

Stable Autonomous Robotic Wheelchair Navigation in the Environment With Slope Way

Chaoqun Wang , *Member, IEEE*, Min Xia , *Member, IEEE*, and Max Q.-H. Meng , *Fellow, IEEE*

Abstract—In this article, we present a path planning approach that is capable of generating a feasible trajectory for stable robotic wheelchair navigation in the environment with slope way. Firstly, the environment is modeled by a lightweight navigation map, with which the proposed sampling-based path planning scheme with a modified extension function can generate a feasible path. Then, the path is further optimized by the proposed utility function involving the human comfort and the path cost. To improve the searching efficiency of an optimal trajectory, we present an adaptive weighting Gaussian Mixture Model (GMM) based sampling strategy. Particularly, the weights of the components in GMM are adjusted adaptively in the planning process. It is also worth noting that the proposed sampling-based planning paradigm can indicate the unsafe regions in the navigation map, which forms a traversable map and further guarantees the safety of the wheelchair robot navigation. Furthermore, the effectiveness and the efficiency of the proposed path planning method are verified in both simulation and real-world experiments.

Index Terms—Path planning, autonomous vehicle, navigation, robot motion.

I. INTRODUCTION

THE single passenger vehicle that incorporates robotic techniques represents a new trend of personal intelligent transport system [1]. As a representative of such vehicles, the intelligent wheelchair robot has attracted much attention in recent years [2], [3]. Unlike the semi-autonomous wheelchair that sometimes requires the operational skill of the human pilot, the fully autonomous wheelchair robot can drastically reduce possible crashes due to lack of operational experience and incomplete perception of the environment.

Nowadays, fully autonomous wheelchair robots are frequently operating in various environments. These environments

are typically structured but might contain level grounds with different heights. Fortunately, more and more environments are designed and built for wheelchairs. They have, for example, provided the slope way for the wheelchair to navigate through. However, it is still challenging for stable wheelchair navigation to identify the traversable area and reach a target through the slope safely. This research aims to generate a safe and human-comfortable path that leads the wheelchair robot to achieve autonomous navigation in these challenging environments.

For the wheelchair robot navigation, one fundamental and critical part is a suitable representation of the complex environment. Recently, a vast amount of research has been devoted to developing various maps [4], [5]. Different maps have their own merits for different applications [6], [7]. However, for robot navigation in environments with sloped way, these maps either provide limited information for safe navigation, or are too redundant and suffer from high computational complexity. Little work has focused on developing a lightweight map that can represent terrain traversability and explicitly indicate the hazardous areas for robot navigation.

With a suitable environment map, path planning strategy is one key technique for achieving stable robot navigation and has been well investigated over the past decades [8]–[10]. Generally, the goal for path planning is to generate a feasible path from the start location to the desired location regarding the vehicle motion or energy constraints. However, for the passenger vehicle that involves multiple subjective evaluation criteria for the navigation performance, few efforts have been put in improving the vehicle safety and the human comfort on sloped terrain to achieve stable wheelchair robot navigation.

In this article, we present a stable navigation strategy for the wheelchair robot in the environment with slope way. The system diagram is shown in Fig. 1. In particular, instead of considering the human comfort in the socially compliant navigation paradigm [11], we focus on the human comfort that results from the vehicle stability on the slope. To the best of our knowledge, this is the first path planning method towards stable wheelchair navigation considering the human comfort on the sloped terrain. Our contributions mainly lie in the following aspects:

- 1) We present a path planning method that can generate a feasible path with a new vehicle stability measure considering the human comfort for wheelchair robot navigation on the slope.
- 2) With the proposed planning scheme, different terrains can be efficiently identified, which forms a traversable map and facilitates safe wheelchair navigation.

Chaoqun Wang is with the Department of Electrical Engineering, The Chinese University of Hong Kong, Shatin 999077, Hong Kong (e-mail: zychaoqun@gmail.com).

Min Xia is with Engineering Department, Lancaster University, Lancashire LA1 4YW, U.K. (e-mail: m.xia3@lancaster.ac.uk).

Max Q.-H. Meng is with the Department of Electronic and Electrical Engineering, Southern University of Science and Technology, Shenzhen 518055, China, with the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, and also with the Shenzhen Research Institute, Chinese University of Hong Kong, Shenzhen 518057, China (e-mail: qhmeng@ee.cuhk.edu.hk).

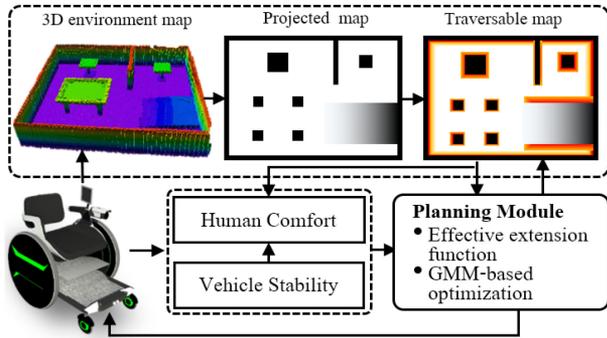


Fig. 1. Schematic diagram of the proposed autonomous navigation scheme. A traversable map for safe navigation is obtained by using the proposed path planning module. The human comfort is related to the vehicle stability and the distance to the hazardous area.

- 3) An adaptive weighting GMM based approach is proposed to improve the efficiency of optimal path planning and the traversable map generation.

The necessity of the proposed framework is corroborated in both simulation and real-world experiments by comparing the method with state of the arts. The experimental results demonstrate that our approach can efficiently find safe and human-comfortable paths in complex environments.¹

The rest of the paper is organized as follows. In Section II, we review the literature related to our study. Then, the preliminaries and formulations of this study are given in Section III. The proposed path planning strategy is introduced in Section IV and Section V. Experiments and results are reported in Section VI. Finally, we conclude the study and give the directions of the future work in Section VII.

II. RELATED WORK

We review the literature on environment representation method and path planning strategy, which are highly related to our study and are two key modules for achieving stable robot navigation on sloped terrain.

A. Environment Representation

It is the precondition for the robot to get an accurate environment model for stable navigation. Recently, much effort has been paid towards robustly perceiving the environment, distinguishing different road terrains (e.g., flat road, slope way), and evaluating their traversability. In [12], a vision-based terrain reconstruction method was proposed for building the map of the environment. Haselich *et al.* [13] proposed a probabilistic method to fuse 3D lidar and camera image data for classifying different road terrains. Santamaria-Navarro *et al.* [14] proposed to learn the traversable area from 3D point clouds. These methods focus more on the terrain classification methods. They generally partition the terrains into traversable and non-traversable areas without considering the properties of different terrains (e.g., slope, roughness). Moreover, the classification results have

not been wrapped into a suitable map, with which the robot can plan a feasible path considering the vehicle stability and mobility on diverse terrains for safe navigation.

Over the years, various kinds of maps are proposed for encoding the information of the environment for the navigation purpose [15]. The 2D grid map [16] that can represent the obstacles and collision-free areas is widely adopted. However, it is inadequate for robot navigation in complex environments with diverse terrains. A hybrid-map with different 2D map layers was used in [17] for robot navigation in complex environments. Gu *et al.* [18] suggested to utilizing a 2.5D map to assess the traversability of different terrains. Differently, in [5], a 3D mesh map was ready for online path planning in rough terrain by a map generation method based on 3D point cloud. More recently, Sock *et al.* [19] proposed a traversable map that simply distinguished different terrains in a probabilistic manner. These methods can indicate the traversable area in the environment, yet the safe region for robot navigation is not considered. Based on the 3D elevation grid map presented in [20], we develop a traversable map that can clearly indicate the unsafe areas for safe vehicle navigation.

B. Path Planning Techniques

With different map representations, there are lots of efforts towards robotic path planning in environments with diverse terrains. By using a B-cubic patch map, a near-optimal path planner for rough terrain navigation was proposed in [21], which adopted a layered control strategy that separated the trajectory planning and path following. Singh *et al.* [22] proposed to use a non-linear time scaling technique to generate an optimal path on uneven terrain, with a special focus on avoiding possible slip of the vehicle. These methods are mainly designed for unmanned vehicles. For passenger vehicles, human comfort should be particularly emphasized, necessitating the development of path planners incorporating this factor.

Recently, Morales *et al.* [11] proposed to involve passenger comfort in the wheelchair navigation. Furthermore, Guzzi *et al.* [23] devoted to planning a comfortable and short path for the wheelchair robot. However, these methods focus more on the human comfort in socially compliant navigation. The human comfort on sloped terrain, which varies with the vehicle stability, is seldom considered in the literature. Practically, the vehicle stability analysis on the slope has been well investigated. The force-angle stability analysis proposed in [24] provided a measure for possible tip-over of the vehicle subject to internal or external forces. This measure is widely adopted for mobile robot or wheelchair stability analysis [25], while it does not take human comfort into account. In this study, we present a stability measure that considers human comfort on the slope way.

