International Journal of Automotive Technology, Vol. 24, No. 2, pp. 459–468 (2023)
DOI 10.1007/s12239–023–0038–1

pISSN 1229–9138/ eISSN 1976–3832

IMPROVED HYBRID A-STAR ALGORITHM FOR PATH PLANNING IN AUTONOMOUS PARKING SYSTEM BASED ON MULTI-STAGE DYNAMIC OPTIMIZATION

Tianchuang Meng[1)], Tianhong Yang[2,5)], Jin Huang[1)*], Wenrui Jin[2,5)], Wei Zhang[1)], Yifan Jia[1)], Keqian Wan[3)], Gang Xiao[3)], Diange Yang[1)] and Zhihua Zhong[1,4)]

[1)]School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China
[2)]School of Mechanical Engineering, Tongji University, Shanghai 201804, China
[3)]Jiangxi Kmax Industrial Co., Ltd., No.1589 Zishi Road, Xinjian District, Nanchang 330100, China
[4)]Chinese Academy of Engineering, No.2 Bingjiaokou Bystreet, Xicheng District, Beijing 100088, China
[5)]The State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha, Hunan 410082, China

(Received 3 January 2022; Revised 24 April 2022; Accepted 4 August 2022)

ABSTRACT–The recent proliferation of intelligent technologies has promoted autonomous driving. The autonomous parking system has become a popular feature in autonomous driving. Hybrid A-star algorithm is a commonly used path planning algorithm for its simplicity to deploy and the good characteristics of the generated paths in the practical engineering. To further enhance the path safety and efficiency of path planning in the autonomous parking system, this paper proposes an improved hybrid A-star algorithm through the safety-enhanced design and the efficiency-enhanced design. The safety-enhanced design integrates the Voronoi field potential into the path searching stage to take more account of path safety. The efficiency-enhanced design proposes a multi-stage dynamic optimization strategy which divides the path planning into multiple stages and performs dynamic optimization in each stage. Through simulation experiments, it is verified that the proposed improved algorithm not only generates a much safer path which stays farther from the obstacles but also significantly improves the searching efficiency in terms of time and space, merely at a finite cost of pre-processing work which can also be repeatedly utilized. We hope this paper will promote relative research on path planning in autonomous parking and serve as a reference for the practical engineering.

KEY WORDS : Autonomous driving, Autonomous parking system, Path planning, Hybrid A-star algorithm, Multi-stage dynamic optimization, Voronoi field

NOMENCLATURE

$A(s)$	: neighboring interest area
d_0	: distance to the nearest obstacle
d_V	: distance to the nearest edge of the generalized voronoi diagram
$f(x, y)$	: total cost function at the point (x, y)
$g(x, y)$	: motion cost function at the point (x, y)
H	: height of the neighboring interest area
$h(x, y)$	: heuristic cost function at the point (x, y)
r	: resolution parameter
S	: state set
θ	: heading angle
$s(x, y, \theta)$	: state variable at the point (x, y) and with the heading angle of θ
u	: weight constants in the dynamical optimization scheme
$v(x, y)$	: safety cost function at the point (x, y)
W	: width of the neighboring interest area
w	: weight constants of the cost functions in the improved algorithm
α	: constant that controls the falloff rate
$\rho_V(x, y)$	: voronoi field potential
∇	: gradient
$\Omega(s)$	: admissible set in the strategy

SUBSCRIPTS

i	: index
mid	: middle point
max	: maximum value
p	: point on RS path
goal	: target state
popped state	: popped state during the searching

**Corresponding author*. e-mail: huangjin@tsinghua.edu.cn

1. INTRODUCTION

Parking is a world-wide problem which may easily cause problems such as collisions and jams, especially in large cities. Narrow parking spots, blinding spots of drivers and sometimes their nervous emotions together increase the difficulty of the parking maneuver. Traditionally there are some approaches like collision warning or reverse imaging to assist parking but the effect is limited. With the rapid advances of autonomous driving techniques in recent years, autonomous parking system has been proposed as a promising solution to alleviate these problems (Ayawli *et al.*, 2018). It aims to lower the parking difficulty, decrease the risks of human errors and enhance the security and efficiency.

Currently the widely-equipped autonomous parking assistant system is at the SAE Level 2 of Driving Automation, which takes over both lateral and longitudinal control of the vehicle (Zhang *et al.*, 2020). It can autonomously search the available parking spots nearby and finish the parking maneuver without human drivers' intervention. Planning a safe trajectory towards the target parking spot is an essential step in the autonomous parking system.

According to González *et al.* (2016), motion planning technologies applied in automated driving scenarios can be categorized into four groups, i.e., graph search-based planners, sampling-based planners, interpolating curve planners and numerical optimization approaches. Dijkstra algorithms (Kala and Warwick, 2013; Anderson *et al.*, 2012), A-star algorithm family (Dolgov *et al.*, 2010; Ferguson *et al.*, 2008) and state lattices (Furgale *et al.*, 2013) are typical methods in the graph search-based planners. Rapidly exploring random trees (RRT) algorithm (LaValle and Kuffner Jr, 2001) is one typical sampling-based planner. For interpolating curve planners, there are Dubins curve (Dubins, 1957), Reeds-Shepp (RS) curve (Reeds and Shepp, 1990; Fraichard and Scheuer, 2004) and other curves (Kim *et al.*, 2014; Vorobieva *et al.*, 2014; Li *et al.*, 2021). Numerical optimization approaches (Gu *et al.*, 2012) typically generate trajectories by optimizing speed, steering speed, lateral accelerations, *etc*.

