



Enhanced path smoothing based on conjugate gradient descent for firefighting robots in petrochemical complexes

Naoki Mizuno, Kazunori Ohno, Ryunosuke Hamada, Hiroyoshi Kojima, Jun Fujita, Hisanori Amano, Thomas Westfechtel, Takahiro Suzuki & Satoshi Tadokoro

To cite this article: Naoki Mizuno, Kazunori Ohno, Ryunosuke Hamada, Hiroyoshi Kojima, Jun Fujita, Hisanori Amano, Thomas Westfechtel, Takahiro Suzuki & Satoshi Tadokoro (2019) Enhanced path smoothing based on conjugate gradient descent for firefighting robots in petrochemical complexes, *Advanced Robotics*, 33:14, 687-698, DOI: [10.1080/01691864.2019.1632221](https://doi.org/10.1080/01691864.2019.1632221)

To link to this article: <https://doi.org/10.1080/01691864.2019.1632221>



Published online: 26 Jun 2019.



Submit your article to this journal [↗](#)



Article views: 37




View Crossmark data [↗](#)

FULL PAPER



Enhanced path smoothing based on conjugate gradient descent for firefighting robots in petrochemical complexes*

Naoki Mizuno ^a, Kazunori Ohno^b, Ryunosuke Hamada^c, Hiroyoshi Kojima^d, Jun Fujita^d, Hisanori Amano^e, Thomas Westfechtel^a, Takahiro Suzuki^c and Satoshi Tadokoro^a

^aGraduate School of Information Science, Tohoku University, Sendai, Japan; ^bNICHe, Tohoku University/RIKEN AIP, Japan; ^cNICHe, Tohoku University, Sendai, Japan; ^dMitsubishi Heavy Industries Ltd., Kobe, Japan; ^eNational Research Institute of Fire and Disaster, Tokyo, Japan

ABSTRACT

The firefighting robot system (FFRS) comprises several autonomous robots that can be deployed to fire disasters in petrochemical complexes. For autonomous navigation, the path planner should consider the robot constraints and characteristics. Specifically, three requirements should be satisfied for a path to be suitable for the FFRS. First, the path must satisfy the maximum curvature constraint. Second, it must be smooth for robots to easily execute the trajectory. Third, it must allow reaching the target location in a specific heading. We propose a path planner that provides smooth paths, satisfy the maximum curvature constraint, and allows a suitable robot heading. The path smoother is based on the conjugate gradient descent, and three approaches are proposed for this path planner to meet all the FFRS requirements. The effectiveness of these approaches is qualitatively and quantitatively evaluated by examining the generated paths. Finally, the path planner is applied to an actual robot to verify the suitability of the generated paths for the FFRS, and planning is applied to another type of robot to demonstrate the wide applicability of the proposed planner.

ARTICLE HISTORY

Received 14 February 2019
Accepted 4 May 2019

KEYWORDS

Path smoothing; motion planning; mobile robot; autonomous ground vehicle; firefighting robot

1. Introduction

In recent years, there have been several fire disasters in petrochemical complexes, with some of them including human casualties. Moreover, firefighters face numerous risks at the disaster site related to heat radiation, explosions, and collapsing structures. Thus, mitigating the risks of firefighting in petrochemical complexes is essential for preserving the safety of people and property.

To reduce the risks that human firefighters face, we have been developing a firefighting robot system (FFRS). The system comprises a team of autonomous mobile robots, as shown in Figure 1, deployed to fires occurring in petrochemical complexes and reduce risks for humans.

In the FFRS, water-shooting and hose-extension robots are operated as follows. The headquarters determine the location for the robot to shoot water. Both types of robots jointly and autonomously navigate in the petrochemical complex to the specified location. After reaching their target location, the hose-extension robot moves back to roll out the hose, which is 300 m and 150 mm in length and thickness, respectively. The hose-extension robot must keep a turning radius above 5 m to prevent hose kinking or losing control during water shooting at

4000 L/m and 1 MPa. After the hose-extension the robot returns to the target location, the firefighters connect the hose to a water source, and water is shot to the fire source.

We are particularly interested in path planning of water-shooting and hose-extension robots. Their operation, however, imposes several constraints on path planning. For instance, the hose imposes a minimum turning radius of 5 m and impedes reverse locomotion of the robot to prevent kinking when water pressure is applied. Other requirements that the path planner for these two types of robots should satisfy for effective deployment and operation include

- (1) a maximum curvature of 0.2 (derived from the minimum turning radius) for the path to prevent hose kinking,
- (2) a smooth path to simplify trajectory execution, and
- (3) correct robot heading at the target location to facilitate water-shooting, as the water cannon has a limited yaw for aiming.

We proposed a path planning method for both the water-shooting and hose-extension robots satisfying

CONTACT Naoki Mizuno  mizuno.naoki@rm.is.tohoku.ac.jp

*Portions of this work were previously presented at the Robotics Symposia 2018, Toyama, Japan.

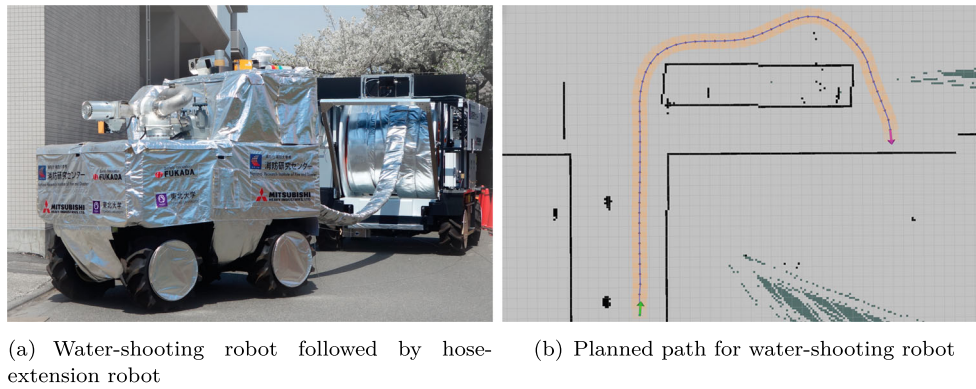


Figure 1. Firefighting robots and a path planned for their deployment. The robots should autonomously navigate together in a petro-chemical complex from the initial location to the specified target location. The line shows the planned path from starting location (bottom arrow pointing up) to the target location (arrow on right pointing down) following a specific trajectory. (a) Water-shooting robot followed by hose-extension robot. (b) Planned path for water-shooting robot.

all these requirements. The planner is based on the hybrid-state A* algorithm that was proposed by Dolgov et al. [1] and allows to compute paths considering the kinematics of steering-type vehicles in real time.