Our path planning strategy builds upon the recent developments of the sampling-based path planning method [26], which are useful tools for incorporating various constraints. Despite their generality, the applications of these methods in challenging environments with sloped terrain are far from straightforward. In light of this challenge, Devaurs *et al.* [27] introduced a T-RRT* method, with which a low-cost trajectory on uneven terrain could

¹Video demonstration: <https://youtu.be/ywjQ8HBC79U>

be generated. Nevertheless, vehicle stability constraints are not involved in this method. Norouzi *et al.* [28] proposed to incorporate the vehicle stability into the planning framework while it is not applicable to the passenger wheelchair navigation involving human factors. In contrast to the existing sampling-based path planning methods, the proposed method can generate a feasible path that improves human comfort on the slope way.

III. PRELIMINARIES

In this study, we aim to plan a safe and human-comfortable trajectory for the wheelchair robot for its navigation in environments with slope ways. The developed traversable map for robot navigation is built on top of the elevation grid map \mathcal{M}_g [20], which is widely used for the 3D environment representation. For a grid $m_j \in \mathcal{M}_g$, $h(m_j)$ denotes its elevation. We project the elevation map into a 2D grid map $\check{\mathcal{M}}_g$, where for $m_j \in \check{\mathcal{M}}_g$, $\check{h}(m_j) = h(m_j)$. The path planning is performed directly on the projected 2D map $\check{\mathcal{M}}_g$ that further takes into consideration the vehicle safety.

The path for the robot is denoted as ξ , consisting of a set of states $s \in \mathcal{X}$, where \mathcal{X} represents the state space. We denote \mathcal{X}_h as the hazardous area for the vehicle navigation. Specially, for a state s , the heading of the robot is denoted as \bar{r} and the stability of the robot is denoted as γ . For robot navigation across different terrains, given the start state s_i and the target state s_t , the total stability of a trajectory ξ is

$$\Gamma(\xi) = \sum_{s \in \xi} \gamma(s). \quad (1)$$

The human comfort in this study is highly related to the vehicle stability. The optimal path planning for the wheelchair robot in the focused environment is then formulated as the problem of improving the stability of the wheelchair while minimizing the motion cost. Accordingly, the objective of the wheelchair robot path planning is given by

$$\begin{aligned} \min \quad & f(\Gamma(\xi), \mathcal{L}(\xi)) \\ \text{s.t.} \quad & \xi \in \mathcal{X} \\ & F(\xi) \leq 0 \\ & \|\xi_i - s\| \geq \kappa, \forall \xi_i \in \xi, s \in \mathcal{X}_h, \end{aligned} \quad (2)$$

where $f(\cdot)$ is the objective function with respect to the vehicle stability on ξ and the path length. $\mathcal{L}(\xi)$ is the distance measure of the trajectory ξ . $F(\xi) \leq 0$ ensures that the trajectory satisfies the robot motion constraints, meaning that the ξ should be a feasible trajectory for the robot to execute. The trajectory should keep a distance κ to the hazardous area in the navigation. The optimization problem is therefore the one that is highly related to the form of the stability function $\Gamma(\cdot)$ and the path length $\mathcal{L}(\cdot)$. In particular, the human comfort is the quantity related to the vehicle stability. As aforementioned, the quantities regarding the social factors are out of the scope of this study. In what follows, we detail the vehicle stability analysis concerning the human comfort and the path generation by using the sampling-based path planning scheme.

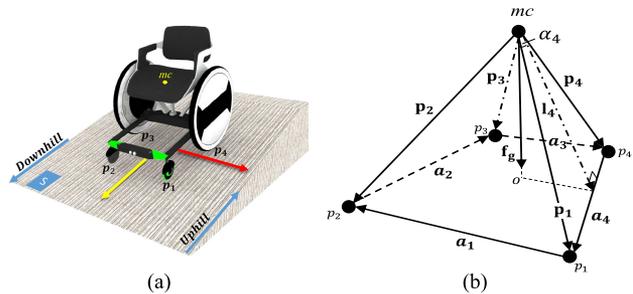


Fig. 2. Wheelchair stability analysis on a slope. (a) Wheelchair on the slope. (b) Vehicle stability pyramid.

IV. VEHICLE STABILITY ANALYSIS

Firstly, to analyze the stability of the wheelchair robot, the stability pyramid [24], [29] is utilized. The stability of the wheelchair inherently indicates the vehicle tip-over risk when it moves on the slope surface. Without loss of generality, we use the wheelchair model with four wheels. As shown in Fig. 2(a), $p_i (i = 1, 2, 3, 4)$ represents the Point of Contact (PoC) of the wheelchair with the slope surface. mc denotes the center of mass of the integrated human-wheelchair system. Then, the stability of the wheelchair is primarily analyzed by using the stability pyramid shown in Fig. 2(b), where $\mathbf{p}_i (i = 1, 2, 3, 4)$ is the vector that connects the mc and the PoC. Denote $\mathbf{a}_i (i = 1, 2, 3, 4)$ as the tip-over axis that

$$\mathbf{a}_i = \mathbf{p}_{i+1} - \mathbf{p}_i, i = 1, 2, 3. \quad (3)$$

Particularly, $\mathbf{a}_4 = \mathbf{p}_1 - \mathbf{p}_4$. As shown in Fig. 2(b), the tip-over normal axis, \mathbf{l}_i , which intersects with the mc , is obtained with

$$\mathbf{l}_i = (\mathbf{I} - \hat{\mathbf{a}}_i \hat{\mathbf{a}}_i^T) \cdot \mathbf{p}_{i+1}, i = 1, 2, 3. \quad (4)$$

In particular, $\mathbf{l}_4 = (\mathbf{I} - \hat{\mathbf{a}}_4 \hat{\mathbf{a}}_4^T) \cdot \mathbf{p}_1$. \mathbf{I} is a 3×3 unit matrix and $\hat{\mathbf{a}}_i = \mathbf{a}_i / \|\mathbf{a}_i\|$ is the unit vector along \mathbf{a}_i . The angle between the gravity vector \mathbf{f}_g and \mathbf{l}_i is denoted as the stability angle that is

$$\alpha_i = \epsilon_i \cos^{-1}(\hat{\mathbf{f}}_g \hat{\mathbf{l}}_i), i = 1, 2, 3, 4, \quad (5)$$

where $\hat{\mathbf{f}}_g = \mathbf{f}_g / \|\mathbf{f}_g\|$ and $\hat{\mathbf{l}}_i = \mathbf{l}_i / \|\mathbf{l}_i\|$. ϵ_i is the symbol that

$$\epsilon_i = \begin{cases} +1, & (\hat{\mathbf{l}}_i \times \hat{\mathbf{f}}_g \hat{\mathbf{a}}_i) < 0 \\ -1, & \text{otherwise.} \end{cases} \quad (6)$$

The stability angle is marked in Fig. 2(b) and the minimum stability angle for the vehicle is given by

$$\eta = \min(\alpha_i), i = 1, 2, 3, 4, \quad (7)$$

which provides a bottom line for the stability of the vehicle. The vehicle with $\eta \leq 0$ is extremely unstable and will fall over. To improve the stability of the wheelchair robot, each stability angle α_i should be kept as large as possible while satisfying $\eta > 0$.

In particular, we propose to incorporate the human comfort into the wheelchair stability evaluation. On the one hand, human comfort is highly related to vehicle stability. People will feel uncomfortable if the vehicle has tip-over risk. On the other hand, human comfort is also related to the heading angles of the vehicle

on the slope. To illustrate this, we divide the slope way \mathcal{S} into flat surface primitives. For each $\mathcal{S}_i \in \mathcal{S}$ shown in Fig. 2(a), the slope direction \vec{r} is denoted as

Definition 1: Slope Direction (SD): The direction on \mathcal{S}_i that leads to increase or decrease of the elevation with the highest change rate is the Slope Direction, which is the uphill or downhill direction.

Specially, the yellow arrow in Fig. 2(a) indicates the case when $|\angle \vec{r}, \vec{r}| = 0^\circ$, which is the SD. The red arrow indicates the case when $|\angle \vec{r}, \vec{r}| = 90^\circ$. Considering the wheelchair user guide² and the habit of human sitting on the wheelchair that the vehicle always attempts to climb a slope without lateral rotation, human will feel uncomfortable when the wheelchair is placed laterally with respect to SD, i.e. $|\angle \vec{r}, \vec{r}| \rightarrow 90^\circ, |\angle \vec{r}, \vec{r}| \in [0^\circ, 90^\circ]$, as demonstrated in [30], [31]. In order to improve the human comfort, we propose to emphasize the lateral stability in the vehicle stability function. Correspondingly, we propose a utility function to measure the wheelchair stability on the slope at state s :

$$\gamma(s) = \sum_{i=1}^4 \mu_i \alpha_i, \quad (8)$$

where the constant μ is used to emphasize the lateral stability considering the human comfort and it is then determined by

$$\mu_i = \begin{cases} > 1, & i = 2 \text{ or } 4 \\ 1, & \text{otherwise.} \end{cases} \quad (9)$$

As shown in Fig. 2, with the design of (8), $\gamma(s)$ will have higher value when $i = 2$ or 4 , indicating that the vehicle is placed laterally with respect to SD and making people feel unsafe and uncomfortable. Therefore the stability angle α_2 and α_4 will have greater impact than α_1 and α_3 in (8). The $\gamma(s)$ is independent among different states. Therefore, the total stability of trajectory ξ for the wheelchair robot to navigate in the environment with slope way can be calculated by (1) with respect to $\eta > 0$. This quantity is then utilized to generate the safe and human-comfortable trajectory for the wheelchair by the sampling-based path planning scheme.

