Annual Reviews in Control 48 (2019) 147–164

Contents lists available at ScienceDirect

Annual Reviews in Control

journal homepage: www.elsevier.com/locate/arcontrol

Collision avoidance of multi unmanned aerial vehicles: A review

Sunan Huang[a,*], Rodney Swee Huat Teo[a], Kok Kiong Tan[b]

[a] Temasek Lab., National University of Singapore, Singapore
[b] Electrical and Computer Engineering, National University of Singapore, Singapore

ARTICLE INFO

Article history:
Received 3 April 2019
Revised 28 July 2019
Accepted 1 October 2019
Available online 24 October 2019

Keywords:
Collision avoidance
Multiple unmanned aerial vehicles
Obstacles
Unmanned systems

ABSTRACT

The control of a multiple unmanned aerial vehicle (UAV) system is popular and attracting a lot of attentions. This is motivated by many practical civil and commercial UAV applications. Collision avoidance is the fundamental in motion planning of multi-UAVs, especially for large teams of UAVs. Although several collision avoidance approaches have been reported, there is a lack of highlighting the key components shared by these approaches. In this work, we aim to provide researchers with a state-of-the-art overview of various approaches for multi-UAV collision avoidance. The existing works on collision avoidance are presented through several classifications based on algorithm used and frameworks designed, and their main features are also discussed. A discussion on the literature summary in multi-UAV collision avoidance is given, Finally, the challenges in the research directions are presented.

Contents

* Corresponding author.
E-mail address: tslhs@nus.edu.sg (S. Huang).

https://doi.org/10.1016/j.arcontrol.2019.10.001
1367-5788/

1. Introduction

Currently, multiple unmanned aerial vehicles (UAVs) with embedded sensor and communication devices are attracting growing interest. This is motivated by growing number of everyday available civil and commercial UAVs and their applications (Cao, Teo, Huang, & Ren, 2019; Dalamagkidis, Valavanis, & Piegl, 2009; Huang et al., 2018; Huang, Teo, Liu, & Dymkou, 2017a; Huang, Teo, Liu, & Dymkou, 2019). One of the core problems in an unmanned system is navigation which is formulated as driving vehicle safely from one place to its goal without colliding with other obstacles. The navigation is often decomposed into two problems: global path planning and local collision avoidance. Global path planning generates a set of waypoints, from an initial position to a final goal point, passing through obstacles in a working space, while local collision avoidance takes a given waypoint assignment as a local goal to avoid obstacles. Path planning has attracted a lot of attention since the 1970s (Sariff & Buniyamin, 2006). Various algorithms have been reported over the past forty years. These algorithms include potential fields method (Cui et al., 2017), heuristic search method (Warren, 1993), and graph method (Lozano-Perez, 1983). Since path planning needs to consider avoiding obstacles, in this paper, both are referred to collision avoidance problem and our survey includes these two parts.

Early collision avoidance strategy focuses on the static obstacle handling by sensing around (Borenstein & Koren, 1991; Lozano-Pérez & Wesley, 1979), using decision trees to avoid various troublesome situations (Minguez & Montano, 2004), and using path planning to avoid the obstacles (Bellingham, Tillerson, Richards, & How, 2003). As moving obstacles are more often found in a real environment, many techniques have been proposed for dealing with this situation. For example, in Watanabe, Calise, Johnson, and Evers (2006), the authors develop a vision-based collision avoidance by using the minimum effort guidance; in Mujumdar and Padhi (2012), the authors present a reactive avoidance method by using the nonlinear differential geometric guidance; in Nikolos, Valavanis, Tsourveloudis, and Kostaras (2003) and Borrelli, Subramanian, Raghunathan, and Biegler (2006), the authors develop a local path planning method for UAV navigation; in Kochenderfer, Holland, and Chryssanthacopoulos (2012), the authors present a collision avoidance algorithm by using the dynamic programming; in Arambula Cosio and Padilla Castaẽda (2004), the authors propose a collision avoidance algorithm based on the potential fields; in Fiorini and Shiller (1998), van den, Guy, Lin, and Manocha (2011), Conroy, Bareiss, Beall, and van den Berg (2014), Kluge and Prassler (2006), Snape, van den Berg, Guy, and Manocha (2011) and Allawi and Abdalla (2014), the authors present the velocity obstacle method for deconflicting control of UAVs. These techniques are further divided into non-cooperative avoidance approach and cooperative avoidance approach. Here, non-cooperative approaches have no prior information about obstacles by using communication, while cooperative approaches use cooperative communication to share their information with each other. For example, in Yang, Mejias, and Bruggemann (2013), a non-cooperative strategy use passive sensors to identify obstacles and avoid them; in Jenie, Kampen, Visser, and Chu (2014), based on the assumptions that the distance, directions, and the range of speed of the obstacle are known, another non-cooperative avoidance technique is developed. In multi-UAV systems, cooperative approaches are desirable, for the safety and design freedom purposes. The cooperative collision avoidance techniques can be found in J. Kochenderfer et al. (2012), Fiorini and Shiller (1998), van den Berg et al. (2011), Allawi and Abdalla (2014) and Jenie, Kampen, Visser, Ellerbroek, and Hoekstra (2016). On the other hand, most existing techniques, however, are implemented in simulation level. Applying proposed algorithms on real vehicle poses a challenge, since many algorithms have strong assumptions which may be not realistic in a real world or they are very difficult to realize in hardware. Implementation in real-flight experiments of the proposed approaches can be found in Roelofsen, Gillet, and Martinoli (2015), Anderson (2011) and Valbuena and Tanner (2012), where the work of Roelofsen et al. (2015) uses cameras to implement reciprocal collision avoidance algorithm on three quadrotors, the work of Anderson (2011) realizes a 3D reactive collision avoidance algorithm (Lalish, 2009) on quadrotors, the work of Valbuena and Tanner (2012) implements potential function strategy on a small four-wheel drive mobile robot with skid steering.

This paper serves as a review for these works and provides a clear trend vision for readers. The main contributions of this paper are twofold: (1) we give a detailed classification and comparison among existing approaches; (2) an analysis of concerning about the collision avoidance problem is given, especially from a practical application viewpoint.

The remainder of this paper is organized as follows: Section 2 presents the evolution of multi-UAV collision avoidance. Section 3 presents the overview of collision avoidance control architectures. In Section 4, six classes are given for existing collision avoidance approaches and detailed illustration is presented. In Section 5, we summarize the works presented in this paper and give our comments regarding these works. In Section 6, we give our view points of challenging work in the collision avoidance problem. Section 7 concludes the paper.

2. Evolution of collision avoidance of multi-UAVs

Before we embark on an in-depth discussion on various technologies, we briefly introduce the evolution of path planning and collision avoidance in multi-UAV control.

Path planning research has attracted attention since the 1970s. The idea is to construct a configuration space, determining those parts of free space which a reference point of the moving object can occupy with avoiding obstacles. Then, a shortest path is found for the reference point through this free space. This approach is mainly focused on handling static obstacles and is suitable for single UAV.