The proposed path planning method for water-shooting and hose-extension robots has the following characteristics:

- (1) Each point in the path satisfies the maximum curvature.
- (2) The resulting path is smooth, i.e. the curvature variation is small.
- (3) The path reaches the target location with the specified heading.

The remainder of this paper is organized as follows: Section 2 summarizes related work and sets the context for this study. Section 3 describes the hybrid-state A* algorithm and the problems for adapting it to the FFRS. Section 4 details the solutions to these problems, and Section 5 verifies the resulting method through simulation and implementation on an actual water-shooting robot. Finally, Sections 6 and 7 present the discussion and conclusions, respectively.

2. Related work

Various approaches commonly used to plan paths for mobile robots include algorithms based on rapidly exploring random trees (RRTs), Voronoi diagrams, and A* search.

RRT is a stochastic search algorithm in which the search tree is randomly expanded, rapidly covering unexplored regions and eventually reaching the goal [2]. Hence, RRT has been used for trajectory planning of manipulators and motion planning for mobile robots [3–5]. Kuffner and LaValle [3] proposed

RRT-connect, which builds two RRTs from both the initial and goal points for faster computation. Likewise, Bruce and Veloso [4] extended the RRT for the goal and cached waypoints to improve the computation time. Urmson and Simmons [5] implemented a heuristic on the RRT to guide the tree expansion toward the goal. Their proposed heuristic considers the cost-so-far and cost-to-go for increasing the solution quality (i.e. shortening the path length) compared with the vanilla implementation of RRT.

Generalized Voronoi diagrams can be used for obstacle-avoidance path planning. Takahashi and Schilling [6] used these diagrams along with several heuristic functions to plan collision-free paths. Considering the trade-off between safeness and the length of a path, generalized Voronoi diagrams have been useful for applications where safety is critical.

Grid-based planning is often used in 2D path planning for mobile robots. Many grid-based algorithms derive from the well-known A* algorithm [7], including the variants D* [8], D* lite [9], and field D* [10]. D* stands for dynamic A* and extends the A* search algorithm, achieving efficient online path searching in dynamic, unknown environments by reusing previous search results. D* lite search is based on lifelong planning A* [11] and resembles D* but with a different approach using a priority queue to efficiently select the nodes for cost updating.

Dolgov et al. [1] proposed the hybrid-state A* algorithm that generates paths for robots with non-holonomic constraints in real time. The hybrid-state A* algorithm consists of two phases: grid-based path searching and smoothing. Path search uses the Reeds–Shepp model [12] that allows considering the vehicle heading during search expansion. The CG smoother is a gradient descent-based smoothing algorithm that generates paths by minimizing an objective function. The objective function is a conjugate of three weighted terms, namely, an

obstacle term that leads the path away from obstacles, a curvature term that prevents the path from violating the maximum curvature, and a smoothness term that aims to maximize the straightness of the path. By controlling these weights, the smoothing outcome can be controlled to prioritize certain terms as needed.

Path smoothing is often applied with path planning. In fact, path smoothers optimize the path to facilitate the navigation of mobile robots.

Bézier curves are commonly used for path smoothing, where multiple control points parametrically describe each curve. Yang and Sukkarieh [13] proposed path smoothing based on Bézier curves satisfying a maximum curvature. To this end, they reduce the three variables of the curve into one design parameter that can be expressed according to the maximum curvature. The paths have a geometrically continuous curvature variation (i.e. G^2 continuity).

Alternatively, B-splines can be used for path smoothing [14]. B-spline curves are similar to Bézier curves given the parametric functions with control points and knots that determine the curve shape. Their application in robotics ranges from planning trajectories for unmanned aerial vehicles [15] and manipulators [16] to smoothing paths for car-like robots [17]. The path smoothing algorithm proposed by Elbanhawi et al. [17] satisfies the maximum curvature constraint using B-spline curves and provides C^2 continuity by inserting midpoints to a segment when the constraint is not satisfied.

For our proposed path planning method, we selected grid-based planning using the hybrid-state A^* algorithm, because the FFRS relies on occupancy grid maps, and the Reeds–Shepp model suitably describes the kinematics of both the water-shooting and hose-extension robots. In addition, we use the CG smoother that is suitable for four-wheeled steering-type robots with non-holonomic constraints, which are similar to autonomous ground vehicles such as self-driving cars. However, the FFRS imposes additional constraints for firefighting, and therefore requires a path planner that generates smooth, constraint-satisfying paths suitable for the intended task.

3. Path planning for FFRS

3.1. FFRS constraints

The FFRS operation imposes several constraints on the robots. In this section, we describe the constraints and requirements of the path planner for the water-shooting and hose-extension robots.

The weight of the 150 mm-thick, 300 m-long hose hinder its carrying by a single robot, given the friction it

generates. Hence, we devised a robot for water shooting to perform the firefighting actions and another robot to lay out the hose and provide water for shooting. These two robots are loosely connected through the hose. Once the destination is determined by the headquarters, a path from the current robot location to the target fire source is computed. The global path, with a length up to few hundred meters, must be computed within 10 s because both the search area and path length are large from a planning viewpoint. The two robots start navigating toward the target location, and once they reach it, the hose-extension robot comes back to roll out the hose.

Although this coordinated operation allows deploying the hose without dragging, it imposes some kinematic constraints on the robots.

For instance, the hose must maintain a minimum turning radius of 5 m when laid out to withstand water pressure and prevent the hose from kinking. The minimum turning radius of 5 m imposes the same turning radius to the hose-extension robot during hose placement; thus, imposing this constraint to the path planner.

Neither the water-shooting nor the hose-extension robot is allowed to move in reverse, because the former guides the hose-extension robot, which should keep the reflector mounted on the back of the water-shooting robot in sight for navigation. In fact, because the water-shooting robot moving in reverse may cause jackknifing, it can lead the other robot to lose sight of the reflect and compromise joint navigation. On the other hand, the hose-extension robot cannot move in reverse because the hose is rolled up tightly onto it, and once rolled out, it requires a human operator to roll back and tighten it. Therefore, the path planner is forbidden to retrieve paths including reverse motions.

Overall, the path planner should present the following capabilities to be usable on the water-shooting and hose-extension robots:

- (1) Path must not include reverse motions.
- (2) Path must avoid collision with obstacles along the trajectory.
- (3) Path with length of few hundred meters should be computed within 10 s.
- (4) Path must not contain kinks.
- (5) Path must satisfy the given minimum turning radius.
- (6) Path should achieve the specified heading for the water-shooting robot.