V. SAFE AND HUMAN-COMFORTABLE PATH PLANNING

Based on the analysis of vehicle stability, a sampling-based planning scheme is proposed to generate a path in the investigated environment, following which the robot could safely travel through the slope way and reach the desired target. Additionally, the sampling scheme could help with distinguishing different terrains, thus further ensuring safe robotic navigation. Moreover, to improve the planning efficiency, an adaptive weighting GMM sampling scheme is incorporated into the path planning pipeline.

As typical sampling-based planning schemes, the Rapidly-exploring Random Tree (RRT) and its variants have manifested themselves as useful tools for planning feasible and smooth paths from the robot current location to the goal. The proposed method builds on top of the variant step size RRT method [26],

Algorithm 1: RRT With Variant Step Size.

```

1  $\mathcal{T}$ .initialize ;
2 while a path is not found do
3    $p_r \leftarrow \text{Sampler}()$  ;
4    $p_n \leftarrow \text{Nearest}(p_r, \mathcal{T})$  ;
5    $\zeta \leftarrow \text{Steer}(p_r, p_n)$  ;
6   if collisionCheck( $\zeta$ ) then
7      $\mathcal{T}$ .add( $p_n, \zeta$ )
8   end
9   pathFinding( $\mathcal{T}$ );
10 end
```

which maintains a tree structure \mathcal{T} and consists of a set of key procedures:

Sampling: A sample point p_r yielded by a sample function $\text{Sampler}(\cdot)$ is generated in the collision-free state space \mathcal{X} .

Nearest: The point p_r searches to get the nearest point p_n on the tree \mathcal{T} with the nearest function $\text{Nearest}(\cdot)$.

Steering: Moving from p_r to p_n with a segment ζ with a $\text{Steer}(\cdot)$ function.

Collision checking: The segment ζ is checked through function $\text{collisionCheck}(\cdot)$ to test its feasibility.

Using these functions, the RRT exhibits the general structure outlined in Algorithm 1. This method is not directly applicable to the path planning across different terrains. In this study, we mainly modify the **Collision checking** and **Sampling** modules to get feasible paths in environments with slope ways.

A. Segment Feasibility

Simply conducting segment collision-checking using the method introduced in [26] is inadequate for planning in the environment with smooth slope surface. We propose a feasibility checking process that differs greatly from the standard RRT pipeline. We take the projected 2D elevation grid map $\check{\mathcal{M}}_g$ as the input for the planning method to generate the segment ζ for the tree extension. To verify the feasibility of the segment, we divide it into a set of small segment elements, i.e., $\zeta = \langle l_\zeta^1, l_\zeta^2, \dots, l_\zeta^n \rangle$.

For a segment element $l_\zeta^j \in \zeta$, let p_s^j and p_e^j be the start point and the end point of the segment, respectively. These points p_s^j and p_e^j are the grids in map $\check{\mathcal{M}}_g$ and the elevation of these two grids are $\check{h}(p_s^j)$ and $\check{h}(p_e^j)$. Therefore, the gradient of the segment connecting these two points can be calculated using

$$\nabla_j = \frac{\check{h}(p_e^j) - \check{h}(p_s^j)}{p_e^j - p_s^j}. \quad (10)$$

The indicator of ∇_j is denoted as δ_j that is given by

$$\delta_j = \frac{1}{1 + e^{(|\nabla_j| - \tau) * (+\infty)}}, \quad (11)$$

where τ is a threshold determined by the gradeability of the vehicle. If $|\nabla_j| > \tau$, then $\delta_j = 0$, meaning the segment is too steep for the robot to climb. Then, the feasibility of the segment

²<http://www.alas.wales.nhs.uk/wheelchairs>

Algorithm 2: SegCheck(\cdot).

Input: Elevation map $\tilde{\mathcal{M}}_g$, segment ζ
Output: Segment feasibility F

```

1  $F = True$  ;
2 for  $l_\zeta^j \in \zeta$  do
3    $\nabla_i \leftarrow gradient(l_\zeta^j)$  ;
4   if  $\nabla_j > \tau$  then
5     return  $F = False$  ;
6   end
7    $\Gamma(\zeta) = \Gamma(\zeta) + \Gamma(l_\zeta^j)$  ;
8 end
9  $P \leftarrow e^{-\lambda\Gamma(\zeta)}$  ;
10 if  $Rand[0, 1] < P$  then
11    $F = False$ 
12 end

```

ζ is calculated by

$$\phi(\zeta) = \prod_{j=1}^n \delta_j. \quad (12)$$

In particular, if a segment ζ satisfies that $\phi(\zeta) = 0$, then ζ is strictly unfeasible. Then, the segment will not be used for extending the tree. Notably, the proposed method devotes to generating a trajectory that is flat with the least fluctuation. To this end, we employ a utility function for determining whether a segment can be accepted or not in a probabilistic manner. The probability for accepting the segment ζ as a branch of the tree \mathcal{T} is given by

$$P = \phi(\zeta)e^{-\lambda\Gamma(\zeta)}, \quad (13)$$

where λ is a predefined parameter in different application cases. With (13), if a segment is strictly unfeasible, the probability $P = 0$ and it will be discarded. While for feasible segments, the segment indicating higher vehicle stability is of higher probability to be accepted for extending the tree.

The feasibility checking of a segment to be accepted as the branch of the tree is depicted in Algorithm 2. Line 2–8 depicts the checking process of the traversability of the segment. Function $gradient(\cdot)$ calculates the gradient of the separated segments on ζ . The segment that fails to past the test will be directly discarded. If a segment is traversable, we propose to select the segment that is more safe and stable for the robot to follow as the branch for the tree extension, as indicated in Line 9–12 in Algorithm 2, where the function $Rand[0, 1]$ accounts for generating a random point in $[0, 1]$.

Note that the proposed path planning method is built upon the RRT path planner with the modification of the tree extension step. Therefore, the proposed method is probabilistic complete, which is directly inherited from the RRT planner.

B. Traversable Map Generation

As shown in Fig. 3(a)–(b), for 2D navigation, a cost map can be generated on a 2D grid map by inflating the obstacles. It clearly indicates the clearance region and ensures safe robotic

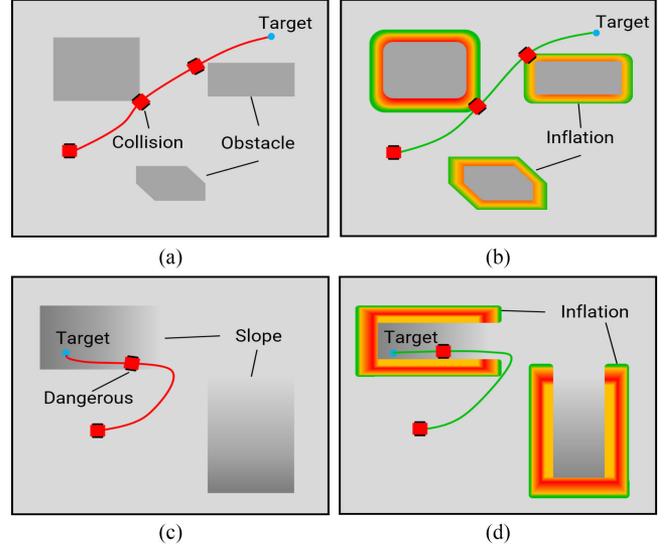


Fig. 3. Different maps for navigation. (a) 2D grid map. (b) 2D Cost map. (c) Elevation map. (d) Traversable map.

navigation in 2D environments. In contrast to the 2D cost map generation, it is hard to explicitly mark the hazardous areas on the elevation grid map shown in Fig. 3(c), as the regions with high elevation values cannot be simply regarded as obstacles. For the fixed chassis wheelchair robot, the obstacle typically refers to the object with high edges or steep slopes, such as the staircase and the steep slope. It is the non-traversable region where the robot cannot navigate through. The hazardous regions for robot navigation are identified as the obstacle areas and the traversable areas that may lead to vehicle tip-over, such as the area near the edge of the slope way.

To generate a safe traversable map for the wheelchair robot, we present a traversable map generation method regarding vehicle motion constraints and hazardous areas. The map $\tilde{\mathcal{M}}_g$, as shown in Fig. 3(d), is generated by exploiting the underlying properties of failure cases in the segment feasibility test in (12). Initially, the traversable map is the same with that of the projected 2D elevation map $\tilde{\mathcal{M}}_g$ shown in Fig. 3(c). Afterword, the traversable map is generated incrementally along with the path planning progress.

To illustrate the generation of the traversable map in detail, we define the unfeasible segment as

Definition 2: Unfeasible Segment (US): The segment ζ satisfies $\phi(\zeta) = 0$ is defined as unfeasible segment and is marked as ζ^- .