With the rapid development of artificial intelligence (AI) technologies, more planning algorithms combined with AI have emerged. Kamoshida *et al.* (2017) presented a route planning method for an automated guided vehicle (AGV) picking system using deep reinforcement learning. Yu *et al.* (2018) proposed an intelligent vehicle model transfer trajectory planning method based on deep reinforcement learning, which is capable of obtaining an effective control action sequence directly. Ayalew *et al.* (2021) proposed a spatial-temporal attentive motion planning network (STAMPNet) to efficiently generate feasible trajectories. Zhou *et al.* (2020) proposed value-estimation-guild (VEG) trajectory planner, which is adaptive to the uncertain interactions. The method uses the deep reinforcement learning to deal with the high uncertainty. Chaulwar *et al.* (2021) proposed a methodology for replacing the computationally intensive modules of conventional trajectory planning algorithms by using different efficient machine learning and analytical methods. Li *et al.* (2020) proposed an onboard deep reinforcement learning algorithm to optimize the real-time trajectory planning of the unmanned aerial vehicle given outdated knowledge on the network states. Naveed *et al.* (2021) proposed a robust-hierarchical reinforcement learning (HRL) framework for learning autonomous driving policies to address these problems and to ensure a robust framework.

Among the above trajectory planning algorithms, the hybrid A-star algorithm (Dolgov *et al.*, 2010) is a representative method and has been widely applied in the practical engineering since it was introduced for its simplicity to deploy and the good characteristics of the generated paths. That is, the planning process takes the vehicle kinetics into consideration by introducing the Reeds-Shepp curves and thus the generated paths satisfy the vehicle kinetics. This largely increases the availability of the generated paths. However, it is found in the practical engineering that the hybrid A-star algorithm can be further improved in terms of the collision-avoidance safety and the searching efficiency. The generated paths can be very near to the obstacles sometimes. What's more, the searching efficiency will significantly decay when surrounded by a large number of obstacles, for example the U-turn obstacles in the parking scenarios. To address these problems and enhance the safety and searching efficiency of the traditional hybrid A-star algorithm, we focus on the path planning problem in autonomous parking system under unstructured road conditions. An improved algorithm based on the conventional hybrid A-star algorithm is proposed through the safety-enhanced design and the efficiency-enhanced design. The safety-enhanced design integrates the Voronoi field potential into the path searching stage to take more account of path safety. The efficiency-enhanced design proposes a multi-stage dynamic optimization strategy which divides the path planning into multi stages and performs dynamic optimization in each stage. Through simulation experiments, it is verified that the proposed improved algorithm not only generates a much safer path which stays farther from the obstacles but also significantly improve the searching efficiency in terms of time and space, merely at a finite cost of pre-processing work which can also be repeatedly utilized.

The organization of this paper is arranged as follows. Section 2 introduces the conventional A-star algorithm, hybrid A-star algorithm and some existing research based on them. Section 3 illustrates the methodology of the proposed improved hybrid A-star algorithm, including the safety-enhanced design and the efficiency-enhanced design. Section 4 conducts a simulation to verify the proposed method and compares it to the conventional hybrid A-star algorithm. Finally, Section 5 concludes this paper.

2. HYBRID A-STAR ALGORITHM

The A-star algorithm proposed by Hart *et al.* (1968) is one of the most widely adopted path planning algorithms (Chen *et al.*, 2019). It is an extension of the Dijkstra algorithm (Dijkstra, 1959) by embedding the heuristics, reducing the blindness and improving the search efficiency. However, it works only for holonomic robots.

Considering the kinematic constraints of vehicles, hybrid A-star searching algorithm was developed, which applies heuristic components to consider nonholonomic constraints. It debuted in 2007 DARPA Urban Challenge and helped Junior (Stanford automated vehicle) with good performances in complex path planning tasks such as navigating parking lots and executing U-turns on blocked roads (Dolgov *et al.*, 2010).

Hybrid A-star algorithm succeeds the framework of traditional A-star algorithm presented in Algorithm 1. The search space (x, y, θ) in the hybrid A-star algorithm is discretized as in conventional A-star algorithm, here x, y and θ denote the coordinates of the position and the heading angle of the vehicle. Different from the traditional A-star algorithm, which ignores the kinematic constraints and orientation of the vehicle and only allows to visit the centers or corners of the grids by a linear maneuver, the hybrid A-star algorithm takes the kinematic constraints and the vehicle orientation into account, and therefore can expand successor nodes in the way that a nonholonomic vehicle is able to follow. Accordingly, it associates with each grid a continuous 3-dimensional state of the vehicle, which means it allows visiting any position of a grid instead of the center only. Hybrid A-star algorithm uses similar cost functions as the conventional A-star algorithm, assigning a cost for each surrounding node to define their weights. The cost of a node is $f(n) = g(n) + h(n)$, in which $g(n)$ represents the cost-so-far at the node n and the heuristic function $h(n)$ represents the cost-to-go at the node n. Both of these costs are much more complex to calculate than those of common A-star algorithm. Through searching in the discretized space and using the improved cost function, the hybrid A-star algorithm helps a planner to find a path connecting the given starting point and the target point with the vehicle kinematics being considered (Hosseininejad *et al.*, 2019).