3.2. Hybrid-State A^* algorithm

The hybrid-state A^* planning algorithm consists of two phases: grid-based planning and gradient descent-based smoothing. Planning, which comprises initial and target locations and a map, expands nodes (i.e. the robot next

state) until reaching the target. The two main differences between the conventional A* search and hybrid-state A* are the heading information contained in the node and the heuristic function to estimate the cost to the target. Specifically, hybrid-state A* uses the Reeds–Shepp model that describes the robot orientation besides its Cartesian x - y position. Hence, the planner can consider the robot minimum turning radius, i.e. maximum curvature. For the FFRS, we ensure that the generated path does not violate the maximum curvature constraint and does not contain any reverse motions by limiting the curvature of the Reeds–Shepp curve and preventing the nodes from expanding in reverse. The heuristic function compares two estimated costs, which constitute the hybrid structure of the algorithm: cost to the goal considering the obstacles but not the nonholonomic constraint of the robot and cost to the goal neglecting the obstacles but considering the nonholonomic constraints. This hybrid approach improves the accuracy of the estimated cost to the goal and consequently the generated-path quality.

The path generated by hybrid-state A* is processed using the smoother based on conjugate gradient or CG smoother. Gradient descent is an iterative optimization algorithm in which an objective function describes the desired state of the solution. The coordinate of each point in the path is updated at each iteration according to the derivative of the objective function, such that it minimizes the function. The objective function is a weighted sum of three terms, namely, the obstacle, curvature, and smoothness terms, given by the first to third terms of Equation (1), respectively. In this equation, \mathbf{x}_i is the Cartesian coordinate of a point, $\Delta\mathbf{x}_i$ is the vector from \mathbf{x}_i to \mathbf{x}_{i+1} , \mathbf{o}_i is the distance to the closest obstacle, d_{max} is the maximum allowable distance to the obstacle, κ_i is the curvature at the i th point, and κ_{max} is the maximum allowable curvature (i.e. inverse of the minimum turning radius). The obstacle term guides the path away from obstacles to avoid collision, whereas the curvature term guides the smoothing to satisfy the maximum curvature, and the smoothness term maximizes the straightness of the segments and maintains a constant interval between points along the path. Three weights w_o , w_k , and w_s control the contribution of each of the three terms.

$$w_o \sum_{i=1}^N (|\mathbf{x}_i - \mathbf{o}_i| - d_{max})^2 + w_k \sum_{i=1}^N (\kappa_i - \kappa_{max})^2 + w_s \sum_{i=1}^N (\Delta\mathbf{x}_{i+1} - \Delta\mathbf{x}_i)^2. \quad (1)$$

We use an open-source implementation of the hybrid-state A* algorithm [18] to generate paths that contain

no reverse motion and no collision (Requirements 1 and 2). The algorithm can perform path planning within the given time limit (Requirement 3). To generate suitable paths, three problems should be addressed for generating suitable paths for the FFRS, as detailed in the sequel.

3.3. Hybrid-State A* for FFRS

In this section, we describe the aspects of the hybrid-state A* algorithm that do not conform with the requirements of the water-shooting and hose-extension robots from the FFRS.

3.3.1. Path with kinks by high weight of curvature term

The smoothed path can contain kinks when setting a high weight to the curvature term of Equation (1), as shown in Figure 2. This occurs when only the curvature term is enabled (i.e. curvature weight w_k set to 1 and other weights to 0) and represents a problem for the FFRS because there are cases where we enable only the curvature term to satisfy the constraint of the 5 m minimum turning radius.

The expected output from the smoother is a path whose curvature remains below 0.2. The second term in Equation (1) minimizes the difference between κ_i and κ_{max} , which if considered as an objective function, retrieves a path with constant curvature κ_{max} .

3.3.2. Path shrinkage by increasing number of iterations

Shrinkage is illustrated in Figure 3 and occurs when the smoother is executed over an excessive number of iterations on either a looping or semi-looping path. In

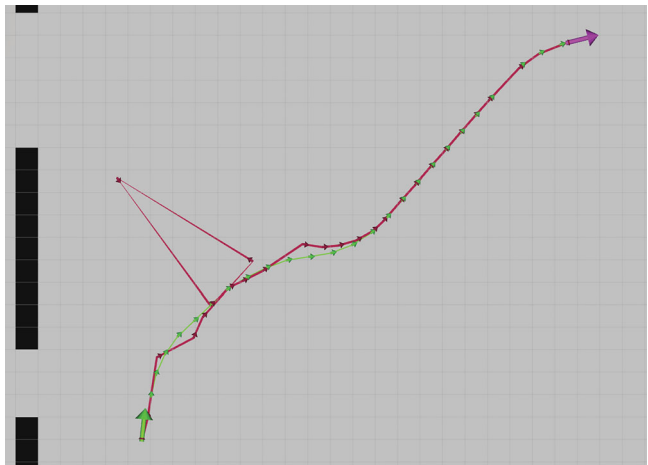


Figure 2. Path contains kinks when enabling only the curvature term. The generated path presents kinks after smoothing the path given to the path smoother using only the curvature term.

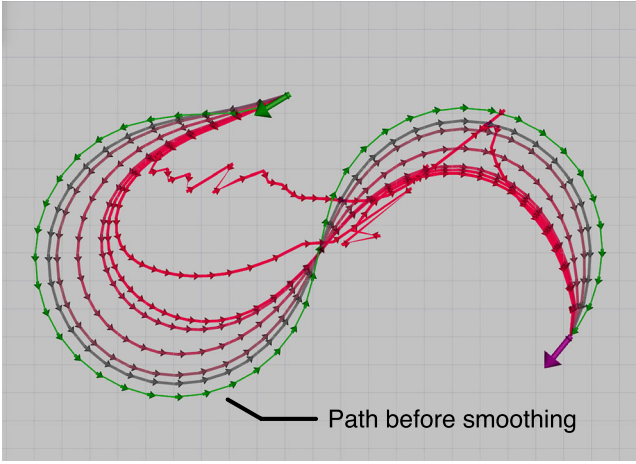


Figure 3. Path shrinkage caused by a high number of smoothing iterations on the looping path. Shrinkage is more severe as the number of smoothing iterations increases. Outcomes are shown for 15, 25, 50, 75, 80, 85, and 100. In this example, shrinkage occurs from approximately 85 iterations.

Figure 3, the path gradually shrinks toward the center of the loop (the two centers of the S-shape) until evolving into an untraversable form.

However, a fixed number of iterations do not solve this problem, because the optimal number of smoothing iterations depends on the length and shape of the path and the deployment environment. For example, the optimal number of iterations for a large-scale petrochemical complex differs from that for narrow roads or parking lots. Furthermore, the environment in which the FFRS will be deployed can change depending on the task, as it can be used in either petrochemical complexes with wide roads or in congested plants. Therefore, instead of fixing the number of smoothing iterations, we should monitor the convergence of the solution and terminate the iterative smoothing process when a suitable path is obtained.