In the 1980s,constructing a representation of C configuration space, was the predominate approach to motion planning. Several improved path-planning algorithms have been proposed. However, one major problem with these algorithms is obstacle avoidance while minimizing the input energy and getting the optimum results.

In the middle 1980s, some works used different techniques to create obstacle models to obtain the optimal results in a real-world system. They introduced many formal algorithms for collision avoidance such as decision theory, potential function, control theory and other heuristic approaches. For example, artificial potential field designs a energy function, called the potential field function (Khatib, 1986) and the energy is minimized by following the negative gradient of the potential energy function. The controlled UAV moves to a lower energy configuration. Although these results can be used for multi-UAV systems, they need further development for efficient application in various scenarios, especially in dealing with multiple moving obstacles. This problem is even more challenging, when designing real-time path planning of multi-UAVs in a rapidly changing environment.

In the late 1990s, several local planning technologies were developed to provide a fast response even though the global solution has been lost. This kind of approaches may be sufficient, i.e. respond appropriately to current sensing information. Examples of such approaches include dynamic window technology (Fox, Burgard, & Thrun, 1997), inevitable collision states (Petti & Fraichard, 2005), and velocity obstacles (Fiorini & Shiller, 1998). However, they do not optimize the trajectories subject to time or energy. This may lead UAVs to a deadlock, i.e., “a robot is temporally blocked by a dynamic obstacle and can not make progress towards achieving its mission, at run time” (Alonso-Mora, DeCastro, Raman, Rus, & Kress-Gazit, 2018).

In the 2000s, some works focus on particular mechanisms for coordinating among multi-UAVs without collision. Most of the works proposed a safe trajectory planning for large UAV teams. These multi-UAV motion planning schemes have been shown to be effective in practice.

In short, the works on the motion planning of multi-UAVs are depicted in Fig. 1. It is observed from existing results that we give a short summary.

- The use of multiple UAV teams is increasing.
- There have been attempts at coordination and motion planning control of multi-UAV teams in complex scenarios
- Distributed architectures for multi-UAV systems are gaining attention because of the growth of the size of the state space of the multi-UAVs.

Year	Works and development
2000's	Motion planning of multi-UAV teams in complex environments
1990's	Local motion planning of multi-UAVs
1985's	Path planning with considering uncertain and dynamical environments
1970's	Classical path planning. Mainly focused on on dealing with static obstacles

Fig. 1. Model of a collision avoidance system.

3. Overview of collision avoidance control architectures

In this section, we briefly give an overview of the planning and control architectures that are used in the literature and implemented to meet the collision avoidance requirement. They include reactive control, deliberative planning and hybrid integration control.

The control architecture of a robot defines how the works of generating actions from perceiving the world around us are organized. Traditionally, robot control community has attempted to solve the task planning and collision-free motion planning separately. The task planning uses deliberative planning architecture, while the collision-free control uses reactive control architecture. As UAV is regarded as a mobile robot, it uses the same architectures in collision avoidance control.

In deliberative planning, with an a global world, UAV senses the world and updates world model. According to this world, UAV adopts automated reasoning and determines an optimal sequence plan of collision-free actions to achieve the desired goal prior to motion execution. Once a plan has been determined, it is executed by the control. Fig. 2 shows the deliberative architecture. However, this approach requires an accurate model of environment and takes a longer time to perform necessary calculations, in order to plan a global path. Therefore, it is not proper in uncertain environments and real-time control.

In reactive control, UAV uses the locally sensed information about the environment and reacts directly to sensor information. This control allows the UAV to respond very quickly to the environments. Fig. 3 represents the architecture of reactive control approach. However, the solution obtained from this approach may stuck in a local minimum that is worse than the global one. To overcome the drawbacks of both architectures, it is natural to develop a hybrid control to enhance navigation performance in a real world. In hybrid control, it is desire to combine the best features of both deliberative and reactive control. One typical hybrid architecture (Nakhaeinia, Tang, Noor, & Motlagh, 2011) is consisted of three layers: deliberative layer, control execution layer and reactive control layer (Fig. 4). The deliberative layer develops an optimal collision-free plan for high level issue. Then, the optimal plan from the higher level is transmitted to the reactive layer to gener-

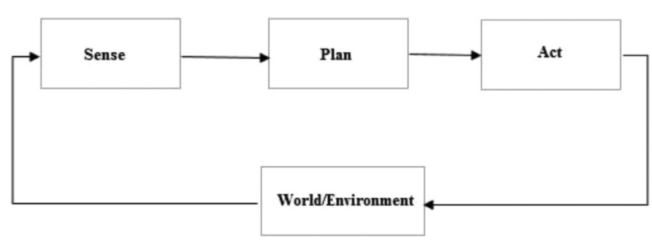

Fig. 2. Deliberative planning architecture.

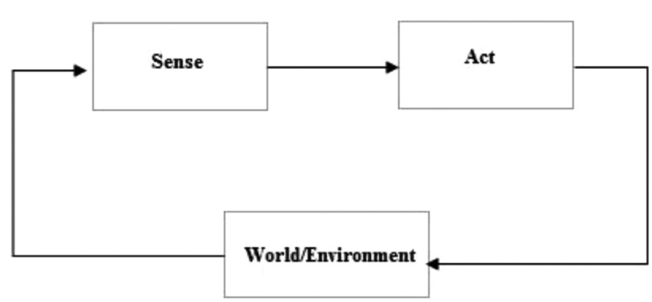

Fig. 3. Reactive control architecture.

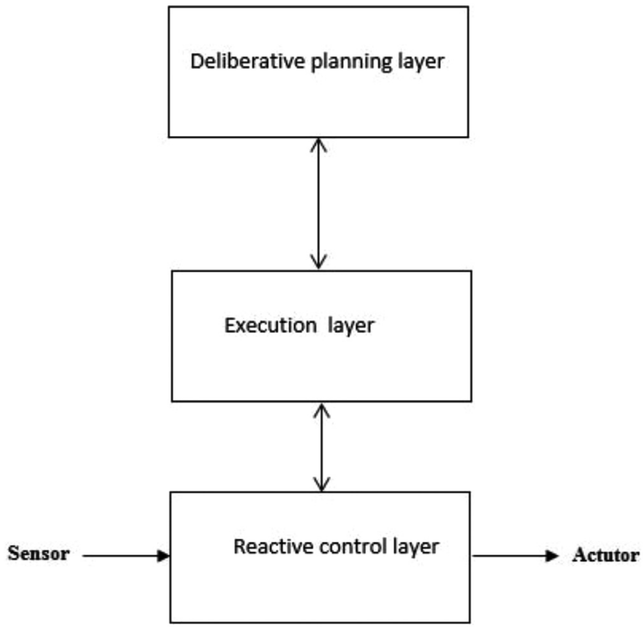

Fig. 4. Hybrid control architecture.

ate the UAV's control action. The execution layer is used to connect the high-level layer with the low-level layer.