Algorithm 1. Conventional Hybrid A-star Algorithm.

```
function Conventional_Hybrid_A_Star(n_s, n_g)
  // n_s=start node, n_g=target node
  Olist = ∅     // Open List
  Clist = ∅     // Close List
  Olist ← n_s
  while Olist ≠ ∅
      n ⟵ the node with the minimum f in Olist
      remove n from Olist to Clist
      find an RS curve from n to n_g
      if the RS curve is collision-free
        break
      end if
      expand n
      for all expended nodes n_i
          if not collision-free from n to n_i
              continue
          else if n_i ∈ Clist
              continue
          else if n_i ∈ Olist
              update n_i if it has lower f value
          else
              Olist ← n_i
        end if
      end for
    end while
    get the final path from n_s to n_g
    return path
```

One main advantage of hybrid A-star algorithm is that maneuvers in tight spaces and reversing are allowed. The generated path is guaranteed to be drivable though it may not the most optimal. Since it was developed, it has been widely used in autonomous robots or vehicles and many furthermore improvements are proposed. Petereit *et al.* (2012) introduced a multi-task structure to ensure the effectiveness of node search. Hemmat *et al.* (2017) combined a hybrid A-star path planner with potential fields to describe a cost-effective path planning method and the proposed real-time planner is capable of finding the optimal, collision-free path in an unstructured environment. Zhang *et al.* (2019) proposed a novel motion planning method based on hybrid A-star for real-time and curvature-contentious path planning with local post smoothing in complex dynamic environments. Shamsudin *et al.* (2017) proposed a two-stage hybrid A-star path planning to overcome the large computation memory and the long execution time in large environments. Tu *et al.* (2019) improved the Hybrid A-star algorithm with a better heuristic function and a better search policy to realize a less-time consuming graph search. In autonomous parking, Sedighi *et al.* (2019) fused the Hybrid A-star with the Visibility Diagram planning to find the shortest possible path in a hybrid environment for rapid autonomous parking. Esposto *et al.* (2017) took ride comfort of the passenger into account by reducing turning and reversing. Sheng *et al.* (2021) introduced a multistage hybrid A-star algorithm to handle narrow passages with obstacles in the cluttered environment. This paper improves the collision-avoidance safety and the searching efficiency of the traditional hybrid A-star algorithm from a different perspective.

3. METHODOLOGY

3.1. Safety-enhanced Design

The conventional hybrid A-star algorithm focuses more on the smoothing of the trajectory and can produce smooth paths that generally satisfy the non-holonomic constraints of the vehicle (Ravankar *et al.*, 2018). However, the safety of the generated trajectory by the conventional hybrid A-star algorithm can be further enhanced. For example, although the generated trajectory can avoid colliding with obstacles, it is sometimes a little close to the obstacles that the risks of collision accidents may largely increase if the subsequent control module has no sufficiently high accuracy to follow the planned trajectory.

To solve this problem, we improve the cost function in traditional hybrid A-star algorithm to enhance the safety of the trajectory. An additional safety term is added to the cost function. To construct this safety term, we adopt Voronoi field to consider the tradeoff between path length and the proximity to the obstacles.

The Voronoi field $\rho_V(x,y)$ is defined as follows.

When $d_O \geq d_O^{\max}$, $\rho_V(x,y) = 0$.

When $d_O < d_O^{\max}$, we have

$$\rho_V(x,y) = \left(\frac{\alpha}{\alpha + d_O(x,y)}\right)\left(\frac{d_V(x,y)}{d_O(x,y) + d_V(x,y)}\right) \left(\frac{d_O - d_O^{\max}}{d_O^{\max}}\right)^2 \tag{1}$$

where x and y are the coordinates of a location, d_O is the distance to the nearest obstacle, d_V is the distance to the nearest edge of the Generalized Voronoi Diagram (GVD), $\alpha > 0$ and $d_O^{\max} > 0$ are positive constants to control the falloff rate and the maximum effective range of the field.

Voronoi field has some beneficial properties. First, it is 0 when $d_O \geq d_O^{\max}$. Second, $\rho_V(x,y) \in [0,1]$ and is continuous on (x,y). Third, the potential value is 1, i.e. the maximum, at the locations of obstacles. Fourth, the potential value is the minimum on the edges of GVD. Compared with conventional potential fields, Voronoi field is scaled in proportion to the total available clearance for navigation so that narrow openings are more easily navigable.

The cost function is designed as follows by considering Voronoi field as the safety term.

$$f(x,y) = w_1 g(x,y) + w_2 h(x,y) + w_3 v(x,y) \tag{2}$$

where w_1, w_2, w_3 are weight constants, $g(x,y)$ is the motion cost from the starting node to current node, $h(x,y)$ is the heuristic cost which measures the cost from current node to the target node with obstacles being considered, $v(x,y)$ is the safety cost measured by the maximal Voronoi field potential at four corners of the bounding box of the vehicle, and $f(x,y)$ is the total cost.

Specifically, the costs are defined as follows.