3.3.3. Unsatisfied target location and heading

The output from the smoother can also present kinks near the initial and target locations, resulting in unsatisfied target heading, as illustrated in Figure 4. Although the path before smoothing satisfies the target heading, the smoothing result does not. In fact, the robot heading when executing the path is not the specified one, because the smoother does not consider headings.

It is essential for proper FFRS operation that the resulting path from the smoother satisfies the specified target heading, given that the water-shooting cannon has a limited yaw angle of $\pm 60^\circ$. Thus, the robot must be facing toward the tank in order to shoot water into the required direction for firefighting.

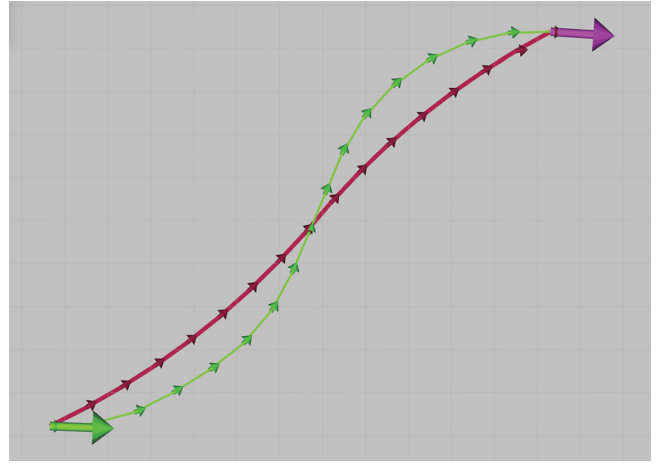


Figure 4. Path after smoothing does not satisfy the specified target heading. The path before smoothing goes straight into the goal, whereas that after smoothing does not achieve the specified target heading due to the approach angle.

4. Proposed path planner for FFRS

In this section, we describe three approaches to solve the above-mentioned problems of hybrid-state A* applied to the FFRS and obtain a path planner that generates suitable paths for the water-shooting and hose-extension robots.

4.1. Partial derivative of curvature term

We examine the partial derivative of the curvature term to prevent the maximum curvature problem when enabling only the curvature term. Curvature κ_i is expressed as

$$\kappa_i = \frac{\Delta\phi_i}{\Delta\mathbf{x}_i}, \quad (2)$$

where $\Delta\phi_i$ is the angle between $\Delta\mathbf{x}_i$ and $\Delta\mathbf{x}_{i+1}$ expressed as

$$\Delta\phi_i = \cos^{-1} \frac{\Delta\mathbf{x}_i^T \Delta\mathbf{x}_{i+1}}{|\Delta\mathbf{x}_i| |\Delta\mathbf{x}_{i+1}|}. \quad (3)$$

The partial derivative of κ_i with respect to \mathbf{x}_i is calculated as follows (the difference from the partial derivatives by Dolgov et al. [1] are highlighted in bold):

$$\frac{\partial \kappa_i}{\partial \mathbf{x}_i} = + \frac{1}{|\Delta\mathbf{x}_i|} \frac{\partial \Delta\phi_i}{\partial \cos(\Delta\phi_i)} \frac{\partial \cos(\Delta\phi_i)}{\partial \mathbf{x}_i} - \frac{\Delta\phi_i}{(\Delta\mathbf{x}_i)^2} \frac{\partial \Delta\mathbf{x}_i}{\partial \mathbf{x}_i} \quad (4)$$

$$\frac{\partial \kappa_i}{\partial \mathbf{x}_{i-1}} = + \frac{1}{|\Delta\mathbf{x}_i|} \frac{\partial \Delta\phi_i}{\partial \cos(\Delta\phi_i)} \frac{\partial \cos(\Delta\phi_i)}{\partial \mathbf{x}_{i-1}} + \frac{\Delta\phi_i}{(\Delta\mathbf{x}_i)^2} \frac{\partial \Delta\mathbf{x}_i}{\partial \mathbf{x}_{i-1}} \quad (5)$$

$$\frac{\partial \kappa_i}{\partial \mathbf{x}_{i+1}} = + \frac{1}{|\Delta\mathbf{x}_i|} \frac{\partial \Delta\phi_i}{\partial \cos(\Delta\phi_i)} \frac{\partial \cos(\Delta\phi_i)}{\partial \mathbf{x}_{i+1}}, \quad (6)$$

where

$$\frac{\partial \Delta \phi_i}{\partial \cos(\Delta \phi_i)} = \frac{-1}{\sqrt{1 - \cos^2(\Delta \phi_i)}}. \quad (7)$$

Using intermediate variables as in [1], we obtain

$$\mathbf{p}_1 = \frac{\Delta \mathbf{x}_i \perp (-\Delta \mathbf{x}_{i+1})}{|\Delta \mathbf{x}_i| |\Delta \mathbf{x}_{i+1}|}; \quad \mathbf{p}_2 = \frac{(-\Delta \mathbf{x}_{i+1}) \perp \Delta \mathbf{x}_i}{|\Delta \mathbf{x}_i| |\Delta \mathbf{x}_{i+1}|},$$

where

$$\mathbf{a} \perp \mathbf{b} = \mathbf{a} - \frac{\mathbf{a}^T \mathbf{b}}{|\mathbf{b}|} \frac{\mathbf{b}}{|\mathbf{b}|}.$$

Then, the following simplifications can be applied:

$$\frac{\partial \cos(\Delta \phi_i)}{\partial \mathbf{x}_i} = -\mathbf{p}_1 - \mathbf{p}_2. \quad (8)$$

$$\frac{\partial \cos(\Delta \phi_i)}{\partial \mathbf{x}_{i-1}} = \mathbf{p}_2. \quad (9)$$

$$\frac{\partial \cos(\Delta \phi_i)}{\partial \mathbf{x}_{i+1}} = \mathbf{p}_1. \quad (10)$$

4.2. Additional termination conditions

To prevent path shrinkage that occurs when increasing the number of smoothing iterations, we introduce two new termination conditions for the smoother when the average curvature increases, or when a point in the path exceeds the maximum curvature allowed. After each smoothing iteration, average curvature $\bar{\kappa}$ of the path resulting from the i th smoothing iteration is calculated using Equation (11). Then, smoothing terminates when either $\bar{\kappa}_i > \bar{\kappa}_{i-1}$ or reaching the maximum curvature to preserve the overall smoothness and prevent exceeding

the maximum curvature allowed

$$\bar{\kappa}_i = \sum_{j=0}^{N-1} \frac{\cos^{-1}(\Delta \mathbf{x}_j \cdot \Delta \mathbf{x}_{j+1})}{|\Delta \mathbf{x}_j| |\Delta \mathbf{x}_{j+1}|}. \quad (11)$$

The average curvature is monitored to maximize smoothness. Figure 5 illustrates the average curvature for smoothing. The average curvature of the path has a minimum point during smoothing. As path smoothing progresses, the average curvature decreases. However, after around the 10th iteration, the average curvature starts to gradually increase for approximately 80 iterations until exhibiting a sharp increase where shrinkage becomes evident.