4. Collision avoidance control strategies

The motivation for collision avoidance may stem from the increasing safety requirement of air traffic control, shipping traffic and multi unmanned aerial vehicle system, where collisions have to be considered to avoid at all situations.

In recent years, various strategies have been proposed for collision avoidance control purpose. The basic idea behind the collision avoidance algorithms is to design a control signal which can result a conflict-free trajectory for a vehicle. Several techniques already exist for collision avoidance control: path selection, graph-based path selection, rule based control, deterministic optimal control, stochastic optimal control, protocol-based control, linear and nonlinear model predictive controls, original potential field function, virtual force function, Harmonic potential function, improved potential field, navigation function, collision cone, and velocity obstacle approaches. Furthermore, according to the different objective functions, we group techniques such as rule based control, deterministic optimal control, stochastic optimal control, and protocol-based control together, called the conflict resolution approach, while we group original potential field function, virtual force function, Harmonic potential function, improved potential field, and navigation function into the potential field function approach. Similarly, we group linear and nonlinear model predictive controls into the model predictive control approach. The global path planning method can also be used for collision avoidance and thus the path selection and graph-based path selection are grouped into the path planning approach. Using geometric concept to design collision avoidance algorithm differs from other groups and thus the collision cone and velocity obstacle techniques are grouped into the geometric guidance approach. Some approaches emphasize motion planning in multi-UAV team. Thus, conflict-based search for multi-agent systems, multi-UAV trajectory planning of large UAV teams, prioritized planning and formal verification are grouped into the motion planning of teams of multi-UAVs. Finally, existing works in designing collision avoidance control fall into six categories (ref Fig. 5):

1) Path planning
2) Conflict resolution
3) Model predictive control
4) Potential field
5) Geometric guidance
6) Motion planning of teams of multi-UAVs

Path planning methods aim to find the shortest path in a map or data base graph where obstacle edges are known already. Conflict resolution methods consider obstacle vehicles whose

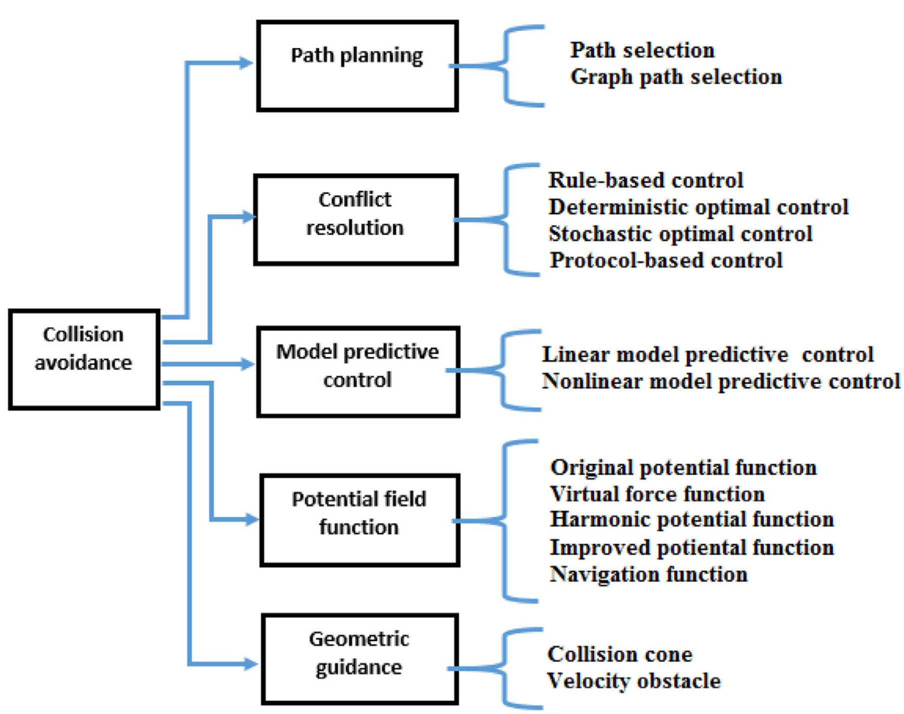

Fig. 5. Collision avoidance and its group.

trajectories are known and find a flyable path (free-route) for a controlled vehicle during some time interval, i.e., planning a trajectory ahead of time. Model predictive control methods consider a model of the vehicle to predict the future behavior of the state in order to optimize the control while limiting the input and state to the admissible sets and obstacles. Potential field methods consider each vehicle in working area to compute the feedback control by using a repulsive function. Geometric guidance methods aim to reactively produce an avoidance control based on conflict geometry. Motion planning of teams of multi-UAVs concentrates on planning in UAV team without collision. The detailed comments regarding these approaches are given below.

4.1. Path planning

This method can be further classified as two groups: path selection and graph path selection.

4.1.1. Path selection

It uses an offline/online planner to produce a curve that connects the starting and target points with a predefined initial direction. The proposed method doesn't need to have a vehicle dynamical model. It is concerning about generating collision free paths in environments, avoiding the obstacles. For example, the work in Nikolos et al. (2003) computes a curved path with desired characteristics in a three-dimensional (3-D) rough environment based on an evolutionary algorithm; the work in Yang et al. (2013) calculates the path of an aircraft flying at a constant velocity by assuming that both moving and static obstacles are particles. The simplest form of this method for vehicles can be given as visiting a set of waypoints, while they try to avoid the obstacles. A typical example is illustrated in Bellingham et al. (2003). Consider a vehicle team which is consisted of N_v vehicles with known waypoint locations, and obstacles which are bounded by polygons. The path planning algorithm generates a detailed trajectory for each vehicle to reach its final waypoint by using a cost which is given by

$$\bar{t} = \max_p t_p \tag{1}$$

$$J = \bar{t} + \frac{\alpha}{N_v}\sum_{p=1}^{N_v} t_p \tag{2}$$

where p represents the pth vehicle, t_p is the time at which vehicle p arrives final destination, and α weights the average completion time.

4.1.2. Graph path selection

This method is to plan a safe path for a polyhedral object moving among known environment. The algorithm requires the obstacles which are represented by polygons so that they can be handled by the graph-based optimization approach. The task is to plan a path which avoids all obstacle polygons. This can be accomplished by searching a path through a graph connecting vertices of the obstacle polygons. A typical optimization approach in this area as shown in the work Lozano-Pérez and Wesley (1979) is to use visibility algorithm which can be used to find Euclidean shortest paths among a set of polygonal obstacles in an environment. The work of Lozano-Pérez and Wesley (1979) is 2D visibility, while the work in Huang and Teo (2019) is a 3D version. However, these works are not suitable for moving obstacles. An extension (El. Khaili, 2014) to this work is to search the path against moving obstacles, where the obstacles are assumed to move at constant speed along a three-dimensional (3D) space at some time. Therefore, the shortest path is obtained by finding straight line segments connecting the vertices visible obstacles (3D object).