If the segment is ζ^- , as aforementioned, it is regarded as unfeasible for the robot to follow and will not be added to the existing tree \mathcal{T} . It is worth noting that the segment ζ^- implicitly indicates the region that is untraversable or hazardous in the map. As shown in Fig. 4, the ζ^- lies in the unsafe region. Therefore, these segments that fail the feasibility test can be used for generating a traversable map towards safe wheelchair navigation.

Specifically, we use two endpoints and the midpoint of the segment ζ^- as the hazardous area indicators, which forms the

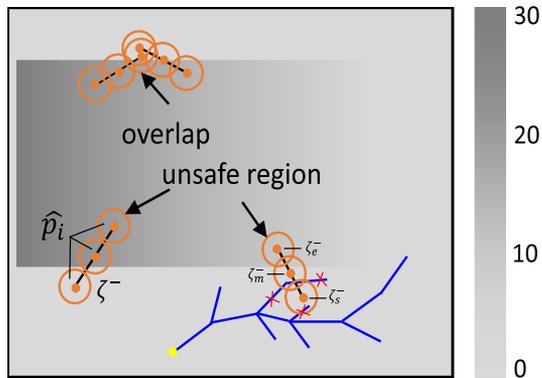


Fig. 4. Traversable map generation using the indicators generated by the sampling-based path planning scheme.

indicator set $\mathcal{I} = \langle \zeta_s^-, \zeta_m^-, \zeta_e^- \rangle$. For a point $\bar{p} \in \mathcal{I}$, the corresponding location in the projected elevation map \check{M}_g will be assigned with a predefined value ν , $\nu \in \mathbb{R}^-$. In addition, consider the point \hat{p}_j in a circular region that satisfies

$$\|\hat{p}_j - \bar{p}\| < \kappa, \bar{p} \in \mathcal{I}, \quad (14)$$

it will also be assigned with the same value ν in map \check{M}_g , as shown by the orange circles in Fig. 4. The parameter κ is tuned according to the robot chassis in realistic applications. Note that there are overlaps among different circles, the elevation value of a grid under m circles is $m * \nu$. For a sample point p_r by the sampler, if $\check{h}(p_r) < \nu$, then this point will not be considered for the tree extension.

Moreover, when the hazardous regions are marked on the map, the edges and nodes of the tree \mathcal{T} in these circular regions are pruned to avoid the possible paths generating in the hazardous areas. For the point in the indicator set \mathcal{I} (e.g., ζ_s^- in Fig. 4), it searches all the nodes on the tree \mathcal{T} . If the node n_j on the tree \mathcal{T} satisfies $\|n_j - \zeta_s^-\| < \kappa$, $\zeta_s^- \in \mathcal{I}$, then the node and the associated edges are pruned from the tree \mathcal{T} .

The traversable map is generated incrementally along with the sampling process in the path planning. Conversely, the successive path planning progress further utilizes the generated traversable map for trajectory generation. Through multiple path planning processes, it will be clear to the robot that some areas in the environment are strictly impassable concerning the robot gradeability. Notably, the traversable map is reusable once it is built. Therefore, the robot can directly use the generated traversable map for the navigation task in the same environment.

C. Optimization of the Trajectory

The objective for the wheelchair path planning takes into consideration both the vehicle stability and the trajectory length. The cost function is defined as

$$f(\xi) = \beta_1 \Gamma(\xi) + \beta_2 \mathcal{L}(\xi), \quad (15)$$

where β_1 and β_2 are predefined constant values in an application. For trajectory optimization, we propose to follow closely the concept of the RRT* approach [32] as it can achieve asymptotic optimality. We have the claim that

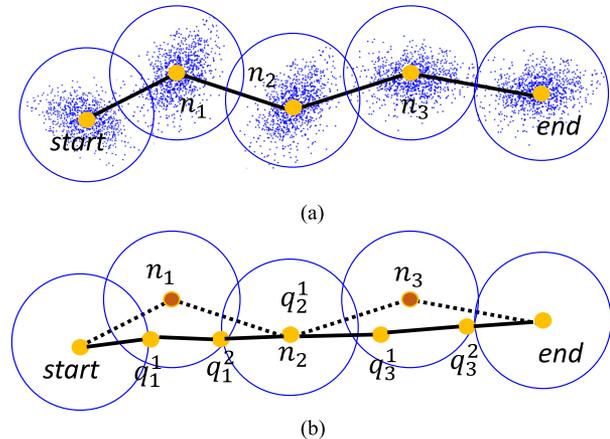


Fig. 5. Trajectory optimization. (a) Sample points generation around the nodes in the generated trajectory. The blue dots represent the new samples. (b) The solid line indicates the new trajectory generated by the rewire mechanism.

Claim 1: The cost function is admissible with the RRT* path planning framework.

Proof: Given all states $\langle s_1, s_2, \dots, s_l \rangle$ on the trajectory ξ . We assume that the calculation of the stability between different states are independent, i.e. $\Gamma(\xi) = \sum \gamma(s)$. $\Gamma(\xi) \geq 0$ is bounded. Note that $\mathcal{L}(\xi) = \sum \|s_{j+1} - s_j\|$, clearly, (15) holds the optimal substructure property. Denote ζ_1 and ζ_2 as two path segments on ξ and $\zeta_1 \zeta_2$ as their concatenation. Firstly, $f(\zeta_1 \zeta_2) = f(\zeta_1) + f(\zeta_2)$, meaning the cost function is additive. Moreover, $\Delta f(\zeta_1) \leq \Delta f(\zeta_1 \zeta_2)$, implying monotonicity. Finally, for our cost function, the concatenate segments have similar cost. The cost function must be Lipschitz continuous, i.e., there exists a K satisfying $\|f(\zeta_1) - f(\zeta_2)\| \leq K \|\zeta_1 - \zeta_2\|$. Thus, the cost function in (15) is admissible within the RRT* framework. ■

Therefore, the RRT* framework can be used for the optimization. Similar to RRT*, the embedded rewire mechanism is used to change the connections of the tree structure to achieve lower objective function value with respect to both the vehicle stability and path length.

Besides, to further improve the efficiency of the optimization process, we present an adaptive weighting sampling mechanism that biases the sampling to the area that contributes to improving the quality of the path. This scheme is enabled when a sparse nominal path $\bar{\xi}$ between the start point and the goal point is found. As the path is found on the tree \mathcal{T} , $\bar{\xi}$ is composed of a set of nodes N and segments E . The proposed sampling-based methods aim to sample around the nodes in N . The probability of getting a sample point x is

$$P(x) = \sum_{j=1}^k \omega_j \mathcal{N}_j(v_j, \sigma_j), \quad (16)$$

where \mathcal{N}_j is the normal distribution with mean v_j and covariance σ_j , ω_j is the corresponding weight in GMM. k is the number of normal distributions in GMM, which is also the number of nodes N in the generated path ξ . Eq. (16) indicates that we draw points randomly around the nodes in the generated trajectory, which are shown by the blue points in Fig. 5(a).

Algorithm 3: Path Optimization.

Input: $\tilde{\mathcal{M}}'_g$, Nominal Trajectory $\tilde{\xi}$, \mathcal{T}
Output: Optimal Feasible Path ξ^*

```

1 while  $\tilde{\xi}$  not converge do
2    $\mathbf{p}_r \leftarrow \text{Sampling}(\tilde{\mathcal{M}}'_g, \tilde{\xi}, \mathcal{W})$ ;
3    $[\mathbf{E}_r, \mathbf{p}_n] \leftarrow \text{Nearest}(\mathcal{T}, \mathbf{p}_r)$ ;
4   if  $\text{SegCheck}(\mathbf{E}_r)$  then
5      $\mathcal{T}.\text{add}(\mathbf{E}_r, \mathbf{p}_n)$ ;
6      $\mathbf{X}_n \leftarrow \text{NearestNbr}(\mathcal{T}, \mathbf{p}_n)$ ;
7      $[\mathbf{f}_\delta^r, \mathbf{q}_r, \tilde{\xi}] = \text{Rewire}(\mathcal{T}, \mathbf{p}_n, \mathbf{X}_n)$ ;
8   end
9    $\mathcal{W} \leftarrow \text{AdjustWgt}(\mathbf{f}_\delta^r, \mathbf{q}_r)$ ;
10 end

```

Instead of uniformly sampling points with the same weight of all the normal distributions in GMM [33], we propose an adaptive weighting sampling method that biases the sampling to the areas that benefit to reducing the trajectory objective value in (15). In a sampling round with the GMM model, we get a set of sample points $\mathcal{P}_k = \langle p_1, p_2, \dots, p_k \rangle$, where k indicates the number of nodes in the generated trajectory. In other words, $p_1 \sim p_k$ are the points near the nodes in the trajectory. The $\text{Rewire}(\cdot)$ function is utilized to change the connections of the nodes on the tree around these sampling points. Through the rewire mechanism, a point $p_j \in \mathcal{P}_k$ contributes to reducing the objective function value with f_δ^j . Moreover, the mechanism generates more $q_j - 1$ nodes in the trajectory, as shown in Fig. 5(b). Temporarily, the weights of the current GMM with k nodes are adjusted as

$$w_j = \frac{e^{f_\delta^j}}{\sum_{j=1}^k e^{f_\delta^j}}. \quad (17)$$

The higher the f_δ^j , the higher the corresponding weight. We use the $e^{f_\delta^j}$ to highlight the impact of a greater f_δ^j . In the next sampling round, for each node $n_j \in N$, it becomes q_j new nodes and there will be q_j Gaussian components $\mathcal{N}_j(v_j, \sigma_j)$ around the node. The corresponding weights of these components in a new GMM are $w'_j = w_j/q_j$. The means and covariances of the GMM components are kept the same in different sampling rounds. Totally, in this new GMM sampling model, there will be $k' = \sum_{j=1}^k q_j$ components. This model is then used for sampling in a new round.