4.3. Near-edge weight

Finally, to overcome the problem of the path smoother not considering the heading direction when approaching the target location, we prevent points close to the target locations from moving considerably. Hence, we introduce weight $w_{e,i}$ for the i -th point in the path. This weight controls how the point variation depends on its location (index) along the path. We use the logistic function to compute $w_{e,i}$ as follows:

$$w_{e,i} = \frac{1}{1 + e^{-(x - x_{sat})}}, \quad (12)$$

where

$$x = \begin{cases} i, & \text{if } i < \frac{L}{2}, \\ L - 1 - i, & \text{otherwise.} \end{cases} \quad (13)$$

L is the length of the path and x_{sat} is the index at which the weight becomes 1. The curvature at index i is checked for the curvature to be satisfied at the initial and target locations of the path.

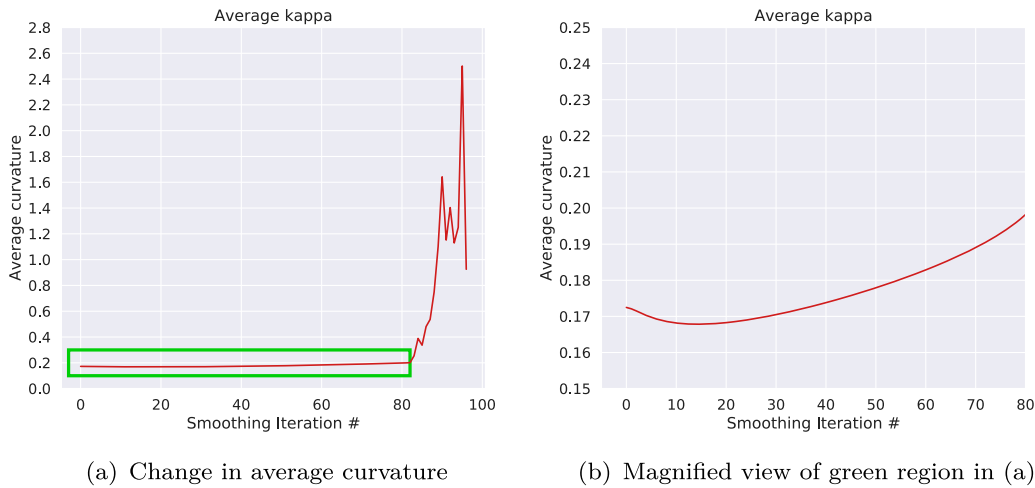


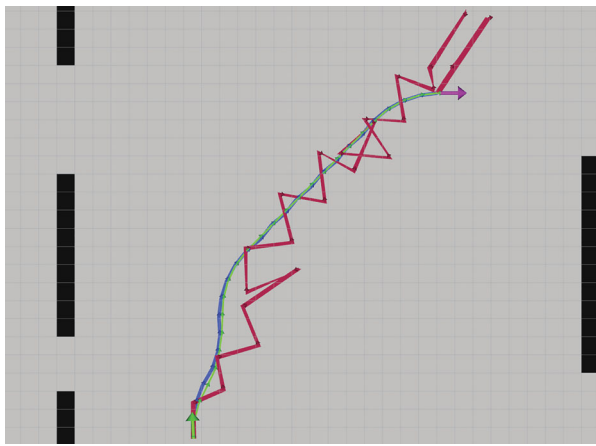
Figure 5. Average curvature for a path where shrinkage occurs. The curvature initially decreases as smoothing progresses but then increases until shrinkage occurs. (a) Change in average curvature. (b) Magnified view of rectangular region in (a).

5. Evaluation of proposed path planner for FFRS

We verified the effectiveness of the proposed modifications in Section 4 to adapt hybrid-state A* to the FFRS. Path searching considered a minimum turning radius of 5 m (i.e. maximum curvature of 0.2). In addition, reverse motion was forbidden, and the grid size of the map was set to 1 m. We simulated path planning on the Robot Operating System (ROS) and used RViz to visualize the generated paths.

5.1. Effect of modifying curvature term

We first modified the curvature term as detailed in Section 4.1. We set w_k to 1, and w_o and w_s to 0.



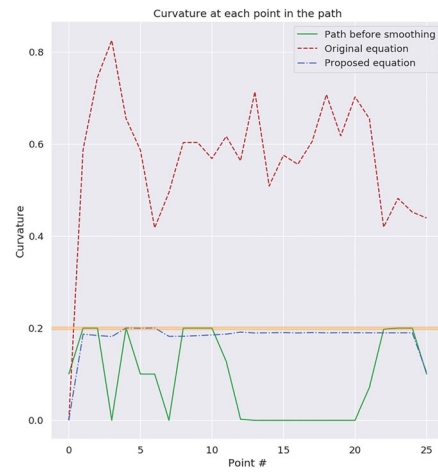
(a) Paths resulting from smoothing

After smoothing for 100 iterations, we obtained the results shown in Figure 6, which presents the paths before smoothing and after smoothing with and without the proposed modifications. Without the modifications, some points along the path do not satisfy the maximum curvature of 0.2, whereas the curvature complies with the limit using the proposed modifications by retrieving an almost constant curvature of 0.2.

5.2. Shrinkage prevention

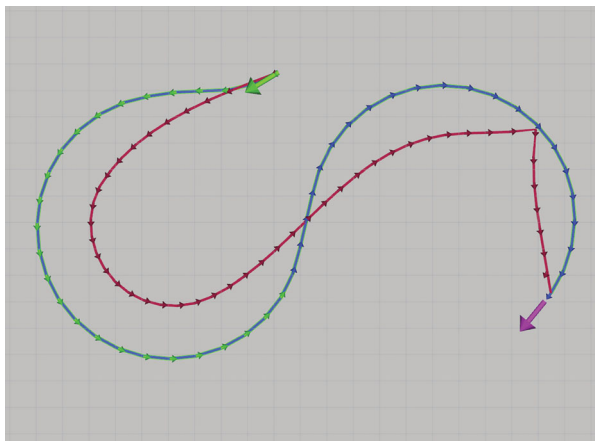
We then executed smoothing for the green path in Figure 7.