In general, global path planning methods as stated in the introduction, can be used in collision avoidance. Some traditional search algorithms for the shortest path problem, for instance, A^* algorithm (Warren, 1993), Dijkstra's algorithm (Dijkstra, 1959) and improved hierarchical search (Zhang, Huang, Liang, & Tan, 2019), can also be used in the collision avoidance problem.

4.2. Conflict resolution approach

Since multi-vehicle systems range widely in type and mission, it is not practical to require them depending on a ground station and predefined trajectories. The use of the self-separation (free-flight) concept (Kuchar & Yang, 2000; Tomlin, Pappas, & Sastry, 1998; Yang & Kuchar, 1997) has been suggested. However, manual control is difficult to implement the free flight task, due to human reaction time delay. Currently, scientists are studying on an conflict detection and resolution approach to execute the task. Various solutions to the conflict detection and resolution have been proposed, including rule-based approach, deterministic and stochastic approaches, and protocol-based approach.

4.2.1. Rule-based control

In Pallottino, Scordio, Bicchi, and Frazzoli (2007b), the authors present a collision avoidance policy in a multi vehicle system. Each vehicle in the system decides its own control by using this policy which is a pre-described rule base. However, the rules-based method may not be properly account for unexpected events.

4.2.2. Deterministic optimal control

On the other hand, conflict resolution can be refined as an optimal control problem where an objective function is minimized, while without collision occurrence. There are two types of solutions to this problem: deterministic and stochastic optimal controls. For deterministic control problems, vehicle dynamics can be represented as ordinary differential equations so there are many numerical methods which can be used for yielding the optimal trajectory. For example, in Tomlin et al. (1998), the authors propose a method for producing safe conflict resolution for two vehicles, where the solution is to compute reachable sets which are given by solving the optimal Hamilton function; in Borrelli et al. (2006), the authors convert conflict resolution problem to a finite dimensional nonlinear program by considering the constraints of the obstacles; in Delsart, Fraichard, and Martinez-Gomez (2009), the authors present a trajectory generation algorithm that computes a feasible trajectory which reaches the destination at a prescribed final time in order to avoid collision with the moving obstacles; in Gerdts, Henrion, Hömberg, and Landry (2012), the conflict resolution is reformulated as a sequential programming with the anti-collision constraints and it produces a convex optimization problem. The deterministic approach is typically illustrated by the work given by Borrelli et al. (2006). Consider the vehicle dynamics

$$\frac{dx}{dt} = v_x(t) \tag{3}$$

$$\frac{dy}{dt} = v_y(t) \tag{4}$$

$$\frac{dz}{dt} = v_z(t) \tag{5}$$

where x, y, z represent the positions of the vehicle along the 3-dimensional axes, and v_x, v_y, v_z represent velocities in the X,Y,Z directions, respectively. The optimal trajectories are obtained by as minimizing the following cost function,

$$\min \sum_{i=1}^{n} J_i(v_x, v_y, v_z) \tag{6}$$

$$\begin{aligned} &\text{s.t.}\\ &\textit{vehicle dynamic}\\ &\textit{separation constraints between}\\ &\textit{different pairs of vehicles}:\\ g_{ineq}(x_i(t), y_i(t), z_i(t), x_j(t), y_j(t), z_j(t)) = 0,&\\ &\textit{all obstacle constraints}:\\ &g^{ineq}_{obs}(x_i(t), y_i(t), z_i(t)) = 0. \end{aligned}$$

Since different vehicle pairs and obstacles are included in the constraints, an optimal trajectory can be obtained without colliding with obstacles.

4.2.3. Stochastic optimal control

For stochastic control problems, the state of the vehicle system is represented by a stochastic process and optimal trajectory is generated by solving an optimization porobplem. For example, in Matsuno and Tsuchiya (2014a), the authors explore conflict detection problem with consideration of system uncertainties, where vehicle dynamics are represented by a set of stochastic differential equations and the flight trajectory is obtained by solving the stochastic optimal control problem; in Matsuno and Tsuchiya (2014b), the authors extend the result of Matsuno and Tsuchiya (2014a) to conflict resolution problem, where the stochastic conflict resolution optimal control is solved by combining the generalized polynomial chaos method with the conflict detection algorithm (Matsuno & Tsuchiya, 2014a); in Liu and Hwang (2014), the authors consider a stochastic optimal control problem in the presence of uncertainties in both vehicle and wind dynamics, where confliction resolution is obtained by using a stochastic approximation algorithm based on the Jacobi iteration. The stochastic control approach can be typically demonstrated by the work of Liu and Hwang (2014). Consider the stochastic vehicle model

$$dX_t = f(X_t, u_t)dt + g(X_t, u_t)dB_t \tag{7}$$

where X_t is the vehicle state vector, u_t is the control input to the vehicle, B_t is the Brown motion, and f and g are continuous functions. Let S be a closed, bounded, and connected set. Define τ as an exit time which is the first time when the vehicle state X_t leaves out the set S. The stochastic optimal control is based on a cost function given by

$$V = E[\int_0^{\tau} c(X^u_t, u_t)dt + r(x^{u_t}_{\tau})] \tag{8}$$

where $c(\cdot)$ is the running cost function and $r(\cdot)$ is the exit cost function. The authors in Liu and Hwang (2014) also define target set D_0, convective weather cell set D_1 and flight region bounded by the set D_2. Thus, the conflict resolution control as shown in Fig. 6, is that for a given the set S, the optimal control u should drive X_t to enter D_0 without colliding D_1. The optimal control is derived by minimining the cost function subject to vehicle dynamics and constraints D_0, D_1, D_2. From a practical view point, the stochastic conflict resolution is more reasonable than the deterministic approach, for it considers various uncertainties during flight. However, the stochastic control approach entails a heavier computational load.

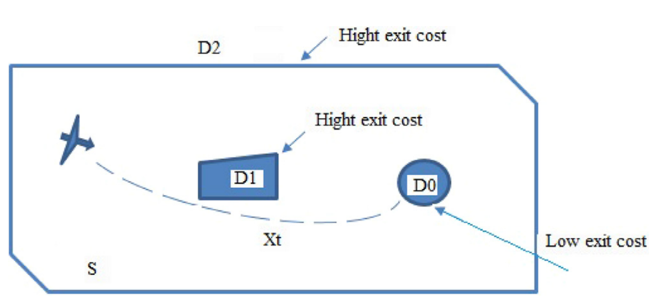

Fig. 6. Illustrative example of conflict control.

4.2.4. Protocol-based control

The last solution to the conflict resolution control is protocol-based approach. In Hwang and Tomlin (2002), the authors assume that all vehicles at the same altitude can change their heading at the same time and present a distributed protocol with guaranteed safety for a multi-vehicle system. This protocol is first obtained from an exact conflict in which all vehicles collide at an exact point; then the protocol is extended to the inexact conflict case. In Stroe and Andrei (2017), the authors propose a new protocol for a general inexact conflict. Still, this protocol requires to only change heading for all vehicles. In Stanley (2005), the author extends the work of Hwang and Tomlin (2002) to 3D case. Unfortunately, the protocol derived is based on an ideal transformation to exact conflict. It is questionable if the proposed protocol can work well at inexact conflict case.