By the proposed method, the weights of the GMM components are changed adaptively in each sampling round. Specially, the weight of the GMM component that significantly contributes to reducing the objective function is enlarged. This process iterates until an optimal path is found. Intuitively, the sampling points would bias to the area that greatly contributes to reducing the objective value of a path, which significantly improves the searching efficiency for an optimal path.

The proposed trajectory optimization method is detailed in Algorithm 3. As opposed to the standard RRT* method, we propose to take the vehicle stability and path cost into consideration. Moreover, we propose an adaptive weighting GMM sampling

TABLE I
VALUES OF THE PARAMETERS IN THE EXPERIMENTS

	Description	Simulation	Real-world
τ	Eq. (2)	0.8	0.8
μ_i	Eq. (9)	2.0	3.0
λ	Eq. (13)	2.0	3.0
κ	Eq. (14)	0.4	0.6
β_1	Eq. (15)	5.0	10.0
β_2	Eq. (15)	5.0	10.0

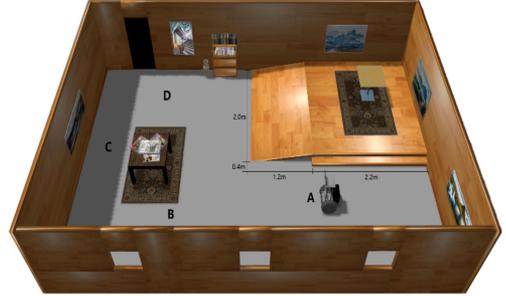


Fig. 6. Simulation environment contains a slope way, a staircase, and other obstacles.

method. As indicated in Line 2, the function $\text{Sampling}(\cdot)$ accounts for generating sample points with the GMM model. The weights are adjusted in Line 9 in Algorithm 3, which is used for the next sampling round.

VI. EXPERIMENTS AND RESULTS

The proposed method was validated in both simulation and real-world experiments by verifying the key components in the present method and comparing it with other strategies for path planning in environments with slope way.

A. Parameter Setting

The values of the parameters in the experiments are recorded in Table I. These parameters are tuned according to the analysis of robot physical properties and trial-and-error experiments. The experiments were conducted in the environments shown in Fig. 6 and Fig. 13. It is assumed that there is no other dynamic objects and the light condition is suitable so that the environment model can be built with little noise.

The parameter τ measures the gradeability of the vehicle for slope climbing. The vehicle in the real-world cannot climb the ramp with the slope higher than 0.8 rad and the simulation vehicle shares the same physical property with the real one. Therefore, τ is set to 0.8 in both simulation and real-world experiments. Similarly, the parameter κ is set according to the chassis of the simulation and real-world vehicle. It is carefully tuned to ensure the safe distance from the robot to the hazardous area in the navigation. The parameter λ is determined by trials with its values starting from 0 in simulation and real-world environments. The current setting can significantly improve the path planning efficiency. The parameters μ_i, β_1, β_2 are key components in the objective function with the weighted sum fashion. We utilize the method in [34] to tune these parameters. Notably,

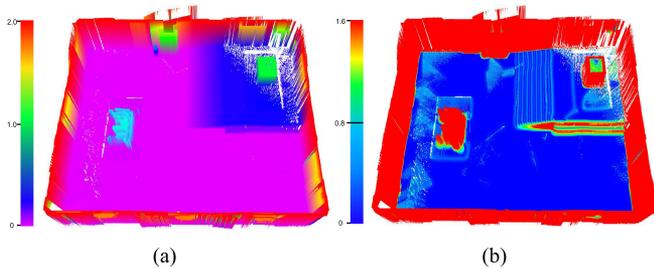


Fig. 7. Different representations of the simulation environment. (a) Elevation map. The color bar indicates the height. (b) Hazardous map. The color bar indicates the hazardous level. The area with the color lighter than 0.8 is hazardous for robot navigation.

we do not set explicit preference to the weights in the object function, thus β_1 and β_2 are set with the same value. Instead, we tune the parameter μ_i through experiments to highlight the vehicle stability. Then, these parameters in the objective function are further tuned in terms of the computational efficiency and data presentation to arrive at the current version.

B. Simulation Experiments

We built a 5 m \times 7 m \times 2 m indoor simulation environment with slope and staircase using Gazebo³ simulation engine, as shown in Fig. 6. The slope is 0.3m above the ground and the horizontal lengths for the slope and staircases are 1.2 m and 0.4 m, respectively. The width of the slope is 2.0 m and the width of the staircase is 2.2 m. A four-wheel differentially driven robot is utilized to simulate the wheelchair robot, which is 0.5 m tall with 0.2 m track. The length and width of the robot are 0.5 m and 0.4 m, respectively. In the simulation engine, the simulated robot and the environment share the same physical properties with real ones. Thus, the experimental result can reflect that in the real world. We use an ASUS laptop with Intel Core i7-6700HQ at 2 GHz, 16 GB memory for the simulation experiment. The software is developed using the C++ programming language and implemented on Ubuntu 14.04 LTS operating system.

1) *Traversable Map Generation*: The localization strategy in the experiment is a learning-based camera relocalization method [35]. For mapping, we use the grid mapping strategy in [20] to build an elevation map of the environment, which is shown in Fig. 7(a). However, this map is not directly suitable for stable wheelchair navigation in the environment with slope way. Planning with this map can lead to unsafe vehicle navigation performance attributed to the inconspicuous distinguishment between the traversable areas and the hazardous areas. With the elevation map, the state-of-the-art map generation method in [36] can produce a 3D hazardous map, as shown in Fig. 7(b), which can explicitly mark the unsafe regions and serve as a traversable map. We conducted repetitive experiments with this method and found it took over 5 s to generate this traversable map, which is time-consuming. Besides, planning with this complex 3D map is inefficient.

Our method projects the elevation map in Fig. 7(a) into a 2D elevation map, which is shown in Fig. 8(a). The generation of the 2D traversable map in a navigation task with the proposed

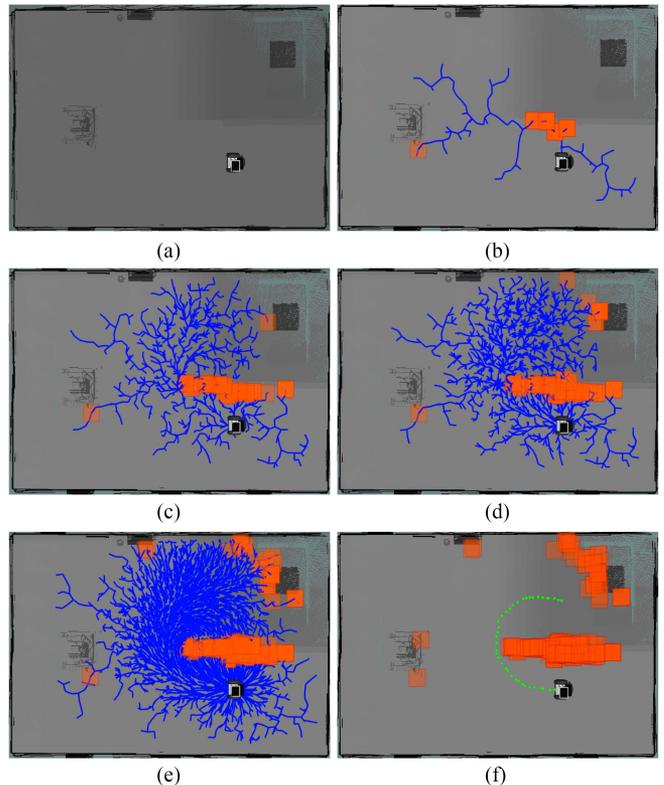


Fig. 8. Incrementally traversable map generation with the proposed path planning strategy. The orange regions partially indicate the hazardous areas. Particularly, (f) indicates the final traversable map for slope way navigation in this task.

method is shown in Fig. 8. A tree is built incrementally to find a path for the robot, as shown by the blue lines. With the tree extension process, the unsafe regions can be marked on the projected grid map. The orange regions in the traversable map indicate the hazardous areas (see Fig. 7(b)) that the robot should avoid during this navigation task. With the path planning process, the orange regions are incrementally generated. Note that the orange regions gather at the areas that are too steep for the robot to pass through, while there is no orange region at the traversable slope areas, implying that our proposed method can help the robot distinguish the slope from the hazardous areas, hence avoiding the possible unsafe path leading to vehicle crash during the navigation task. Fig. 8 shows an example of 2D traversable map generation process with the given start point and target point. The traversable map is not complete without indicating all the hazardous areas but is enough for the safe robot navigation in this task. Practically, all the hazardous areas can be marked on the map through multiple turns of path planning process with different targets. Our method needs only 0.1 s on average for the 2D traversable map generation. Planning with the developed traversable map is more efficient than using the complex 3D hazardous map.