The number of iterations was set to 100 and w_o , w_k , w_s to 0.1. The generated path smoother without

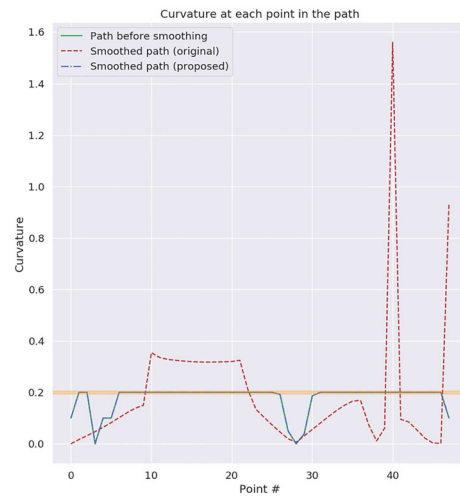


(b) Curvature along path

Figure 6. Proposed modifications to the curvature term for avoiding kinks. Kinking is observed before changing the curvature equation, but it is prevented applying the proposed modifications to the curvature equation. The horizontal line at 0.2 in (b) indicates the maximum curvature allowed. (a) Paths resulting from smoothing. (b) Curvature along path.



(a) Paths resulting from smoothing



(b) Curvature at each point along path

Figure 7. Smoothing before and after applying the proposed termination conditions. Shrinkage is observed before changing the termination conditions for 100 iterations, but is avoided after introducing the termination conditions. The horizontal line at 0.2 in (b) indicates the maximum curvature allowed. (a) Paths resulting from smoothing. (b) Curvature at each point along path.

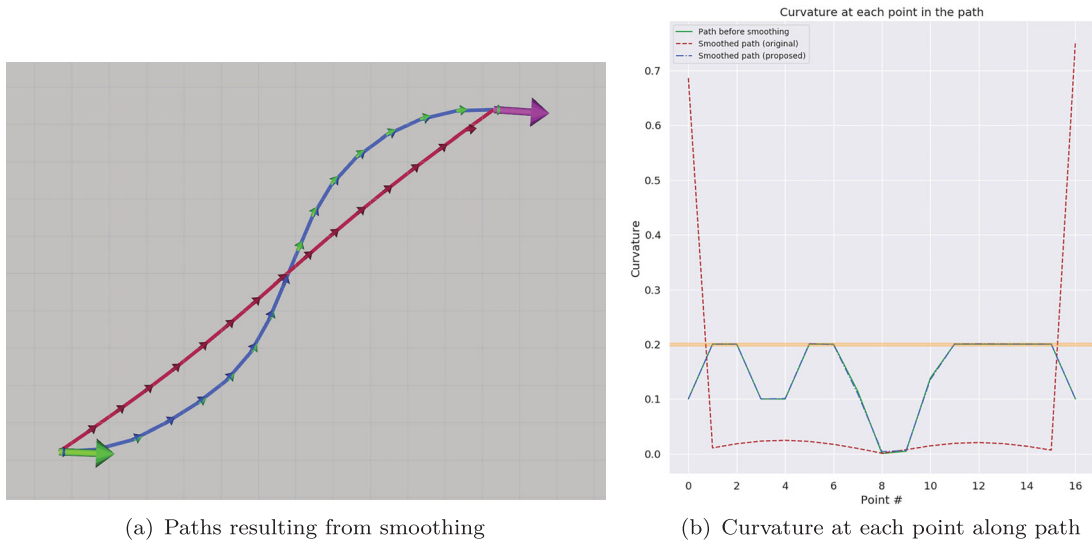


Figure 8. Smoothing before and after introducing the proposed near-edge weight. The path does not exhibit the appropriate heading without the weight, but a smooth connection is achieved from the initial to the target location after introducing the near-edge weight. The horizontal line at 0.2 in (b) indicates the maximum curvature allowed. (a) Paths resulting from smoothing. (b) Curvature at each point along path.

the proposed termination conditions is not smooth and would be extremely difficult for the robot to follow. The curvature plot in Figure 7(b) indicates that the curvature can reach notably above the maximum curvature limit of 0.2 (horizontal line at 0.2 in Figure 7(b)). On the other hand, the path using the proposed termination conditions does not shrink, and the curvature at each point along the path is below the limit.

5.3. Effect of the near-edge weight

Figure 8 shows the path before and after introducing the near-edge weight for $x_{sat} = 3$. Without the near-edge weight, the path approaches the target location at an angle that does not satisfy the maximum curvature. Hence, the robot would not reach the target location facing the specified heading. On the other hand, by applying the proposed near-edge weight to prevent the points at the edge from excessively varying, the curvatures near the initial and target locations do not violate the limit, enabling the robot to approach the target location in the specified heading for effective firefighting.

Table 1. Summary of 10,000 simulation results: the proposed modifications improved the performance of the smoother.

	Original	Proposed
Number of erroneous paths	8762 (87.6%)	0 (0%)
Number of erroneous points	43716 (2.4%)	0 (0%)
Curvature variation	0.021 ± 0.202	0.015 ± 0.034
Difference from target angle (°)	18.81 ± 19.90	6.50 ± 1.91
Computation time per path (s)	3.85 ± 2.86	2.22 ± 1.64

5.4. Simulation experiments

We conducted simulations for multiple paths to verify the effect of the three proposed modifications. Path planning and smoothing were conducted for 10,000 randomly chosen initial/target locations and headings in a virtual experiment field. The size of the virtual field was 300×150 m, which is approximately the designed area for



Figure 9. Area used for the field experiment. The grid map is overlaid on the photograph, and the initial and target locations, obstacles, and generated path are indicated within the grid. The aerial view was retrieved from Google Maps.

FFRS operation. The maximum curvature allowed was set to 0.2. A path was determined for every pair of initial and target locations. The total number of points from the 10,000 paths was 1,819,185. Table 1 lists the simulation results.

The proposed method suitably generated paths satisfying the maximum curvature, verified from the zero erroneous paths. On the other hand, 87.6% of the paths

generated using the original method (i.e. conventional hybrid-state A* algorithm) had at least one point in the path exceeding the maximum curvature. Both the modified curvature term and the termination conditions contributed to eliminate points with excessive curvature.

The proposed method also retrieved smoother paths, as verified from the smaller curvature variation compared with the original method. The condition for



Figure 10. Autonomous navigation of the water-shooting robot using the proposed path planner. The generated path (thick line) was suitable to be followed by the robot (thin line). The path planner generated the trajectory for the water-shooting robot in real time. The robot autonomously navigated from the initial to the target location while avoiding obstacles.

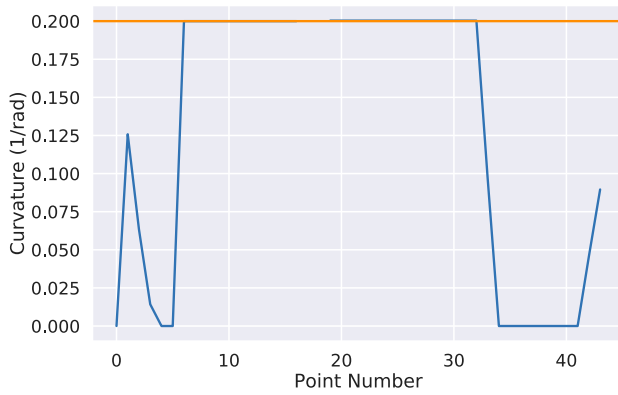


Figure 11. Curvature along the path during field experiment. The maximum curvature requirement of 0.2 is satisfied throughout the path.

smoothing to terminate at the minimum average curvature contributed to generating smoother paths.