4.3. MPC approach

In recent years model predictive control (MPC) has attracted increasing attention for multirotors due to its capability of simultaneously handling with different constraints and cost functions by optimization (Huang, Tan, & Lee, 2002; Muske & Rawlings, 1993; Nguyen, Tan, & Huang, 2011). Naturally, MPC is also applied to collision avoidance. In this section, we present the MPC collision avoidance strategies based on the vehicle model which usually yields a solution describing the outcome of some measurements. Several algorithms are presented. According to the predictive models used, MPC approach is divided into two classes: linear and nonlinear MPCs. These results are briefly discussed below.

4.3.1. Linear MPC approach

Linear MPC is referred to control algorithms that compute a state profile by using a linear model to optimize a linear or quadratic open-loop performance function subject to linear constraints over a future time horizon. Briefly speaking, a linearized model in linear MPC is used and the receding horizon approach is applied to produce the optimal control sequence. For collision avoidance, a UAV problem is required to obey obstacle constraints.

In Richards and How (2003), the authors use a discretized linear vehicle dynamics to predict the future state profile and linear program (LP) to generate optimal control, including constraints such as obstacles and goal assignment. Since sometimes the LP optimization may not give a feasible solution due to a disturbance, the authors propose the corrections to state prediction to include the disturbance. Utilizing these corrections guarantee robust feasibility without increasing the problem complexity.

In Alrifaee, Kostyszyn, and Abel (2016), the linear MPC is applied to design a controller for collision avoidance of networked vehicles, where vehicle-to-vehicle communication is used to broadcast the state information of the controlled vehicle to other vehicles. The vehicle model is linearized by using Taylor series around a steady state operation point, while the vehicle collision avoidance is considered as the constraints which are formulated as the quadratic functions. Finally, the vehicle control is obtained by using MPC optimization.

On the other hand, design and implementation of linear MPC is a challenging problem, requiring optimization knowledge and coding skills. The work of Richards, Kuwata, and How (2003) gives a detailed experimental result, where the multi-vehicle testbed based on MPC is designed and the architecture is discussed.

4.3.2. Nonlinear MPC approach

It is well-known that the UAV dynamics is nonlinear. The linearization of the original nonlinear dynamics is valid only in a small region around an operating point, thereby resulting in the MPC solution is approximate, even LP solution is called the global one in the linear MPC problem space. The worst thing about linear MPC is that the richness of the non-linear system is lost. This motivates the development of the nonlinear MPC for collision avoidance.

In Chao, Zhou, Ming, and Zhang (2012), the authors present a collision-free formation control of multi-vehicles based on the nonlinear MPC. Considering static circular obstacles with a safe radius, the authors define a dangerous distance l_D, when the distance between the controlled vehicle and obstacle is less than l_D, a cost function designed is added to the vehicle MPC objective function and vehicle starts to avoid it while achieving formation.

In Nikou, Verginis, Heshmati-alamdari, and Dimarogonas (2017), the authors present a nonlinear MPC algorithm in a workspace with obstacles which are assumed to be ellipsoids form, where the control objective is that the N-vehicles collaborate with each other to grasp an object subject to control constraints. Utilizing the nonlinear MPC, the authors prove the feasibility and convergence of the proposed algorithm. Simulation shows the effectiveness of the proposed approach.

In Carvalho et al. (2013), the authors develop a tailored scheme for solving the nonlinear MPC problem for an unmanned ground vehicle. The nonlinear vehicle dynamics is linearized at each time step, while the nonlinear obstacle constraints are approximated as linear convex expression. Thus, the nonlinear MPC optimization becomes as a sequential quadratic programming problem. The computational load of the proposed algorithm is decreased.

In Hang, Huang, Chen, and Tan (2020), the authors present a nonlinear MPC scheme incorporating 2D visibility, where the visibility finds a global waypoints without collision with static obstacles, and MPC scheme avoids the moving obstacles locally to reach a closest waypoint. The vehicle dynamic is given by

$$x_{k+1} = f(x_k, u_k) \tag{9}$$

$$y_k = g(x_k, u_k) \tag{10}$$

At each time step we have to find the optimal MPC control sequence over a finite future horizon of N steps. The relevant MPC optimization formulation is given by

$$J = \min_{u_t,\dots,u_{t+N-1}} \left\{ \sum_{k=0}^{N-1} \|y_{t+k} - y_r(t)\|_Q^2 + \rho \|u_{t+k}\|_R^2 \right\} \tag{11}$$

$$s.t.: x_{t+k+1} = f(x_{t+k}, u_{t+k}) \tag{12}$$

$$y_{t+k} = g(x_{t+k}, u_{t+k}) \tag{13}$$

$$u_{\min} \le u_{t+k} \le u_{\max} \tag{14}$$

$$y_{\min} \le y_{t+k} \le y_{\max} \tag{15}$$

$$x_t = x(t), k = 0, \dots, N-1 \tag{16}$$

where $u_{\min}$, $u_{\max}$ are the control input range, while $y_{\min}$, $y_{\max}$ are the output range. Moreover, unlike the previous results (Carvalho et al., 2013; Chao et al., 2012; Nikou et al., 2017), the work in Hang et al. (2020) develops a probability model to estimate the future position and velocity of a moving obstacle since no communication is available. Thus, the probability density and belief values are added into the objective function (11) as a constraint violation penalty. However, this result assumes that the moving obstacle follows a nonlinear Gaussian distribution and MPC solution may be local one. In Zhu and Alonso-Mora (2019), the authors present a probabilistic collision avoidance approach for multi-UAVs that accounts for uncertainties in both UAV and motion. The proposed approach uses the collision probability between each UAV and obstacle, this is called the chance constraints. It assumes that the uncertainties are Gaussian distributed and transforms the chance constraints into deterministic linear constraints. Such a design can avoid local minima while optimizing MPC problem. This approach is a distributed control and can generate more efficient trajectories for the UAVs while maintaining safety. However, the MPC control is a local one, thereby resulting a local optimal solution. In Zhang, Ma, Huang, Cheng, and Lee (2019), the authors propose an integration algorithm for path planning in a 3D environment. The global path planning is obtained by using a hierarchical search, while the local avoidance law uses a nonlinear MPC. Thus, the result in Zhang, Ma et al. (2019) can avoid local solution and achieve an approximated optimal control.

4.4. Potential field function

The potential field is a useful tool in vehicle motion control. The basic idea behind the potential field is as follows: We design an attractive potential at the target and a repulsive potential is constructed for the obstacles. Thus, during vehicle navigation, the vehicle follows the gradient of potentials to yield the path which is attracted by the target and repelled by obstacles. Various potential field algorithms have been developed in the past two decades. Several typical works are discussed below.

