= \cos\left(\alpha_1 L_1 L_2 + \frac{\alpha_2 L_2^2}{2}\right) \cos(\delta_1) \\ &- \sin\left(\alpha_1 L_1 L_2 + \frac{\alpha_2 L_2^2}{2}\right) \sin(\delta_1) \\ &+ s_F \cdot (-\sin \Psi_F)(\alpha_1 L_1 + \alpha_2 L_2) \end{aligned} \quad (\text{A.16})$$

$$\begin{aligned} \frac{\partial Y_F}{\partial L_2} &= \cos\left(\alpha_1 L_1 L_2 + \frac{\alpha_2 L_2^2}{2}\right) \sin(\delta_1) \\ &+ \sin\left(\alpha_1 L_1 L_2 + \frac{\alpha_2 L_2^2}{2}\right) \cos(\delta_1) \\ &+ s_F \cdot (\cos \Psi_F)(\alpha_1 L_1 + \alpha_2 L_2) \end{aligned} \quad (\text{A.17})$$

$$\frac{\partial \Psi_F}{\partial L_2} = \alpha_1 L_1 + \alpha_2 L_2 \quad (\text{A.18})$$

$$\frac{\kappa_F}{\partial L_2} = \alpha_2 \quad (\text{A.19})$$

A.5. Components with respect to s_0

$$\frac{\partial X_F}{\partial s_0} = 1 \quad (\text{A.20})$$

$$\frac{\partial Y_F}{\partial s_0} = 0 \quad (\text{A.21})$$

$$\frac{\partial \Psi_F}{\partial s_0} = 0 \quad (\text{A.22})$$

$$\frac{\partial \kappa_F}{\partial s_0} = 0 \quad (\text{A.23})$$

A.6. Components with respect to s_F

$$\frac{\partial X_F}{\partial s_F} = \cos(\Psi_F) \quad (\text{A.24})$$

$$\frac{\partial Y_F}{\partial s_F} = \sin(\Psi_F) \quad (\text{A.25})$$

$$\frac{\partial \Psi_F}{\partial s_F} = 0 \quad (\text{A.26})$$

$$\frac{\partial \kappa_F}{\partial s_F} = 0 \quad (\text{A.27})$$

2. POSE OPERATORS

These operators are defined to simplify working with rigid bodies in two dimensions. A rigid body in a plane has $2 + 1$ parameters, two for translation in the plane and one for heading angle. The position in metres and the heading in radians can be assembled into a three vector, called the pose.

$$\mathbf{p}_1 = [x_1, y_1, \psi_1]^T \quad (\text{A.28})$$

$$\mathbf{p}_2 = [x_2, y_2, \psi_2]^T \quad (\text{A.29})$$

Working with higher dimensional spaces it is common to use homogeneous coordinates to simplify operations on rigid bodies. The two operators defined here are convenient for working in 2D without the complexity of homogeneous notation. They are inverse operators in the sense that $\mathbf{p}_3 = \mathbf{p}_1 \oplus \mathbf{p}_2 \rightarrow \mathbf{p}_3 \ominus \mathbf{p}_2 = \mathbf{p}_1$. The decompose operator \ominus is order dependent like ordinary subtraction. They are defined below using the algebra of homogeneous coordinates given in [14] Part A Robotics Foundations, Section 2.2.3 Homogeneous Transforms p16.

A.1. Compose \oplus

This operator takes the second pose and rotates it into the frame of the first before vector addition of all three components.

$$\mathbf{p}_3 = \mathbf{p}_1 \oplus \mathbf{p}_2 = \begin{bmatrix} x_1 + x_2 \cos(\psi_2) - y_2 \sin(\psi_2) \\ y_1 + x_2 \sin(\psi_2) + y_2 \cos(\psi_2) \\ \psi_1 + \psi_2 \end{bmatrix} \quad (\text{A.30})$$