The effectiveness of the proposed method is verified in this experiment. As shown in Fig. 8, the tree structure can extend from the start point to the goal point and build the feasible path shown in Fig. 8(f). Obviously, following the planned trajectory, the robot can reach the target above the ground through the slope way instead of the staircase. The planned trajectory can

³<http://gazebosim.org>

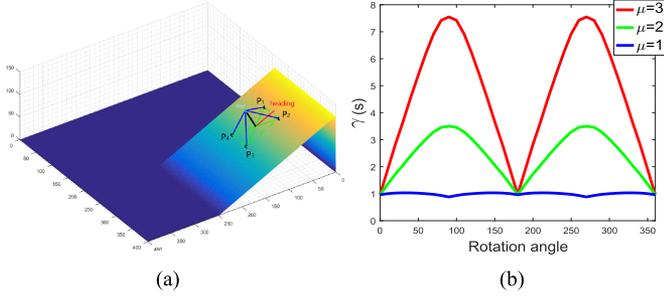


Fig. 9. Wheelchair stability on a slope. (a) Stability pyramid on the slope. (b) Vehicle stability with different rotation angles.

also improve the vehicle stability and keep the vehicle a safe distance away from the hazardous areas in the navigation.

2) *Vehicle Stability Function Evaluation*: To evaluate the effectiveness of the proposed stability function in (8), experiments were conducted in a simplified environment that is similar to the simulation environment with slope way, as shown in Fig. 9(a). The wheelchair on the slope is represented by the stability pyramid where $p_1 \sim p_4$ represent four PoCs. Initially, the robot is in the SD direction. The robot was set to rotate at a fixed point on the slope and the stability values of $\gamma(s)$ in (8) were recorded. The higher the stability value $\gamma(s)$, the more unstable the vehicle. As a key to achieve the emphasis of lateral direction, we conducted experiments with different μ in (9). Firstly, we set $\mu_i = 1$ for $i = 1, 2, 3, 4$, meaning that there is no emphasis on the lateral direction with respect to the SD. Secondly, we record the stability value for the experiment with the setting of $\mu > 1$.

Fig. 9(b) describes the stability value varying with vehicle rotations with different settings. The blue line represents the stability value of the vehicle when $\mu = 1$. The vehicle is in the most stable state on the slope when it heads towards to the slope direction ($|\angle \vec{r}, \vec{r}| \simeq 0^\circ$) or is placed laterally to the slope ($|\angle \vec{r}, \vec{r}| \simeq 90^\circ$). This is because there is no much difference between the width and length of the simulated vehicle, thus the stability is similar in these two cases. However, people may feel uncomfortable when $|\angle \vec{r}, \vec{r}| \simeq 90^\circ$. When $\mu > 1$, as shown by the green line and the red line in Fig. 9(b), the lowest stability value is achieved when the vehicle is in the slope direction, i.e., $|\angle \vec{r}, \vec{r}| \simeq 0^\circ$. The higher the angle $|\angle \vec{r}, \vec{r}|$ ($|\angle \vec{r}, \vec{r}| \in [0^\circ, 90^\circ]$), the more unstable the vehicle. It demonstrates the emphasis on the lateral direction with the proposed method. Moreover, the higher the value of μ , the more significant the emphasis. Therefore, the proposed stability measure with suitable μ can ensure the safety and human comfort for wheelchair navigation on the slope way.

3) *Evaluation of the Trajectory Optimization Method*: To verify the effectiveness of the proposed optimization method, we compared it with the path planning method adopting the proposed framework without GMM optimization. As shown in Fig. 10(a), the cyan line represents the trajectory generated by the proposed method, while the green and red lines represent the trajectories generated by the comparative methods. Notably, with the proposed method, the planner generates a trajectory that heads towards the slope direction. The stability values $\gamma(s)$ of the robot changing with the navigation time along different trajectories are recorded in red, green, and cyan lines in

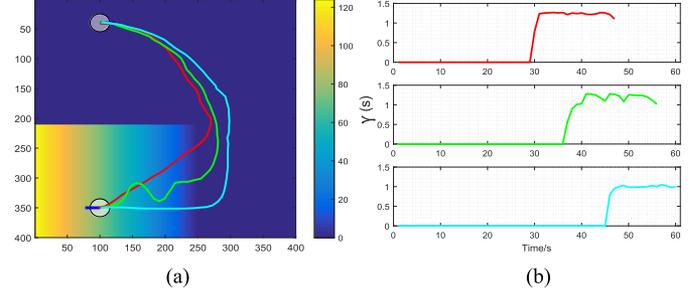


Fig. 10. Vehicle stability evaluation. (a) Slope climbing trajectories. (b) Vehicle stability on different trajectories.

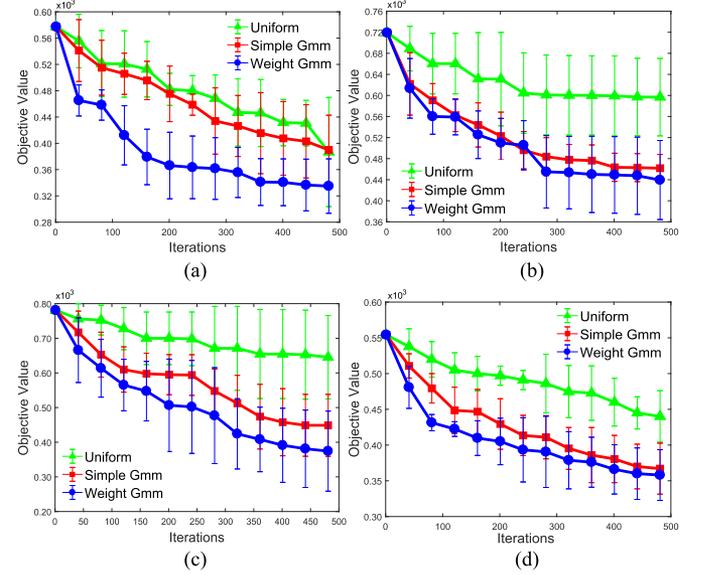


Fig. 11. Comparison of the three methods: the proposed adaptive weighting GMM sampling method, the uniform sampling method, and the simple GMM sampling method in four experiments.

Fig. 10(b). The path lengths of the three trajectories are 4.7m, 5.6m, and 6.0m, respectively. The total vehicle stability values along these trajectories calculated by (1) are 47.3, 54.6, and 32.2, respectively. This figure exhibits that the proposed method could plan a path with higher stability than the method without proposed optimization mechanism. With all the methods, the instability increases when the robot moves on the slope. It is unstable for the vehicle to climb the slope with large angles with respect to the SD. Comparatively, the proposed method drastically helps reduce vehicle instability.

We conducted experiments by comparing the adaptive weighting sampling scheme with the method that uses the uniformly sampling and the simple GMM sampling. Specially, the simple GMM sampling strategy uniformly distributes the weights for the composed normal distributions. The experiments were conducted in the simulation environment and in these experiments, the robot was assigned with a task to reach a designated target above the ground through a slope with four different start positions, A-D, shown in Fig. 6(a). We conducted 20 repetitive experiments for each experimental scenario. Meanwhile, the objective value in (15) for each generated trajectory is recorded during the optimization process, which is specified in Fig. 11. Generally, in all the four start positions, both the simple GMM

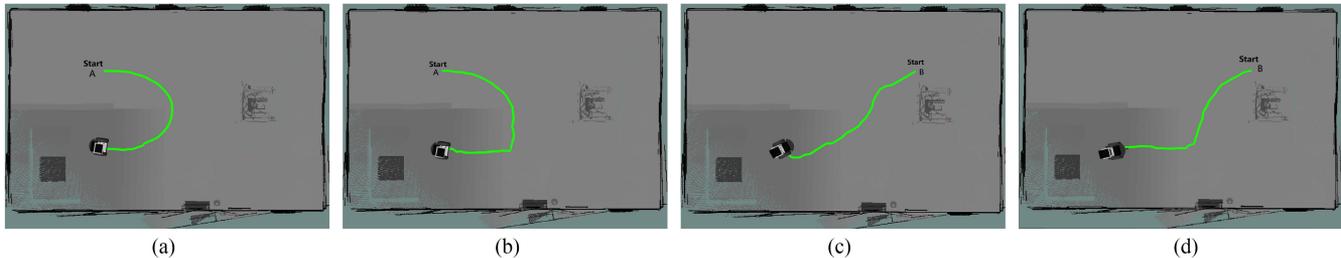


Fig. 12. Comparison between our method and the T-RRT* method. (a)–(d) record the navigation trajectories at different start points using these two methods.