4.4.1. Original potential field approach

The potential field is originally suggested by Khatib (1986), where a vehicle is treated as a point and a single obstacle O is in a search space. Assuming that x_d is the desired target position, the vehicle is controlled to avoid the obstacle O subject to the potential field

$$U = U_{x_d} + U_O \tag{17}$$

Applying the potential field function, the command is given by

$$F = F_{x_d} + F_O \tag{18}$$

with

$$F_{x_d} = -\mathrm{grad}[U_{x_d}] \tag{19}$$

$$F_O = -\mathrm{grad}[U_O] \tag{20}$$

where F_{x_d} represents an attractive force allowing the vehicle to reach the desired target x_d, and F_O represents a repulsive force from the obstacle created by the field U_O. This control can lead a collision avoidance algorithm for the vehicle motion control. The authors in Arambula Cosio and Padilla Castaẽda (2004) apply the original potential field to collision avoidance problem. However, the original potential field has some inherent limitations such as trap situation due to local minima and oscillation situations. Some progress has been made in improve the original potential field approach. These include virtual force approach,harmonic potential approach, and improved potential field approach.

4.4.2. Virtual force approach

The idea of this approach is to combine the potential field approach with a certainty grid to generate motion control for mobile vehicles (Borenstein & Koren, 1989; Koren & Borenstein, 1991). Assume that the vehicle's sensor can scan the size of the window is $m \times m$ cells. A repulsive force at each cell to the vehicle is given by

$$F(i,j) = F_{cr}(i,j)\left[\frac{x_i - x_0}{d(i,j)}\dot{x} + \frac{y_i - y_0}{d(i,j)}\dot{y}\right] \tag{21}$$

where F_{cr} is the factor corresponding to the i, jth cell,$d(i, j)$ is the distance between the cells, (x_0, y_0) is the vehicle's coordinates, (x_i, y_i) is the cell's coordinates, and $(\dot{x}, \dot{y})$ is the velocity along x-and y-axes, respectively. The total repulsive force F_r is given by

$$F_r = \sum_{i,j} F(i, j). \tag{22}$$

The attractive force F_t is produced by the target and it is given by

$$F_t = F_{ct}\left[\frac{x_t - x_0}{d}\dot{x} + \frac{y_t - y_0}{d}\dot{y}\right] \tag{23}$$

where F_{ct} is the factor, d is the distance between the vehicle and target, and (x_t, y_t) is the target's position. This approach can solve the issues of both trap states and oscillation situation.

4.4.3. Harmonic potential approach

Since a harmonic function is the principle of superposition, this attracts scientists to use it as a potential function (Connolly, B. Bums, & Weiss, 1990; Kim & Khosla, 1992; Panati, Baasandorj, & Chong, 2015; Shi, Zhang, & Peng, 2007). For example, in Kim and Khosla (1992), the authors use the following harmonic function as potential functions

$$\phi = \frac{\lambda}{2\pi} log r \tag{24}$$

where λ is the strength of the source or the sink, and r is the distance from obstacle i to the potential at any point (x, y). In Connolly et al. (1990), Shi et al. (2007) and Panati et al. (2015), the authors use the same harmonic function defined in $\Omega \subset R^n$ which satisfies Laplace equation:

$$\nabla^2\phi = \sum_{i=1}^{n} \frac{\partial^2\phi}{\partial x_i^2} = 0 \tag{25}$$

where x_i is the ith Cartesian coordinate and n is the dimensionality; for planning path, a 2-dimensional structure as shown below is used

$$\frac{\partial^2\phi}{\partial x_i^2} + \frac{\partial^2\phi}{\partial y_i^2} = 0. \tag{26}$$

The solution to Laplace's equation needs to check the boundary of Ω which includes path, start to target, and the boundary of all obstacles. Laplace's equation is solved by using finite-difference method. In Daily and Bevly (2008), the authors extend the result of Connolly et al. (1990), Shi et al. (2007) and Panati et al. (2015) to arbitrarily shaped obstacles in analytical fields, where multiple circles are used to cover obstacles of arbitrary shape. It should be noted that the harmonic functions satisfy the min-max principle and thus the problem of local minima as shown in the original potential function approach is eliminated.

4.4.4. Improved potential field approach

In the original potential field approach, apart from local minima, there exists an additional problem, that is goals nonreachable with obstacles nearby (GNRON). This occurs when an obstacle is close to a target. In this situation, when the vehicle goes to that target, at the same time, it also goes to that obstacle. In most design patterns, the repulsive force from the obstacle will be larger than the force from the target. Therefore, the vehicle may not arrive at its target. In Ge and Cui (2000), the authors try to solve this problem by scaling the repulsive potential function for path planning, that is

$$U_{rep} = \begin{cases} \frac{1}{2}\left(\frac{1}{\rho(x, x_{obs})} - \frac{1}{\rho_0}\right)^2 \rho^n(x, x_d), & if\ \rho(x, x_{obs}) \leq \rho_0 \\ 0, & if\ \rho(x, x_{obs}) > \rho_0 \end{cases} \tag{27}$$

where $\rho(x, x_{obs})$ is the distance between the vehicle x and the obstacle, $\rho(x, x_d)$ is the distance between the vehicle and the desired target x_d, ρ_0 is the distance of influence of the obstacle, and n is a positive constant. Therefore, the vehicle can reach its global minimum point by properly tuning n. However, the result of Ge and Cui (2000) is only available for the stationary target. For dealing with dynamic target, the authors in Tingbin and Qisong (2013) revise the gravitational field (attractive potential function) as

$$U_{att} = \alpha_x||x_d - x||^m + \alpha_v||v_d - v||^n \tag{28}$$

where α_x and α_v are proportion coefficients to position and velocity differences, respectively, v and v_d are the velocity and desired velocity, respectively, and m and n are positive constants. Thus, the vehicle following the negative gradient direction from the revised potential function, can avoid obstacles to seize mobile target. However, these existing results for path planning are based on a known environment and cannot drive vehicle moving when the environment is unknown. In Fakoor, Kosari, and Jafarzadeh (2015), the authors use fuzzy logic to design potential functions and the moving direction of the vehicle is obtained from fuzzy artificial potential field. Such a design can cope with unknown environment and avoid the obstacles in the unknown environment. However, the fuzzy logic approach is still a nonlinear function. This implies that the proposed approach cannot solve the local minima issue as shown in the original potential function.