The motion cost $g(x,y)$ contains the following five parts.

$$g(x,y) = PathLengthCost + ReverseCost + SwitchDirCost + SteerCost + SteerChangeCost \tag{3}$$

where $PathLengthCost$ measures the cost of the path length from the starting point to current node, $ReverseCost$ measures the motion cost of vehicle's reverse driving, $SwitchDirCost$ measures the motion cost of switching driving directions from forward to reverse or vice versa, $SteerCost$ measures the motion cost of holding a steering angle, $SteerChangeCost$ measures the motion cost of changing steering angle. On the whole, the motion cost term $g(x,y)$ is intended to lower the cost of vehicle motion, including shorter path length, less reverse driving, less switching driving directions, less steering and less steering angle change.

The heuristic cost $h(x,y)$ is defined as follows.

$$h(x,y) = \max\begin{Bmatrix} NonHolonomicWithoutObst, \\ ShortestDistanceToTarget \end{Bmatrix} \tag{4}$$

where $NonHolonomicWithoutObst$ denotes the heuristic cost by ignoring obstacles but taking into account the non-holonomic nature of the vehicle, and $ShortestDistanceToTarget$ is a dual of the first and denotes the other heuristic cost by ignoring the non-holonomic nature of the vehicle but using the obstacles to compute the shortest distance to the goal by performing A-star or Dijkstra algorithm. The detailed descriptions can be found in this reference (Dolgov *et al.*, 2010).

The safety cost $v(x,y)$ is defined as follows.

$$v(x,y) = \max\{\rho_{V1}(x,y), \rho_{V2}(x,y), \rho_{V3}(x,y), \rho_{V4}(x,y)\} \tag{5}$$

where $\rho_{Vi}(x,y), i = 1,2,3,4$ is the Voronoi field potential at the ith corner of the bounding box of the vehicle when arriving at the point (x,y). The safety cost $v(x,y)$ takes the value of the maximum of the four corners' potential values.

To conclude, this safety-enhanced design focuses on improving the cost function of the conventional hybrid A-star algorithm by supplementing a safety term $v(x,y)$ into the total cost function. This design fully utilizes the advantages of Voronoi field. Since trajectory planning in the parking lot faces the difficulty of existing many obstacles (such as many border lines around the parking spots) and narrow openings (such as the entrance of the parking spot), this design improves the safety of the generated trajectory without increasing the navigating difficulty at those narrow openings which is not always the case for other standard potential fields.

3.2. Efficiency-enhanced Design

One main challenge for parking path planning especially under unstructured road conditions is in that there are many border lines considered as obstacles around the parking spot, which should be avoided intersecting when the vehicle drives along the generated trajectory. To achieve this, conventional hybrid A-star algorithm usually takes a long time and a large searching space near the parking spots.

With the aim of enhancing searching efficiency in terms of searching time and space, we design a dynamically optimized multi-stage planning framework based on conventional hybrid A-star algorithm and the above safety-enhanced improvement.

In our strategy, parking path planning is divided into two stages. The first stage plans the path from the entrance of the parking lot to the middle point nearing the target parking spot. The second stage plans a more complicated path from the middle point to the target parking spot with avoiding intersecting surrounding border lines and satisfying the non-holonomic constraints of the vehicle. The first planning stage is easier than the second because the vehicle mainly drives along the road without many obstacles. The emphasis is laid on increasing the searching efficiency of the second planning stage.

To divide the whole path planning into two stages, we construct an available list which consists of all available middle points around the target parking spot. Here, we define a point is available when there exists at least one collision-free RS path from this point to the target parking spot. After this available set is constructed, the middle point is dynamically chosen through a specially designed optimization framework. That is to say, when the first planning stage is in progress, a middle point is simultaneously dynamically chosen according to the designed optimization scheme. As long as the first planning stage accomplishes the path searching from current point to the middle point, all the searching work is done because the second planning stage assures an RS path from the middle point to the target parking spot. Therefore, the searching efficiency both in time and space is decided totally by the first planning stage and thus largely enhanced at the cost of some extra pre-processing work. However, considering that the environment including road conditions and parking spot layout is static, the pre-processing work can be repeatedly utilized and does not need to be carried out before every parking path planning mission.

The proposed method first determines a neighboring interest area $A(s_{\text{goal}})$ around the target state s_{goal} in the target spot. It is a rectangle with a width of W and a height of H, as illustrated in Figure 1. The direction of the width is parallel with the heading angle of the target state in the target parking spot.

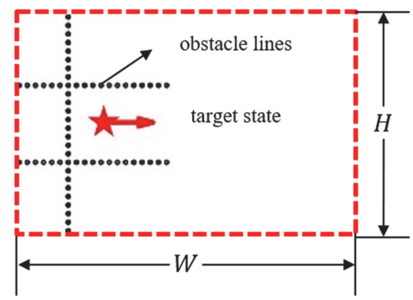

Figure 1. Neighboring interest area around the target point in the target spot.

The searching state (x, y, θ) contains the location and the heading angle in the hybrid A-star algorithm. After determining the interest area $A(s_{\text{goal}})$, we define three resolution parameters r_x, r_y, r_θ to discretize the state in $A(s_{\text{goal}})$ and construct a discrete state set $S_A = \{s(x, y, \theta)|(x, y) \in A(s_{\text{goal}}), \theta \in (-\pi, \pi]\}$ with these three resolution parameters. For example, if the interest area $A(s_{\text{goal}})$ has a width of 5 meters and a height of 4 meters and the resolutions are $r_x = 1\text{m}, r_y = 1\text{m}, r_\theta = \pi/6\text{rad}$, then the discrete state set S_A contains overall $6 \times 5 \times 12 = 360$ states.