TABLE II
STATISTICS OF PATH LENGTH [M] AND STABILITY VALUES WITH
DIFFERENT METHODS

	Ours		T-RRT*		Stable-RRT	
	Len.	Stab.	Len.	Stab.	Len.	Stab.
A	4.5	30.37	4.2	42.21	4.9	37.17
B	5.2	29.23	5.1	40.21	5.8	33.82

method and the adaptive weighting GMM method overwhelm the uniform sampling method in reducing the objective function with fewer iterations. Particularly, the proposed adaptive weighting sampling strategy shows better performance than the simple GMM method, as shown by the blue lines and the red lines in Fig. 11. It can achieve higher efficiency in reducing the objective value even the robustness of these two methods is not distinguishable. The experimental results demonstrate that by focusing on sampling in the area that can rapidly reduce the objective value, the optimization efficiency can be improved.

4) *Comparison With Other Path Planning Methods:* This study proposes a path planning approach towards safe and human-comfortable navigation for the wheelchair robot. To date, few comparative methods devote to reducing the human discomfort related to the vehicle stability. The proposed method is compared with the slope climbing RRT* (T-RRT*) method [27] that is able to generate a shorter path from the start point to the goal in uneven terrain environment. Another method for comparison is the Stable-RRT method proposed in [28] devoted to planning stable paths for the vehicle. The experiments were conducted in location *A* and *B* in the simulation environment. The path length (Len.) of the trajectory, together with the stability value (Stab.) in (1), are recorded in Table II. Obviously, the robot shows higher vehicle stability with lower stability value using the proposed method during the navigation compared with the T-RRT* and Stable-RRT method in these two cases. The Stable-RRT method achieves the task with relatively higher vehicle stability while its path length is longer than the T-RRT* method.

In particular, the performance of our method and the T-RRT* method is depicted in Fig. 12. The green lines show trajectories of the two methods. The T-RRT* method plans the shortest trajectory to the target, which is similar to the performance of path planning on even terrain. Comparatively, the proposed method can plan a path that firstly biases towards the slope direction and then leads the robot to climb it, which avoids lateral



Fig. 13. Real-world environment settings. The green dots indicate the start points and the orange dots indicate the target points. (a) Environment 1. (b) Environment 2.

movement with respect to the slope direction. This can improve the vehicle stability and human comfort, as indicated in (8). Thus the trajectories by our method are more safe and stable for robot navigation.

C. Real-World Experiments

We implemented the developed method on a real wheelchair robot platform. We built a robot prototype with RGB-D and Lidar sensors for perceiving the environment and localization. The wheelchair robot is developed with 150 kg payload and the developed method is implemented on a laptop equipped on the vehicle. The height, length, and width of the wheelchair are 0.94m, 1.17m, and 0.62m, respectively. We conducted experiments in two realistic environments with slopes, as shown in Fig. 13. The robot is tasked for reaching the targets above the ground. In practice, the robot turned itself towards the slope direction before climbing it by using the proposed method, as shown in Fig. 14. Then, the robot can successfully reach the targets through the slope ways.

In the real-world experiments, the slope is wide. For safe wheelchair navigation and coping with possible turns of the wheelchair on the slope, there is a stipulation [37] by Americans with Disabilities Act (ADA) that the clear space for wheelchair performing 180° turn is a circular region with radius of 0.76 m. The minimum horizontal distance from the vehicle center to the hazardous region (e.g., slope edge) is 0.76 m, which is a safe distance for free robot turning. The proposed traversable map scheme is effective in keeping the safe distance to the hazardous slope edge during the navigation, as shown in Fig. 14. We conducted 20 experiments in these environments at two different start locations and recorded the success rate of the path planning in Table III. The success here means that the robot

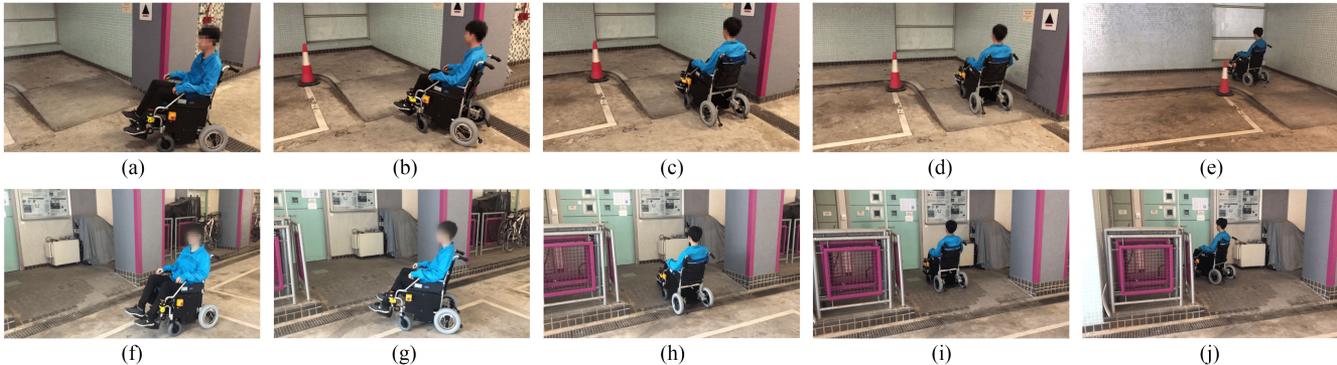


Fig. 14. Real-world experiments for wheelchair navigation on a slope. The target for the wheelchair robot is above the ground that can be reached through a slope while keeping a reasonable distance from the hazardous areas. (a)–(e) Environment 1. (f)–(j) Environment 2.

TABLE III
SUCCESS RATE OF THE PROPOSED PLANNING APPROACH

	Env. 1(a)	Env. 1(b)	Env. 2(a)	Env. 2(b)
Success Rate	0.95	1.00	0.90	0.95



Fig. 15. Real-world experiments with two comparative methods. The colorful stripe indicates the distance to the hazardous region. (a) Ours. (b) T-RRT*.

can successfully navigate from its current location to the target above the ground through the slope. As the table shows, the success rate of the proposed method is higher than 90%, which exhibits the robustness of the proposed method in real-world applications. The failure cases are mainly caused by the quality of the sensor data in the real-world application, which may make the localization unreliable and deteriorate the navigation performance [38].

We compared our method with the T-RRT* method in Environment 1. There is no human on the wheelchair in these experiments for safety purpose. Intuitively, as shown in Fig. 15, the trajectory planned by our method is more safe for the robot to follow compared with the T-RRT* method. The colorful stripe measures the distance from the robot left side to the edge of the terrain above the ground. The lighter the color, the higher the risk. With the T-RRT* method, the robot could move to the hazardous area in close proximity to the terrain edge. Comparatively, the proposed method ensures the safety and human comfort for the wheelchair navigation. Details can be found in the video demonstration.⁴

⁴Video demonstration: <https://youtu.be/ywjQ8HBC79U>

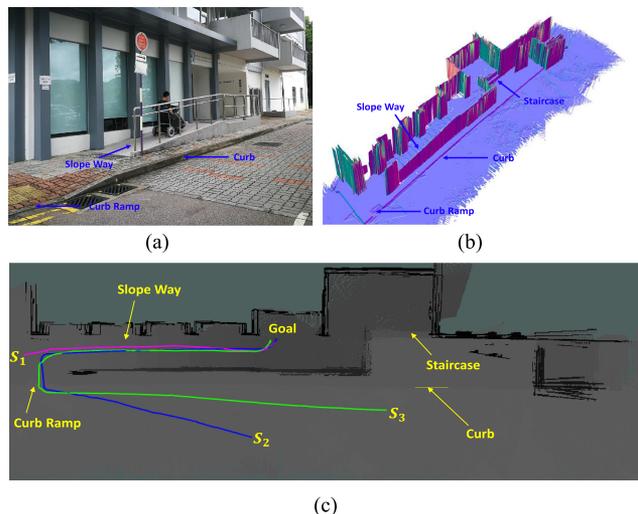


Fig. 16. Real-world experiment at a building entrance. (a) Real-world environment. (b) Built environment model indicating environment configurations. (c) Wheelchair trajectories of three experiments.

The experiments were also conducted at a building entrance with slope way and staircase. Fig. 16(a) shows the environment and Fig. 16(b) intuitively indicates the locations of different accesses. We conducted experiments at three different start locations. The navigation trajectories at three start locations are shown in Fig. 16(c). The robot can successfully reach the target through the slope instead of the staircase by using the developed method. The curb is too steep for the wheelchair to climb. The robot could avoid the hazardous areas near the curb and the slope edge in the navigation.

D. Discussions

The evaluation environments contain slopes and staircases in our study. The proposed method can be extended for motion planning in more complex scenarios due to the nature of the stability analysis method, which provides options for safe and human-comfortable navigation in more general environments with different terrains.