4.4.5. Navigation function

The improved potential function has provided one solution to GNRON or track the moving target problems. However, the local minima is still not solved completely. In Rimon and Koditschek (1992), the authors suggest to construct a navigation function for exact vehicle motion planning to solve this issue. In Roelofsen et al. (2015) and Valbuena and Tanner (2012), the authors examine the effectiveness of the navigation function approach using AscTec quadrotors and mobile robots. Moreover, in Rahmani, Kosuge, Tsukamaki, and Mesbahi (2008), the authors present a deconfliction algorithm for the multi-vehicle system by constructing an appropriate navigation function. The algorithm is based on a parameterized navigation function given by

$$V_i = \frac{\gamma_i(x)}{(\gamma_i^k(x) + \beta_i(x))^{1/k}} \tag{29}$$

where γ_i, β_i are appropriate functions, and k is parameter. The function γ_i is often given by

$$\gamma_i(x) = ||x_i - x_{di}||^2 \tag{30}$$

where x is all the team state of the multi-vehicle system. The function $\beta_i(x)$ is given by

$$\beta_i(x) = \prod_{x_j \in O_{ij}} \beta(r_{ij}) \tag{31}$$

where O_{ij} is the jth protected zone for the ith vehicle, $r_{ij} = (||x_j - x_i|| - \rho_j)/\delta_j$, and $beta(r_{ij})$ is a continuous function described by

$$\beta(r_{ij}) = \begin{cases} 0 & r_{ij} < 0 \\ f(r_{ij}) & 0 \leq r_{ij} < 1 \\ 1 & r_{ij} \geq 1 \end{cases} \tag{32}$$

where f is a high-order polynomial. By constructing a navigation function as a potential, the gradient-following routine of the function and the local minima is solved. Unfortunately,guaranteeing safe separation distance between vehicle and obstacle or vehicle to another vehicle is not proven for the multi-vehicle system.

4.5. Geometric guidance

As stated above, the path planning approach is mainly used to static obstacle situation, while most confliction resolution algorithms guarantee avoidance without limiting control. In fact, most

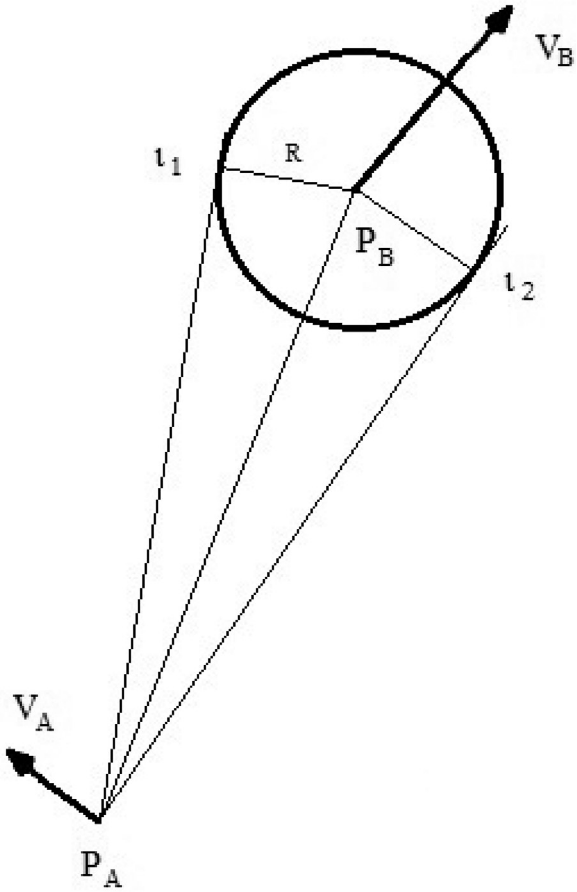

Fig. 7. Collision cone.

existing potential function approaches also assume that the vehicle must have a capability of infinite control for guaranteeing avoidance. This is unreasonable for a real vehicle control and limits to practical applications. In addition, most existing approaches need a heavy computational cost and hence they are not suitable for reactive collision avoidance of the multi-vehicle system. Over the past two decades, the geometric collision avoidance algorithms have been developed and are potentially best suitable for collision avoidance in the multi-vehicle system (Jenie, Kampen, Visser, Ellerbroek, & Hoekstra, 2015b), for it is not necessary to do extensive predictions and analysis. Basically, there are two approaches in geometric guidance: one is collision cone approach, and the other one is velocity obstacle.

4.5.1. Collision cone

The concept of the collision cone (CC) comes from the work of Chakravarthy and Ghose (1998). In Fig. 7, the vehicle is represented by a point P_A, the obstacle P_B is a circle with radius R, and two tangent lines are l_1, l_2. The collision cone is consisted of $l_1P_Al_2$, which represents the set of vehicle velocities that will cause a collision with the obstacle if each of both the vehicle and obstacle is constant. In Watanabe, Calise, and Johnson (2006), based the concept of the collision cone, the authors design a minimum effort guidance control by minimizing an acceleration cost function. However, this paper has no constraints on velocity and this may violate the safety ball before the vehicle reaches the target point. In Mujumdar and Padhi (2011), the authors present a nonlinear geometric guidance approach for avoiding the obstacle, where the guidance algorithms give the velocity command along the tangent line over collision cone. This work only gives the simulation to the two vehicle case but its effect on safety for many vehicle cases is not clear. In Lalish (2009), the authors use the collision cone concept to tune the vehicle's heading until the direction is outside the collision cone. In Anderson (2011), the author extends the approach suggested in Lalish (2009) to the 3D case. A simple 3D cone is used as a collision cone and the designed vehicle's heading should be outside this cone avoiding collision.

4.5.2. Velocity obstacle

This idea is first introduced in Fiorini and Shiller (1998). The velocity obstacle as shown in Fig. 8, is defined in a velocity space by moving the collision cone by the obstacle velocity V_B. The collision cone is selected for the vehicle A, which avoid a future collision with an obstacle. In Kuwata, Wolf, Zarzhitsky, and Huntsberger (2011), the authors incorporate the ssing on the rightrule into the velocity obstacle algorithm to further improve the safety. The algorithm is based on the rules of the sea, the International Regulations for Preventing Collisions at Sea (COLREGS). In Jenie et al. (2015b), the authors extend the results of Fiorini and Shiller (1998) and Kuwata et al. (2011) to develop a novel collision avoidance algorithm, where the maneuver uses three modes: turn, maintain and mission. In this work, the authors try to get the maneuver while satisfying the minimum avoidance turning rate. Unfortunately, the simulation is tested only for two vehicles. Further analysis is necessary to examine if it is valid for multi-vehicle cases.

It is found in van der, Lin, and Manocha (2008) that the undesirable oscillation behavior is observed in the original VO approach during the vehicle motion while keeping the safety. To overcome this phenomenon, the authors in van der Berg et al. (2008) suggest to move the collision cone by $\frac{1}{2}(V_A + V_B)$ replacing V_B as in the original velocity obstacle approach. This algorithm is called the reciprocal velocity obstacle (RVO). Furthermore, in Allawi and Abdalla (2014), the authors apply the genetic algorithm to RVO to find the optimal velocity command. However, it is still observed that the undesirable oscillation exists in the RVO approach when more than two vehicles are tested.