. If each pose is expressed in homogeneous 3x3 form,

$$\mathbf{T}_1 = \begin{bmatrix} \cos(\psi_1) & -\sin(\psi_1) & x_1 \\ \sin(\psi_1) & \cos(\psi_1) & y_1 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A.31})$$

the same operation is a matrix multiplication

$$\mathbf{T}_3 = \mathbf{T}_1 \mathbf{T}_2 \quad (\text{A.32})$$

A.2. Decompose \ominus

This operator finds the pose of \mathbf{p}_3 expressed relative to pose \mathbf{p}_1

$$\mathbf{p}_1 = \mathbf{p}_3 \ominus \mathbf{p}_2 \quad (\text{A.33})$$

If each pose is expressed in homogeneous 3x3 form, the same operation is a matrix multiplication by the the inverse

$$\mathbf{T}_1 = \mathbf{T}_3 \mathbf{T}_2^{-1} \quad (\text{A.34})$$

REFERENCES

- [1] V. Digani, F. Caramaschi, L. Sabattini, C. Secchi, and C. Fantuzzi, "Obstacle avoidance for industrial AGVs," *Proceedings - 2014 IEEE 10th International Conference on Intelligent Computer Communication and Processing, ICCP 2014*, pp. 227–232, 2014.
- [2] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles," *arXiv preprint arXiv:*, pp. 1–27, 2016.
- [3] I. F. Vis, "Survey of research in the design and control of automated guided vehicle systems," *European Journal of Operational Research*, vol. 170, no. 3, pp. 677–709, 2006.
- [4] N. Boysen, R. de Koster, and F. Weidinger, "Warehousing in the e-commerce era: A survey," *European Journal of Operational Research*, vol. 277, no. 2, pp. 396–411, 2019.
- [5] S. M. LaValle and D. Leidner, "Chapter 6: Combinatorial Motion Planning," in *Planning Algorithms*, ch. 6, pp. 249–310, Cambridge University Press, 2006.
- [6] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015.
- [7] J. Moreau, P. Melchior, S. Victor, L. Cassany, M. Moze, F. Aioun, and F. Guillemard, "Reactive path planning in intersection for autonomous vehicle," *IFAC-PapersOnLine*, vol. 52, no. 5, pp. 109–114, 2019.
- [8] A. Chebly, R. Talj, A. Charara, A. Chebly, R. Talj, A. Charara, M. Planning, C. Alia, T. Reine, and C. Ali, "Maneuver Planning for Autonomous Vehicles , with Clothoid Tentacles for Local Trajectory Planning," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 2017.
- [9] E. Bertolazzi and M. Frego, "Interpolating clothoid splines with curvature continuity," *Mathematical Methods in the Applied Sciences*, vol. 41, no. 4, pp. 1723–1737, 2018.
- [10] E. Lambert, R. Romano, and D. Watling, "Optimal Path Planning with Clothoid Curves for Passenger Comfort," in *Proceedings of the 5th International Conference on Vehicle Technology and Intelligent Transport Systems* (O. Gusikhin and M. Helfert, eds.), (Heraklion, Crete), pp. 609–615, SCITEPRESS, 2019.
- [11] M. Frego, E. Bertolazzi, F. Biral, D. Fontanelli, and L. Palopoli, "Semi-analytical minimum time solutions for a vehicle following clothoid-based trajectory subject to velocity constraints," *2016 European Control Conference, ECC 2016*, pp. 2221–2227, 2017.
- [12] V. Girbés, L. Armesto, and J. Tornero, "Path following hybrid control for vehicle stability applied to industrial forklifts," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 910–922, 2014.
- [13] M. Pivtoraiko, R. A. Knepper, and A. Kelly, "Differentially constrained mobile robot motion planning in state lattices," *Journal of Field Robotics*, vol. 26, no. 3, pp. 308–333, 2009.
- [14] B. Siciliano and O. Khatib, eds., *Springer Handbook of Robotics*. Springer Handbooks, Berlin: Springer, 2 ed., 2016.
- [15] Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli, and J. P. How, "Real-Time Motion Planning With Applications to Autonomous Urban Driving," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1105–1118, 2009.
- [16] S. M. LaValle and J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," *4th Workshop on Algorithmic and Computational Robotics: New Directions*, pp. 293–308, 2000.
- [17] S. Yoon, D. Lee, J. Jung, and D. H. Shim, "Spline-based RRT* Using Piecewise Continuous Collision-checking Algorithm for Car-like Vehicles," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 90, no. 3-4, pp. 537–549, 2018.
- [18] R. Deits and R. Tedrake, "Efficient mixed-integer planning for UAVs in cluttered environments," *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 42–49, 2015.
- [19] C. Rosmann, F. Hoffmann, and T. Bertram, "Kinodynamic trajectory optimization and control for car-like robots," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2017-Septe, pp. 5681–5686, 2017.
- [20] J. C. Kim, D. S. Pae, and M. T. Lim, "Obstacle Avoidance Path Planning based on Output Constrained Model Predictive Control," *International Journal of Control, Automation and Systems*, vol. 17, no. 11, pp. 2850–2861, 2019.
- [21] M. Yue, X. Wu, L. Guo, and J. Gao, "Quintic Polynomial-based Obstacle Avoidance Trajectory Planning and Tracking Control Framework for Tractor-trailer System," *International Journal of Control, Automation and Systems*, vol. 17, no. 10, pp. 2634–2646, 2019.
- [22] R. Levien, "The Elastica: A Mathematical History," tech. rep., University of California, Berkeley, 2008.