After determining the discrete state set S_A according to the target state, we check every state s in S_A that whether there exists a collision-free RS path from the state s to the target state. The states that can be linked with the target state by a collision-free RS path will be added to the admissible set $\Omega(s_{\text{goal}})$. In practical, if we properly set the resolution parameters (as possible as small under acceptable computation cost), the admissible set $\Omega(s_{\text{goal}})$ will be not empty.

After determining the admissible set $\Omega(s_{\text{goal}})$, we dynamically choose one temporarily best middle point s_{mid} from $\Omega(s_{\text{goal}})$ at each searching step through the following optimization framework.

$$\min_{s \in \Omega} g(s) = u_1 g_1(s) + u_2 g_2(s) + u_3 g_3(s) + u_4 g_4(s) \quad (6)$$

where $g(s)$ is the cost function which is the weighted sum of four different cost subitems $g_1(s), g_2(s), g_3(s), g_4(s)$, and the constants u_1, u_2, u_3, u_4 are corresponding weights. The four cost subitems are defined as the followings.

$$g_1(s) = \rho_V(s) \quad (7)$$

$$g_2(s) = \nabla_\theta \rho_V(s) \quad (8)$$

$$g_3(s) = \Sigma_{p \in RS(s \to s_{\text{goal}})} \rho_V(p) \quad (9)$$

$$g_4(s) = \left|\theta_s - \theta_{\text{popped state}}\right|^2 \quad (10)$$

The first subitem $g_1(s)$ measures the Voronoi field potential $\rho_V(s)$ at the state s. Smaller $g_1(s)$ means smaller Voronoi field potential and farther from the obstacles at the state s. Therefore, this subitem aims to guide the middle point to stay away from the obstacles.

The second subitem $g_2(s)$ measures the gradient of the Voronoi field potential at the state s with respect to the heading angle θ of s. Smaller gradient means that the heading angle of the current state is more parallel to the direction of the drivable road. Therefore, this subitem aims to guide the middle point to own a more parallel heading angle with the direction of the drivable road.

The third subitem $g_3(s)$ measures the integral of the Voronoi field potential along the RS path from the state s to the target state. Smaller integral means that the RS path is farther from the obstacles and thus safer. Therefore, this subitem aims to guide the middle point to own a safer RS path to the target state.

The last subitem $g_4(s)$ measures the difference between the heading angles of the state s and current expanding point at current searching step which is popped from the open list. Smaller difference means that in the first planning stage from the starting state to the middle state, it is easier to find a collision-free RS path from the currently popped expanding point to the middle point. Therefore, this subitem aims to accelerate the first planning stage.

In practical engineering, there may exist many other different cost functions for various intentions under this dynamically optimized multi-stage planning framework.

4. RESULTS

To verify the proposed improved algorithm, we preset a scenario illustrated in Figure 2. The parking lot has two entrances. The black lines denote the boundaries of the parking lot and the orange lines denote the boundaries of the available parking spots. All these lines are considered as obstacle lines which the vehicle need to avoid colliding with. To better illustrate the geometry of the parking lot and focus on the trajectories, we use the format of Figure 3 to describe the path planning. The vehicle begins at the upper entrance of the parking lot and aims to park at a relatively far parking spot as shown in Figures 2 and 3. This parking lot has a size of $30\ \mathrm{m} \times 45\ \mathrm{m}$. The red stars and arrows denote the locations and headings of the starting and the goal state. The black lines denote the border lines of the parking spots and the parking lot and are considered as obstacles to avoid intersecting for the vehicle.

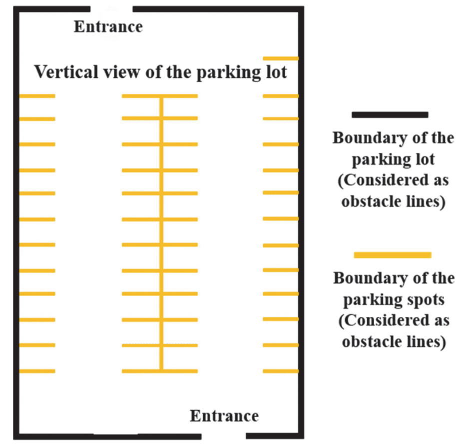

Figure 2. Layout of the parking lot.

Firstly, only the safety-enhanced design is applied to the conventional hybrid A-star algorithm and obtain the results of the conventional hybrid A-star algorithm and the improved hybrid A-star algorithm shown in Figures 4 and 5 respectively. The red line is the final generated path. The blue circles are the popped states in the searching process. The thin black lines are the locally generated trajectory from those popped states by using vehicle motion model. The parameters in this simulation are listed in Table 1. Compared with conventional hybrid A-star algorithm, the safety-enhanced design apparently improves the path safety of the portion in the blue circle in Figure 5 by guiding the path farther from the border lines of the parking spots. Therefore, it verifies the performance of the proposed safety-enhanced design.

Next, we apply the efficiency-enhanced design to the conventional hybrid A-star algorithm together with the safety-enhanced improvement. Figure 6 shows the results of the improved hybrid A-star algorithm. Compared with the conventional algorithm shown in Figure 4, the improved hybrid A-star algorithm largely enhanced the searching efficiency both in time and space. The time for the searching period is 1.9088 seconds by the conventional hybrid A-star algorithm while only 1.0740 seconds by adopting the proposed improved algorithm, which is decreased by 43.7300 %. The total number of the popped states is 64 by the conventional hybrid A-star algorithm while 38 by the proposed improved algorithm, which is decreased by 40.6250 %.