The difference between the specified and path target angles was smaller using the proposed method than the original method because of the introduced weight. As the yaw of the water-shooting cannon is $\pm 60^\circ$, both the original and proposed methods retrieved paths within this limit. However, the proposed method achieved a smaller difference from the specified heading, which is more favorable for effective firefighting.

The average computation time per path decreased using the proposed method compared with the original one. This is attributed to the termination conditions. Appropriate termination conditions improve smoothing by reducing the number of iterations and consequently the computation time per path.

5.5. Field experiments

We conducted experiments on the actual water-shooting robot of our FFRS to verify the trajectory tracking of a generated path. The proposed path planner was implemented on the water-shooting robot and evaluated in an area of approximately 90×50 m, whose aerial view is shown in Figure 9. Figure 10 shows a sequence of the water-shooting robot executing the generated path. The smooth trajectory was completed by the robot in approximately 80 s. At each point in the path, the maximum curvature constraint of 0.2 was satisfied, as shown in Figure 11.

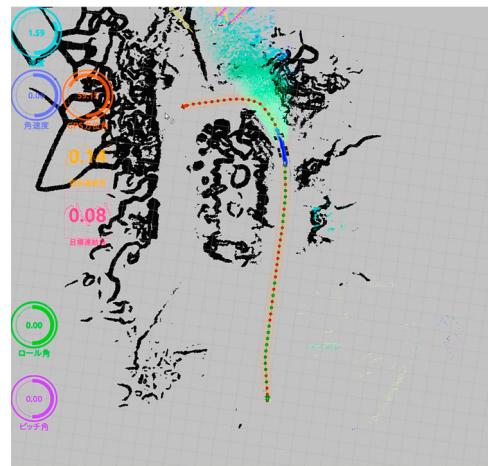
6. Discussion

Besides the FFRS robots, the proposed path planner can be applied to robots with different steering mechanisms such as electric vehicles with steering wheel and articulated dump trucks with skid-steering. Figure 12 shows a dump trucks autonomously driving through the trajectory retrieved from the proposed path planner. The requirements of path planning for the dump truck differ from those for the FFRS. Nevertheless, the maximum curvature should be maintained and the path should lead to a specified heading at the target location. We found that the proposed path planner is effective for autonomous driving of a dump truck.

An improvement of the path planner can be achieved by adding geometric or parametric smoothing after the CG smoother to ensure parametrically continuous trajectories. The conjugate nature of the CG smoother on the



(a) Autonomous navigation of a 40-ton dump truck



(b) Smooth trajectory obtained from the proposed path planner

Figure 12. Application of the proposed path planning to an autonomous vehicle with different mechanism and specifications. The generated path was autonomously executed by the dump truck. The suitable navigation confirmed that the proposed path planner can generate paths suitable not only for steering-type robots but also skid-steering robots. (a) Autonomous navigation of a 40-ton dump truck. (b) Smooth trajectory obtained from the proposed path planner.

one hand considers multiple terms that are not limited to the trajectory geometry (e.g. distance from obstacles). On the other hand, the resulting path may not be geometrically continuous as it is with Bézier curves and B-splines.

Further improvement can be achieved by modifying the smoothness term such that the angular variation over several path segments is calculated instead of being calculated over adjacent segments. This can help to increase the overall smoothness of the path by reducing the number of macroscopic turns.

7. Conclusion

We propose a path planner that can generate suitable paths for the autonomous navigation of water-shooting and hose-extension robots of the FFRS. We developed three modifications to hybrid-state A* for generating paths that meet the requirements of these robots. The resulting path planner implements these modifications and smoothing based on the conjugate gradient descent. The main contributions of this paper are summarized as follows:

- (1) Generated paths satisfy the specified maximum curvature.
→ *Method*: the curvature term of the smoother was examined and corrected.
- (2) Generated paths are smooth with no kinks.
→ *Method*: termination conditions were introduced to increase the smoothness of the path while maintaining the maximum curvature.
- (3) Generated paths reach the target location with the specified heading.
→ *Method*: a weight was included into the objective function to control the movement caused by smoothing.

We evaluated the path planner on an actual water-shooting robot and in a simulated environment. In addition, we verified qualitatively and quantitatively that the proposed smoother can generate smooth trajectories that satisfy the maximum curvature limit, even for vehicles with other types of mechanisms, such as skid-steering.

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes on contributors

Naoki Mizuno received his BS degree in Computer Engineering from Miami University, Ohio, in 2016 and his MS degree in Information Science from Tohoku University, Japan, in 2019.

His research interests are mainly in motion planning and SLAM for mobile robots.

Kazunori Ohno received BS, MS, and doctor of engineering from Tsukuba University in 1999, 2001, and 2004, respectively. He was a researcher at Kobe University from 2004 to 2005. He became an assistant professor and a lecturer at Tohoku University from 2005 to 2010. He became a researcher at Japan Science and Technology Agency researcher PRESTO from 2008 to 2012. He is as an associate professor of New Industry Creation Hatchery Center (NICHe), Tohoku University since 2012 and a visiting researcher of RIKEN AIP since 2017. His research fields are field robotics, disaster response robotics, and robot intelligence. His main works are tracked vehicles Quince, which was used for inspection of Fukushima Nuclear Power Plant, Cyber-enhanced Rescue Canines, and Autonomous Outdoor Carriers of TOYOTA Motor East Japan Co. He served PI of Cyber-enhanced Rescue Canine of ImPACT Tough Robotics Challenge and PI of NEDO SIP Aged Bridge Inspection using UAVs. He establishes TC on Data Enhanced Robotics of RSJ in 2012. He received Kiso awards at 2008 and 2012, RSJ research award at 2005, best paper finalist at SII2010 and SSRR 2008, 2009 and 2011, SICE SI award at 2006, 2009 and 2010, and the best presentation award at FAN2006. He is a member of Robotics Society of Japan (RSJ), Japan Society of Mechanical engineers (JSME), virtual Reality Society of Japan (VRSJ), and IEEE.

Ryunosuke Hamada received his B.E. from Osaka University in Osaka, Japan, in 2011, and his M.E. and Ph.D. from Nara Institute of Science and Technology in Nara, Japan, in 2013 and 2016, respectively. He works at Tohoku University as an assistant professor from 2016. His research interests include time series modeling based on latent variable models, machine learning, and signal processing, and its application to model human/animal behaviors.