- [23] S. Gim, L. Adouane, S. Lee, and J. P. Dérutin, "Clothoids Composition Method for Smooth Path Generation of Car-Like Vehicle Navigation," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 88, no. 1, pp. 129–146, 2017.
- [24] J. Henrie and D. Wilde, "Planning Continuous Curvature Paths Using Constructive Polylines," *Journal of Aerospace Computing, Information, and Communication*, vol. 4, no. 12, pp. 1143–1157, 2007.
- [25] M. Lundberg, *Path planning for autonomous vehicles using clothoid based smoothing of A* generated paths and optimal control*. PhD thesis, KTH Royal Institute of Technology, 2017.
- [26] T. Fraichard and A. Scheuer, "From Reeds and Shepp's to Continuous-Curvature Paths," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 1025–1035, 2004.
- [27] F. Lamiroux and J. P. Laumond, "Smooth motion planning for car-like vehicles," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 498–502, 2001.
- [28] M. Fliess, J. Levine, P. Martin, and P. Rouchon, "Flatness and defect of non-linear systems: Introductory theory and examples," *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.
- [29] J. A. Reeds and L. A. Shepp, "Optimal paths for a car that goes forwards and backwards," *Pacific Journal of Mathematics*, vol. 145, no. 2, pp. 367–393, 1990.
- [30] S. Gim, *Flexible and Smooth Trajectory Generation based on Parametric Clothoids for Nonholonomic Car-like Vehicles*. PhD thesis, Université Clermont Auvergne, 2017.
- [31] R. Gobithaasan, Y. Wei, K. Miura, and M. Shanmugavel, "Optimal Path Smoothing with Log-Aesthetic Curves based on Shortest Distance, Minimum Bending Energy and Curvature Variation Energy," in *Proceedings of CAD'19*, (Singapore), pp. 397–402, 2019.
- [32] J. E. Solanes, L. Armesto, J. Tornero, P. Muñoz-Benavent, and V. Girbés, "Mobile robot obstacle avoidance based on quasi-holonomic smooth paths," in *Joint of the 13th Annual Conference on Towards Autonomous Robotic Systems, TAROS 2012 and the 15th Annual FIRA RoboWorld Congress*, vol. 7429, pp. 152–163, 2012.
- [33] J. A. Silva and V. Grassi, "Clothoid-based global path planning for autonomous vehicles in urban scenarios," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 4312–4318, 2018.
- [34] P. F. Lima, M. Trincavelli, J. Martensson, and B. Wahlberg, "Clothoid-based model predictive control for autonomous driving," *2015 European Control Conference, ECC 2015*, pp. 2983–2990, 2015.
- [35] M. G. Plessen, P. F. Lima, J. Martensson, A. Bemporad, and B. Wahlberg, "Trajectory planning under vehicle dimension constraints using sequential linear programming," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, vol. 2018-March, pp. 1–6, 2018.
- [36] D. Shin, S. Singh, and W. Whittaker, "Path Generation for a Robot Vehicle Using Composite Clothoid Segments," *IFAC Proceedings Volumes*, vol. 25, no. 6, pp. 443–448, 1992.
- [37] D. J. Walton and D. S. Meek, "A controlled clothoid spline," *Computers and Graphics (Pergamon)*, vol. 29, no. 3, pp. 353–363, 2005.
- [38] S. Thrun and A. Buecken, "Integrating grid-based and topological maps for mobile robot navigation," *Proceedings of the National Conference on Artificial Intelligence*, vol. 2, no. August, pp. 944–950, 1996.
- [39] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. Cambridge, Mass.: MIT Press, 2005.
- [40] S. M. LaValle and D. Leidner, "Chapter 6: Combinatorial Motion Planning," in *Planning Algorithms*, ch. 6, pp. 249–310, Cambridge University Press, 2006.
- [41] R. Deits and R. Tedrake, "Computing large convex regions of obstacle-free space through semidefinite programming," *Springer Tracts in Advanced Robotics*, vol. 107, pp. 109–124, 2015.
- [42] Tianyu Gu, *Improved Trajectory Planning for On-Road Self-Driving Vehicles Via Combined Graph Search, Optimization & Topology Analysis*. Doctor of philosophy, Carnegie Mellon University, 2017.
- [43] Hyster, "E30-40HSD Technical guide." www.hyster.com, August 2020.
- [44] R. Baird, "An Autonomous Forklift Research Platform for Warehouse Operations," 2018.
- [45] G. Raballand, "How do differing standards increase trade costs? THE CASE OF PALLETS Gaël," *World Bank Policy Research Working Paper 3519*, pp. 1–20, 2005.
- [46] R. Bostelman, R. Norcross, J. Falco, and J. Marvel, "Development of standard test methods for unmanned and manned industrial vehicles used near humans," *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2013*, vol. 8756, p. 87560P, 2013.
- [47] M. P. Kelly, "Transcription Methods for Trajectory Optimization: a beginners tutorial," tech. rep., 2017.
- [48] "MATLAB Help Center - Choosing the Algorithm." <https://uk.mathworks.com/help/optim/ug/choosing-the-algorithm.html>, August 2020.
- [49] L. F. Shampine, "Vectorized adaptive quadrature in MATLAB," *Journal of Computational and Applied Mathematics*, vol. 211, no. 2, pp. 131–140, 2008.