To further illustrate the performances of the proposed improved hybrid A-star algorithm, we change the heading

Table 1. Parameters in the simulation.

α	$d_O^{\max}$ [m]	w_1	w_2	w_3	u_1	u_3	u_4	W[m]	H[m]	r_x[m]	r_y[m]	r_θ[rad]
5	3	1	10	3	500	5	10	6	4	0.5	0.5	$\frac{\pi}{12}$

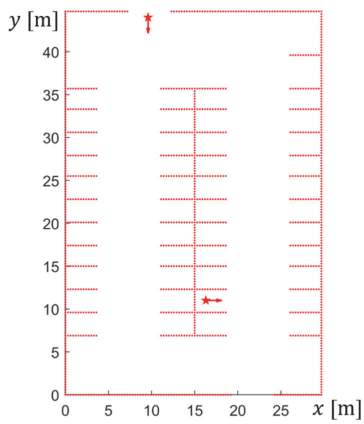

Figure 3. Illustration of the starting state and the target state.

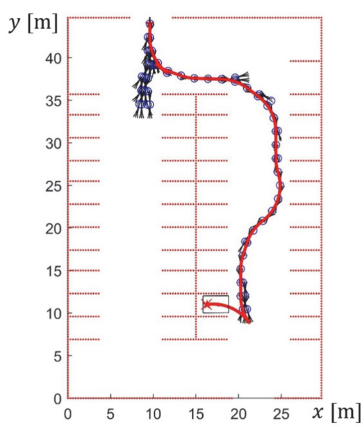

Figure 4. Searching results of conventional hybrid A-star algorithm.

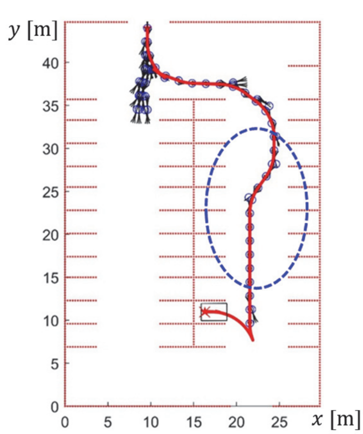

Figure 5. Searching results of the improved hybrid A-star algorithm with only safety-enhanced design.

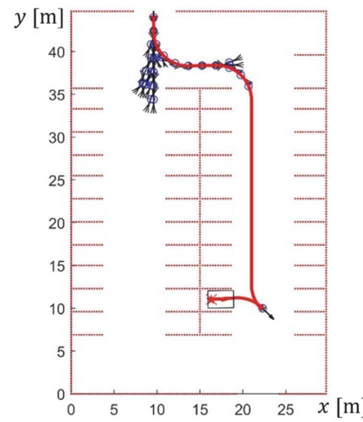

Figure 6. Searching results of the improved hybrid A-star algorithm with both safety-enhanced design and efficiency-enhanced design.

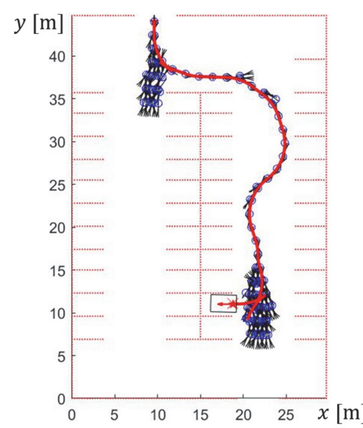

Figure 7. Searching results of conventional hybrid A-star algorithm with the opposite heading direction of the goal state.

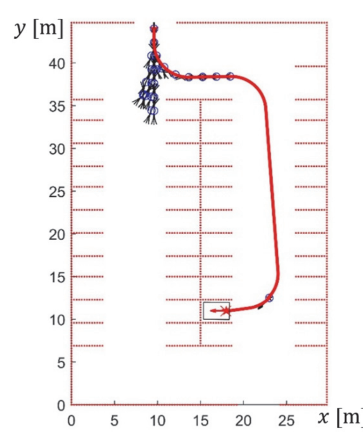

Figure 8. Searching results of the improved hybrid A-star algorithm with both safety-enhanced design and efficiency-enhanced design with the opposite heading direction of the goal state.

of the goal state to the opposite direction, i.e. the vehicle desires to park with its front facing inward. Figures 7 and 8 respectively show the results of the conventional hybrid A-star algorithm and the proposed improved hybrid A-star algorithm. The time for the searching period is 5.6730 seconds by the conventional hybrid A-star algorithm while only 1.4680 seconds by adopting the proposed improved algorithm, which is decreased by 74.1230 %. The total number of the popped states is 108 by the conventional hybrid A-star algorithm while 34 by the proposed improved algorithm, which is decreased by 68.5185 %. When the vehicle desires to park with its front facing inward, the improved searching efficiency is much apparent and significant since it needs to take lots of searching in the vicinity outside the target parking spot because of the intensive obstacle lines by the conventional hybrid A-star algorithm while the improved algorithm specially addresses this problem through the dynamically optimized multi-stage approach.