Note that the slope in our experiment setting is wide. However, it is also common for wheelchair navigation in narrow slopes. While in this case, the performance of the T-RRT* method and our method is approximately the same. It is because that the planning space is constrained and there is no much space for the trajectory to bias to other directions on the slope. Comparatively, our proposed features that it is effective on different slopes.

Currently, the proposed method considers the human-comfort that results from the vehicle stability without considering the human comfort involves social factors. However, the socially-aware navigation scheme can be easily integrated into this framework as the generality of the sampling-based path planning methods. Besides, the application of the proposed algorithm is not limited to the wheelchair robot. The proposed navigation framework can be used in other passenger vehicles.

VII. CONCLUSION AND FUTURE WORK

In this study, a path planning method for the robotic wheelchair navigation in the environment with slope way is proposed, with which the robot can climb the slope above the ground steadily. The effectiveness of each separated module and the whole framework have been demonstrated by both simulation and real-world experiments. With the suggested objective function that takes into consideration both the vehicle stability and the path length, the planner could generate a feasible path that utmost ensures the vehicle safety and the human comfort. A sampling strategy ensuring the fast convergence of the proposed method makes the path planning process more efficient. Moreover, the proposed traversable map generation strategy by using the sampling-based path planning scheme can incrementally and successfully generate a traversable map. With this map, the path planner could plan the safe path that keeps the robot a reasonable distance from hazardous areas.

In the future, we will consider implementing the proposed method in more challenging environment scenarios, for example, in densely populated dynamic environments. We will also develop control techniques for more smooth vehicle slope climbing.

REFERENCES

- [1] T. Iimura and K. Yamamoto, "2A1-H03 development of single-passenger mobility-support robot 'ROPITS': Verification of self-localization function with camera images," *Robot., Mechatron. Lecture Presentat. Summ.*, vol. 2014, pp. 2A1-H03, 2014.
- [2] H. Wang, Y. Sun, and M. Liu, "Self-supervised drivable area and road anomaly segmentation using RGB-D data for robotic wheelchairs," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4386–4393, Oct. 2019.
- [3] G. Feng, L. Buşoniu, T.-M. Guerra, and S. Mohammad, "Data-efficient reinforcement learning for energy optimization of power-assisted wheelchairs," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9734–9744, Dec. 2019.
- [4] G.-P. Gwon, W.-S. Hur, S.-W. Kim, and S.-W. Seo, "Generation of a precise and efficient lane-level road map for intelligent vehicle systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 4517–4533, Jun. 2016.
- [5] S. Pütz, T. Wiemann, J. Sprickerhof, and J. Hertzberg, "3D navigation mesh generation for path planning in uneven terrain," *IFAC-PapersOnLine*, vol. 49, no. 15, pp. 212–217, 2016.
- [6] J.-W. Choi and K. Huhtala, "Constrained global path optimization for articulated steering vehicles," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 1868–1879, Apr. 2015.
- [7] G.-X. Liu, L.-F. Shi, S. Chen, and Z.-G. Wu, "Focusing matching localization method based on indoor magnetic map," *IEEE Sensors J.*, vol. 20, no. 17, pp. 10012–10020, Sep. 2020.
- [8] C. Guo, D. Li, G. Zhang, and M. Zhai, "Real-time path planning in urban area via vanet-assisted traffic information sharing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5635–5649, Jul. 2018.
- [9] X. Zhang, J. Wang, Y. Fang, and J. Yuan, "Multilevel humanlike motion planning for mobile robots in complex indoor environments," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1244–1258, Jul. 2018.
- [10] C. Wang *et al.*, "Autonomous mobile robot navigation in uneven and unstructured indoor environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 109–116.
- [11] Y. Morales *et al.*, "Including human factors for planning comfortable paths," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 6153–6159.
- [12] D. Gingras, T. Lamarche, J.-L. Bedwani, and É. Dupuis, "Rough terrain reconstruction for rover motion planning," in *Proc. IEEE Can. Conf. Comput. Robot Vision*, 2010, pp. 191–198.
- [13] M. Häselich, M. Arends, N. Wojke, F. Neuhaus, and D. Paulus, "Probabilistic terrain classification in unstructured environments," *Robot. Auton. Syst.*, vol. 61, no. 10, pp. 1051–1059, 2013.
- [14] À. Santamaria-Navarro, E. H. Teniente, M. Morta, and J. Andrade-Cetto, "Terrain classification in complex three-dimensional outdoor environments," *J. Field Robot.*, vol. 32, pp. 42–60, 2015.
- [15] Y. Toda and N. Kubota, "Self-localization based on multiresolution map for remote control of multiple mobile robots," *IEEE Trans. Ind. Informat.*, vol. 9, no. 3, pp. 1772–1781, Aug. 2013.
- [16] H. Carrillo, P. Dames, V. Kumar, and J. A. Castellanos, "Autonomous robotic exploration using occupancy grid maps and graph SLAM based on Shannon and Rényi Entropy," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2015, pp. 487–494.
- [17] Y. Wang and W. Chen, "Hybrid map-based navigation for intelligent wheelchair," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 637–642.
- [18] J. Gu and Q. Cao, "Path planning for mobile robot in a 2.5-dimensional grid-based map," *Ind. Robot: An Int. J.*, vol. 38, no. 3, pp. 315–321, 2011.
- [19] J. Sock, J. Kim, J. Min, and K. Kwak, "Probabilistic traversability map generation using 3D-LIDAR and camera," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 5631–5637.
- [20] A. Souza and L. M. Gonçalves, "Occupancy-elevation grid: An alternative approach for robotic mapping and navigation," *Robotica*, vol. 34, no. 11, pp. 2592–2609, 2016.
- [21] K. Iagnemma, S. Shimoda, and Z. Shiller, "Near-optimal navigation of high speed mobile robots on uneven terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 4098–4103.
- [22] A. K. Singh, K. M. Krishna, and S. Saripalli, "Planning trajectories on uneven terrain using optimization and non-linear time scaling techniques," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 3538–3545.
- [23] J. Guzzi and G. A. Di Caro, "From indoor GIS maps to path planning for autonomous wheelchairs," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4773–4779.
- [24] E. Papadopoulos and D. A. Rey, "A new measure of tipover stability margin for mobile manipulators," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1996, pp. 3111–3116.
- [25] S. Yu, T. Wang, X. Li, C. Yao, Z. Wang, and D. Zhi, "Configuration and tip-over stability analysis for stair-climbing of a new-style wheelchair robot," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, 2010, pp. 1387–1392.
- [26] C. Wang and M. Q.-H. Meng, "Variant step size RRT: An efficient path planner for UAV in complex environments," in *Proc. IEEE Int. Conf. Real-Time Comput. Robot.*, 2016, pp. 555–560.
- [27] D. Devaurs, T. Siméon, and J. Cortés, "Optimal path planning in complex cost spaces with sampling-based algorithms," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 415–424, Apr. 2015.
- [28] M. Norouzi, F. De Bruijn, and J. V. Miró, "Planning stable paths for urban search and rescue robots," in *Robot Soccer World Cup*. Berlin, Germany: Springer, 2011, pp. 90–101.
- [29] K. D. Iagnemma, A. Rzepniewski, S. Dubowsky, P. Pirjanian, T. L. Huntsberger, and P. S. Schenker, "Mobile robot kinematic reconfigurability for rough terrain," in *Sens. Fusion and Decent. Cont. in Robot. Syst. III*, The International Society for Optics and Photonics (SPIE): Boston, MA, vol. 4196, 2000, pp. 413–421.
- [30] L. Moody, A. Woodcock, M. Heelis, C. Chichi, S. Fielden, and D. Stefanov, "Improving wheelchair prescription: An analysis of user needs and existing tools," *J. Prevention, Assessment Rehabil.*, vol. 41, no. Supplement 1, pp. 1980–1984, 2012.
- [31] R. L. Kirby *et al.*, "Wheelchair stability: Effect of body position," *J. Rehabil. Res. Develop.*, vol. 32, no. 4, pp. 367–72, 1995.

- [32] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.
- [33] P. Lehner and A. Albu-Schäffer, "Repetition sampling for efficiently planning similar constrained manipulation tasks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 2851–2856.
- [34] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization: New insights," *Structural Multidisciplin. Optim.*, vol. 41, no. 6, pp. 853–862, 2010.
- [35] C. Wang, J. Wang, C. Li, T. Yan, L. Meng, and M. Q.-H. Meng, "Safe and robust mobile robot navigation in uneven indoor environments," *Sensors*, vol. 19, no. 13, pp. 2993–3012, 2019.
- [36] P. Fankhauser and M. Hutter, "A universal grid map library: Implementation and use case for rough terrain navigation," in *Robot Operating System*. Berlin, Germany: Springer, 2016, pp. 99–120.
- [37] U.S. Architectural and Transportation Barriers Compliance Board (Access Board), "Americans with disabilities act (ADA) accessibility guidelines for buildings and facilities," *Federal Register*, Washington, DC, 1991, pp. 56–173.
- [38] G.-X. Liu, L.-F. Shi, and D.-J. Xin, "Data integrity monitoring method of digital sensors for Internet of Things applications," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4575–4584, May 2020.