- [50] M. A. Cooper, J. F. Raquet, and R. Patton, "Range information characterization of the Hokuyo UST-20LX LIDAR sensor," *Photonics*, vol. 5, no. 2, 2018.
- [51] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." <http://cvxr.com/cvx>, Mar. 2014.



Edward Derek Lambert received his MEng in Engineering Science from the University of Oxford, graduating with a first class degree in 2013. After some time in industry, he began an EPSRC Doctoral Training Partnership Studentship in 2018, working towards a PhD degree at the Institute for Transport Studies at the University of Leeds. His research interests include

optimisation-based path planning and multiple vehicle motion coordination.



Richard Romano has over twenty five years of experience developing and testing AVs and ADAS concepts and systems which began with the Automated Highway Systems (AHS) project while he directed the Iowa Driving Simulator in the early 1990's. He received his BSc and MSc in Engineering Science and Aerospace Engineering respectively from the University

of Toronto, Canada and a PhD in Motion Drive Algorithms for Large Excursion Motion Bases, Industrial Engineering from the University of Iowa, USA. In addition to a distinguished career in industry he has supervised numerous research projects and authored many journal papers. In 2015 he was appointed Professor of Driving Simulation at the Institute for Transport Studies, University of Leeds, UK. His research interests include the development, validation and application of transport simulation to support the human-centred design of vehicles and infrastructure.



David Watling's primary research focus is the development of mathematical models and methods for analysing transport systems, especially those that represent the interactions between travellers' decision-making and the physical infrastructure. He has particularly developed methods for modelling, simulating or optimizing transport networks with random, dynamic or

unreliable elements. With a B.Sc. degree in mathematics from the University of Leeds and a Ph.D. from the Department of Probability and Statistics at the University of Sheffield, he has held the post of Centenary Chair of Transport Analysis at the University of Leeds since its instigation in 2004, where he is co-leader of the Spatial Modelling and Dynamics group in the Institute for Transport Studies.