Hiroyoshi Kojima received his ME degree from the University of Tsukuba, in 2004. He currently works at Mitsubishi Heavy Industries, Ltd. as a mechanical and software engineer.

Jun Fujita received Master of Engineering in mechanical systems engineering course from Kanazawa University in 1997. In the same year, he joined Mitsubishi Heavy Industries, Ltd. He is designing robots used to maintain Nuclear Power Plants. He acquired a P.E.Jp (Mechanical Engineering) in 2010.

Hisanori Amano graduated from the Osaka Prefectural University in 1988. He received Doctor of Informatics from Kyoto University in 2004. He is the senior executive researcher of National Research Institute of Fire and Disaster, Fire and Disaster Management Agency, Japan, since 2014. He is a member of RSJ, IEEE, SICE, and JSME.

Thomas Westfechtel is a researcher at New Industry Creation Hatchery Center (NICHe), Tohoku University. He received his MSc degree in Electrical Engineering from Paderborn University, Germany, in 2015 and his PhD degree from the Tohoku University, Japan, in 2019. His research interest are mainly perception and knowledge acquisition for mobile robots.

Takahiro Suzuki received BS, MS, and Doctor of Engineering from Department of Mechano-Informatics at the University of Tokyo, in 1993, 1995, and 1998, respectively. He became a Lecturer and an Associate Professor in the Institute of Industrial

Science (IIS), the University of Tokyo in 1998, 2000, respectively. He was an Associate Professor in the Interfaculty Initiative in Information Studies (III) at the same university from 2004 to 2010. In 2010, he temporarily moved to Nagasaki Prefectural Government as a Director General specially for EV&ITS promotion from 2010 to 2013, and simultaneously he was a Guest Associate Professor in IIS, Univ. of Tokyo. He moved back to IIS, Univ. of Tokyo in 2013, and is a Professor and Vice Director of New Industry Creation Hatchery Center (NICHe), Tohoku University since 2014. (He is also a Guest Professor in Nagasaki Institute of Applied Science (NiAS) from 2013.) His research fields are nonlinear robot dynamics and control, nonholonomic system, ITS (intelligent transport systems), science and technology communication, and social diffusion of advanced technology. He is a member of RSJ, JSME, JSAE (Society of Automotive Engineers in Japan), SICE, IEEE, ITS Japan, and Japan Society for Research Policy and Innovation Management. He is also a Director of Association for the Promotion of Electric Vehicles (APEV).

Satoshi Tadokoro graduated from the University of Tokyo in 1984. He was an associate professor in Kobe University in 1993–2005, and is a professor of Tohoku University since 2005, and was a vice/deputy dean in 2012–2014, and is the director of Tough Cyberphysical AI Research Center since 2019. He is the president of International Rescue System Institute since 2002 and was the IEEE RAS President in 2016–2017. He served as a program manager of MEXT DDT Project on rescue robotics in 2002–2007, and was a project manager of Japan Cabinet Office ImPACT Project on disaster robotics in 2014–2019 having 62 international PIs. His team developed various rescue robots, two of which called Quince and Active Scope Camera are well known because they were used in disasters such as in nuclear reactor buildings of the Fukushima-Daiichi Nuclear Power Plant Accident. IEEE Fellow, RSJ Fellow, JSME Fellow, and SICE Fellow.

ORCID

Naoki Mizuno  <http://orcid.org/0000-0002-4859-2370>

References

- [1] Dolgov D, Thrun S, Montemerlo M, et al. Path planning for autonomous vehicles in unknown semi-structured environments. *Int J Rob Res*. 2010;29(5):485–501.
- [2] LaValle SM. Rapidly-exploring random trees: a new tool for path planning. Computer Science Department, Iowa State University; 1998 October. tR 98-11.
- [3] Kuffner JJ, LaValle SM. Rrt-connect: an efficient approach to single-query path planning. *IEEE International Conference on 2000 Robotics and Automation, Proceedings. ICRA'00; Vol. 2. IEEE*. 2000. p. 995–1001.
- [4] Bruce J, Veloso MM. Real-time randomized path planning for robot navigation. *Robot Soccer World Cup*. Springer; 2002. p. 288–295.
- [5] Urmson C, Simmons R. Approaches for heuristically biasing RRT growth. *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Proceedings. 2003 (IROS 2003). Vol. 2. IEEE*. 2003. p. 1178–1183.
- [6] Takahashi O, Schilling RJ. Motion planning in a plane using generalized Voronoi diagrams. *IEEE Trans Rob Autom*. 1989;5(2):143–150.
- [7] Hart PE, Nilsson NJ, Raphael B. Correction to “A formal basis for the heuristic determination of minimum cost paths”. *SIGART Bull*. 1972;37:28–29.
- [8] Stentz A, Mellon IC. Optimal and efficient path planning for unknown and dynamic environments. *Int J Rob Autom*. 1993;10:89–100.
- [9] Koenig S, Likhachev M. Fast replanning for navigation in unknown terrain. *IEEE Trans Robot*. 2005;21(3):354–363.
- [10] Ferguson D, Stentz AT. The field d* algorithm for improved path planning and replanning in uniform and non-uniform cost environments. *Carnegie Mellon University. Pittsburgh, PA*. 2005 June. Tech Rep CMU-RI-TR-05-19.
- [11] Koenig S, Likhachev M, Furcy D. Lifelong planning a*. *Artif Intell*. 2004;155(1–2):93–146.
- [12] Reeds JA, Shepp LA. Optimal paths for a car that goes both forwards and backwards. *Pacific J Math*. 1990;145(2):367–393. <https://projecteuclid.org/443/euclid.pjm/1102645450>.
- [13] Yang K, Sukkarieh S. An analytical continuous-curvature path-smoothing algorithm. *IEEE Trans Robot*. 2010;26(3):561–568.
- [14] Schoenberg IJ. Contributions to the problem of approximation of equidistant data by analytic functions: part b on the problem of osculatory interpolation. a second class of analytic approximation formulae. *Quart Appl Math*. 1946;4(2):112–141. <http://www.jstor.org/stable/43633544>.
- [15] Nikolos IK, Valavanis KP, Tsourveloudis NC, et al. Evolutionary algorithm based offline/online path planner for UAV navigation. *IEEE Trans Syst Man Cybern Part B (Cybern)*. 2003;33(6):898–912.
- [16] Bobrow JE. Optimal robot plant planning using the minimum-time criterion. *IEEE J Robot Autom*. 1988;4(4):443–450.
- [17] Elbanhawi M, Simic M, Jazar RN. Continuous path smoothing for car-like robots using b-spline curves. *J Intell Rob Syst*. 2015;80(1):23–56.
- [18] Kurzer K. Hybrid a* path planner for the kth research concept vehicle. 2016. Accessed 2018-09-09. Available from: https://github.com/karlrurzer/path_planner.