As stated above, the proposed improved algorithm has been verified to be effective and compared with the traditional hybrid A-star algorithm in terms of efficiency and safety through the simulation case. Practical hardware-in-loop experiment is also being prepared to further enhance the reliability of the proposed improved algorithm. In the following experimental verification, the vehicle dynamics control module will also be integrated into the autonomous parking system, which intervenes to control the vehicle to drive along the path generated by the proposed algorithm.

From the above results, it can be concluded that the proposed improved algorithm has the following advantages.

(1) The generated path by the proposed improved algorithm stays farther from the obstacles and thus is safer considering the actual vehicle size.
(2) The searching difficulty around the target parking spot is largely decreased.
(3) The searching efficiency in terms of both time and space is significantly improved.
(4) The pre-processing work can be repeatedly utilized considering that the environment including road conditions and parking spot layout is static.

5. CONCLUSION

This paper proposes an improved hybrid A-star algorithm for the path planning in autonomous parking in terms of collision-avoidance safety and searching efficiency. The proposed algorithm is mainly composed of two-fold designs, i.e., a safety-enhanced design and an efficiency-enhanced design. The safety-enhanced design efficiently integrates the Voronoi field potential into the searching process to take more account of collision-avoidance safety. The efficiency-enhanced design is achieved through multi-stage dynamic optimization. Both the stage dividing and the dynamic optimization efficiently speed the path searching process. Compared with the conventional hybrid A-star algorithm, the proposed improved algorithm not only generates a safer path which stays farther from obstacles, but also significantly decreases the searching difficulty and improving the searching efficiency. Although some additional pre-processing efforts are needed to deploy the proposed improved algorithm, they can be repeatedly utilized. We hope this paper will promote relative research and serve as a reference for practical engineering. Future research will be concentrated on applying the proposed algorithm to more sophisticated parking scenarios. Besides, vehicle dynamics control module will be added after the planning stage.

ACKNOWLEDGEMENT−This research is sponsored in part by the NSFC Program (No. 61872217, No. U1701262, No. U1801263). Besides, the research is also sponsored in part by the Guangdong Provincial Key Laboratory of Cyber-Physical Systems, the State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body at Hunan University (No. 32115018) and the Industrial Internet innovation and development project of ministry of industry and information technology.

REFERENCES

Anderson, S. J., Karumanchi, S. B. and Iagnemma, K. (2012). Constraint-based planning and control for safe, semi-autonomous operation of vehicles. *IEEE Intelligent Vehicles Symp. (IV)*, Madrid, Spain.

Ayalew, M., Zhou, S., Assefa, M. and Yilma, G. (2021). Spatial-temporal attentive motion planning network for autonomous vehicles. *18th Int. Computer Conf. Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Chengdu, China.

Ayawli, B. B. K., Chellali, R., Appiah, A. Y. and Kyeremeh, F. (2018). An overview of nature-inspired, conventional, and hybrid methods of autonomous vehicle path planning. *J. Advanced Transportation*, **2018**, 8269698.

Chaulwar, A., Al-Hashimi, H., Botsch, M. and Utschick, W. (2021). Sampling algorithms combination with machine learning for efficient safe trajectory planning. *Int. J. Machine Learning and Computing* **11**, **1**, 1–11.

Chen, X. and Gao, P. (2019). Path planning and control of soccer robot based on genetic algorithm. *J. Ambient Intelligence and Humanized Computing* **11**, **12**, 6177–6186.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik* **1**, **1**, 269–271.

Dolgov, D., Thrun, S., Montemerlo, M. and Diebel, J. (2010). Path planning for autonomous vehicles in unknown semi-structured environments. *The Int. J. Robotics Research* **29**, **5**, 485–501.

Dubins, L. E. (1957). On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American J. Mathematics* **79**, **3**, 497–516.

Esposto, F., Goos, J., Teerhuis, A. and Alirezaei, M. (2017). Hybrid path planning for non-holonomic autonomous vehicles: An experimental evaluation. *5th IEEE Int. Conf. Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, Naples, Italy.

Ferguson, D., Howard, T. M. and Likhachev, M. (2008). Motion planning in urban environments. *J. Field Robotics* **25**, **11-12**, 939–960.

Fraichard, T. and Scheuer, A. (2004). From Reeds and Shepp's to continuous-curvature paths. *IEEE Trans. Robotics* **20**, **6**, 1025–1035.

Furgale, P., Schwesinger, U., Rufli, M., Derendarz, W., Grimmett, H., Mühlfellner, P., Wonneberger, S., Timpner, J., Rottmann, S., Li, B., Schmidt, B., Nguyen, T. N., Cardarelli, E., Cattani, S., Brüning, S., Horstmann, S., Stellmacher, M., Mielenz, H., Köser, K., Beermann, M., Häne, C., Heng, L., Lee, G. H., Fraundorfer, F., Iser, R., Triebel, R., Posner, I., Newman, P., Wolf, L., Pollefeys, M., Brosig, S., Effertz, J., Pradalier, C. and Siegwart, R. (2013). Toward automated driving in cities using close-to-market sensors: An overview of the v-charge project. *IEEE Intelligent Vehicles Symp. (IV)*, Gold Coast, Australia.

González, D., Pérez, J., Milanés V. and Nashashibi, F. (2016). A review of motion planning techniques for automated vehicles. *IEEE Trans. Intelligent Transportation Systems* **17**, **4**, 1135–1145.

Gu, T. and Dolan, J. M. (2012). On-road motion planning for autonomous vehicles. *Int. Conf. Intelligent Robotics and Applications (ICIRA)*, Montreal, Canada.

Hart, P. E., Nilsson, N. J. and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Systems Science and Cybernetics* **4**, **2**, 100–107.

Hemmat, M. A. A., Liu, Z. and Zhang, Y. (2017). Real-time path planning and following for nonholonomic unmanned ground vehicles. *Int. Conf. Advanced Mechatronic Systems (ICAMechS)*, Xiamen, China.

Hosseininejad, S. and Dadkhah, C. (2019). Mobile robot path planning in dynamic environment based on cuckoo optimization algorithm. *Int. J. Advanced Robotic Systems* **16**, **2**, 1729881419839575.

Kala, R. and Warwick, K. (2013). Multi-level planning for semi-autonomous vehicles in traffic scenarios based on separation maximization. *J. Intelligent & Robotic Systems* **72**, **3**, 559–590.

Kamoshida, R. and Kazama, Y. (2017). Acquisition of automated guided vehicle route planning policy using deep reinforcement learning. *6th IEEE Int. Conf. Advanced Logistics and Transport (ICALT)*, Bali, Indonesia.

Kim, J. M., Lim, K. I. and Kim, J. H. (2014). Auto parking path planning system using modified reeds-shepp curve algorithm. *11th Int. Conf. Ubiquitous Robots and Ambient Intelligence (URAI)*, Kuala Lumpur, Malaysia.

LaValle, S. M. and Kuffner Jr, J. J. (2001). Randomized kinodynamic planning. *The Int. J. Robotics Research* **20**, **5**, 378–400.

Li, K., Ni, W., Tovar, E. and Guizani, M. (2020). Deep reinforcement learning for real-time trajectory planning in UAV networks. *Int. Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus.

Li, M., Zhou, P., He, X., Lv, H. and Zhang, H. (2021). Parallel parking path planning and tracking control based on adaptive algorithms. *Int. J. Automotive Technology* **22**, **4**, 949–965.

Naveed, K. B., Qiao, Z. and Dolan, J. M. (2021). Trajectory planning for autonomous vehicles using hierarchical reinforcement learning. *IEEE Int. Intelligent Transportation Systems Conf. (ITSC)*, Indianapolis, Indiana, USA.

Petereit, J., Emter, T., Frey, C. W., Kopfstedt, T. and Beutel, A. (2012). Application of hybrid A* to an autonomous mobile robot for path planning in unstructured outdoor environments. *7th German Conf. Robotics (ROBOTIK)*, Munich, Germany.

Ravankar, A., Ravankar, A. A., Kobayashi, Y., Hoshino, Y. and Peng, C. C. (2018). Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. *Sensors* **18**, **9**, 3170.

Reeds, J. and Shepp, L. (1990). Optimal paths for a car that goes both forwards and backwards. *Pacific J. Mathematics* **145**, **2**, 367–393.

Sedighi, S., Nguyen, D. V. and Kuhnert, K. D. (2019). Guided hybrid A-star path planning algorithm for valet parking applications. *2019 5th Int. Conf. Control, Automation and Robotics (ICCAR)*, Beijing, China.

Shamsudin, A. U., Ohno, K., Hamada, R., Kojima, S., Mizuno, N., Westfechtel, T., Suzuki, T., Tadokoro, S., Fujita, J. and Amano, H. (2017). Two-stage hybrid A* path-planning in large petrochemical complexes. *IEEE Int. Conf. Advanced Intelligent Mechatronics (AIM)*, Munich, Germany.

Sheng, W., Li, B. and Zhong, X. (2021). Autonomous parking trajectory planning with tiny passages: A combination of multistage hybrid A-Star algorithm and numerical optimal control. *IEEE Access*, **9**, 102801–102810.

Tu, K., Yang, S., Zhang, H. and Wang, Z. (2019). Hybrid A∗ based motion planning for autonomous vehicles in unstructured environment. *IEEE Int. Symp. Circuits and Systems (ISCAS)*, Sapporo, Japan.

Vorobieva, H., Glaser, S., Minoiu-Enache, N. and Mammar, S. (2014). Automatic parallel parking in tiny spots: Path planning and control. *IEEE Trans. Intelligent Transportation Systems* **16**, **1**, 396–410.

Yu, L., Shao, X., Wei, Y. and Zhou, K. (2018). Intelligent land-vehicle model transfer trajectory planning method based on deep reinforcement learning. *Sensors* **18**, **9**, 2905.

Zhang, S., Chen, Y., Chen, S. and Zheng, N. (2019). Hybrid A*-based curvature continuous path planning in complex dynamic environments. *IEEE Intelligent Transportation Systems Conf. (ITSC)*, Auckland, New Zealand.

Zhang, J., Shi, Z., Yang, X. and Zhao, J. (2020). Trajectory planning and tracking control for autonomous parallel parking of a non-holonomic vehicle. *Measurement and Control* **53**, **9-10**, 1800–1816.

Zhou, W., Jiang, K., Cao, Z., Deng, N. and Yang, D. (2020). Integrating deep reinforcement learning with optimal trajectory planner for automated driving. *IEEE 23rd Int. Conf. Intelligent Transportation Systems (ITSC)*, Rhodes, Greece.