The authors in Snape et al. (2011) present a hybrid reciprocal velocity obstacle (HRVO) approach to improve RVO, where the velocity obstacle of RVO is taken as an asymmetrical shape as shown in Fig. 9. If v_A lies in the right half side of the center line of the RVO collision cone, the vehicle A selects a velocity which is close to the right side of the center line of RVO; in this situation, the left side of RVO is enlarged by the edge of VO. If v_A is in the left half-side of the RVO cone, the opposite cone shape is formed.

The work of HRVO does not discuss smoothness and collision avoidance by giving any formal proofs. The optimal reciprocal collision avoidance (ORCA) approach suggested by van den Berg et al. (2011) and Alonso-Mora, Breitenmoser, Beardsley, and Siegwart (2012) gives a formal proof regarding smoothness and collision avoidance for arbitrary number of vehicles. In this work, the authors introduce a time τ, that is at least τ time for guaranteeing the collision-free for both vehicles A and B. Thus, the velocity obstacle is truncated as shown in Fig. 10. The avoiding velocity is given by finding the following minimum cost

$$v_{new} = \mathrm{argmin}_{v \in ORCA\ cone} ||v - v_{des}|| \tag{33}$$

where v_{des} is the desired velocity. Moreover, the authors in (van den, Wilkie, Guy, Niethammer, & Manocha, 2012) and Bareiss and van den Berg (2013) consider the vehicle model as a linear dynamics and design a linear quadratic regulator (LQR) feedback controller. This controller combines with the velocity obstacle approach to generate a novel collision avoidance algorithm. The basic idea of this approach is to predict a closed-loop dynamic equation for a small time step Δt and guarantee the collision-free during this interval. It is similar to the work of ORCA, but

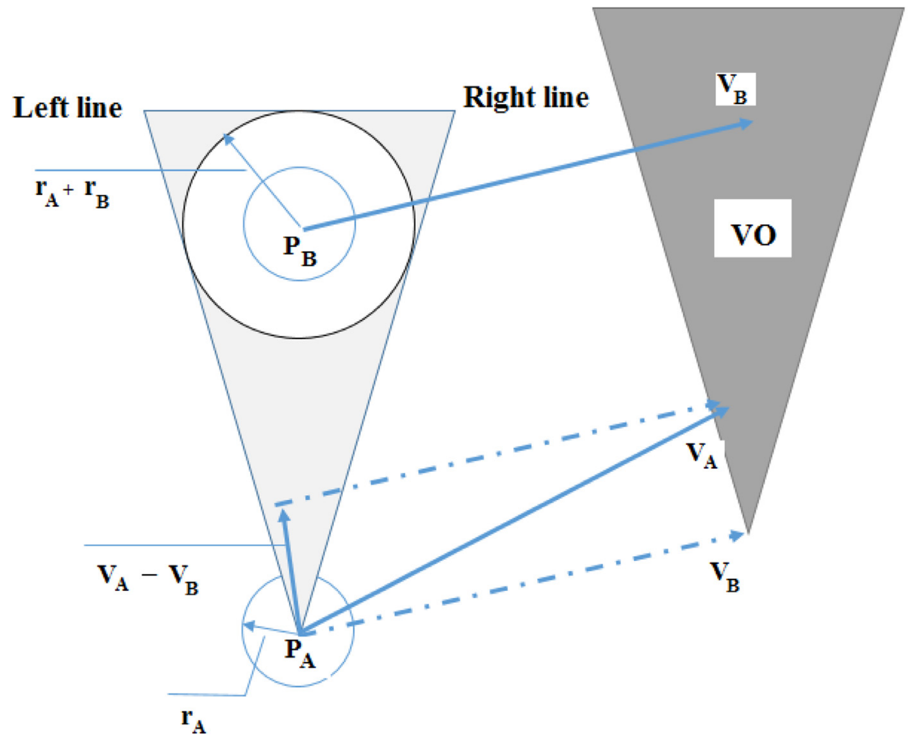

Fig. 8. Velocity obstacles concept. Assume that we have two vehicles A and B with the respective radius r_A and r_B at the respective positions p_A and p_B. The velocity vectors of vehicles A and B are v_A and v_B, respectively. The light collision cone stands for the velocity obstacle of vehicle A if vehicle B were stationary. The dark cone denoted as $VO_B^A(v_B)$ stands for the same cone form of velocity obstacle when considering vehicle B's velocity v_B.

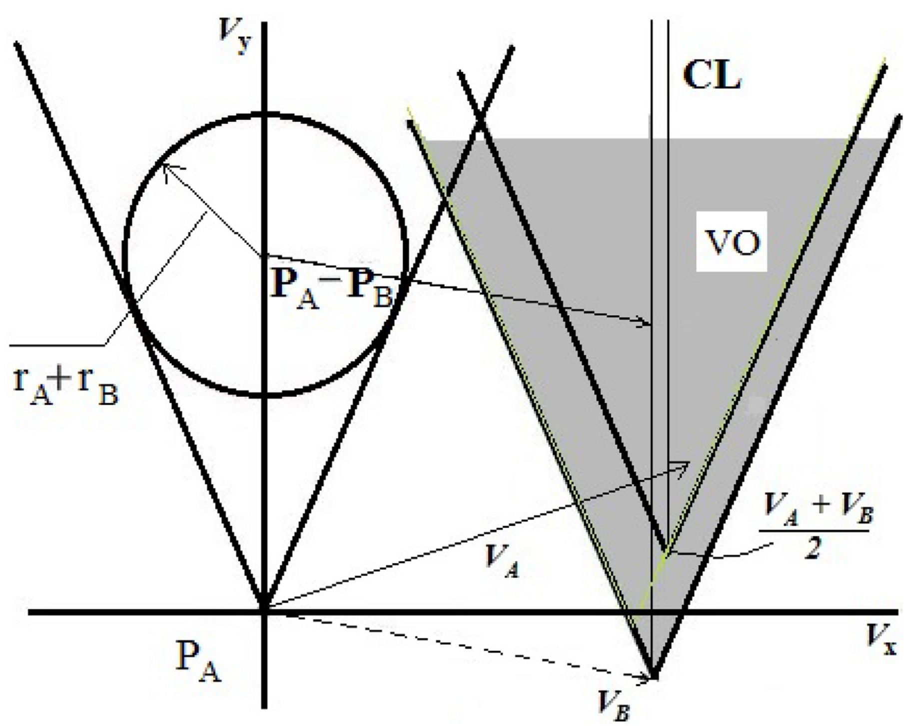

Fig. 9. Hybrid reciprocal velocity obstacle (HRVO) (Snape et al., 2011). Assume that we have two vehicles A and B with the respective radius r_A and r_B at the respective positions p_A and p_B. The velocity vectors of vehicles A and B are v_A and v_B, respectively. The dark collision cone denoted as $VO_B^A(v_B)$ stands for the same cone form when considering vehicle B's velocity v_B. The cone covered by green line represents the HRVO cone when v_A is in the right side of the center line (CL) of the RVO cone. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

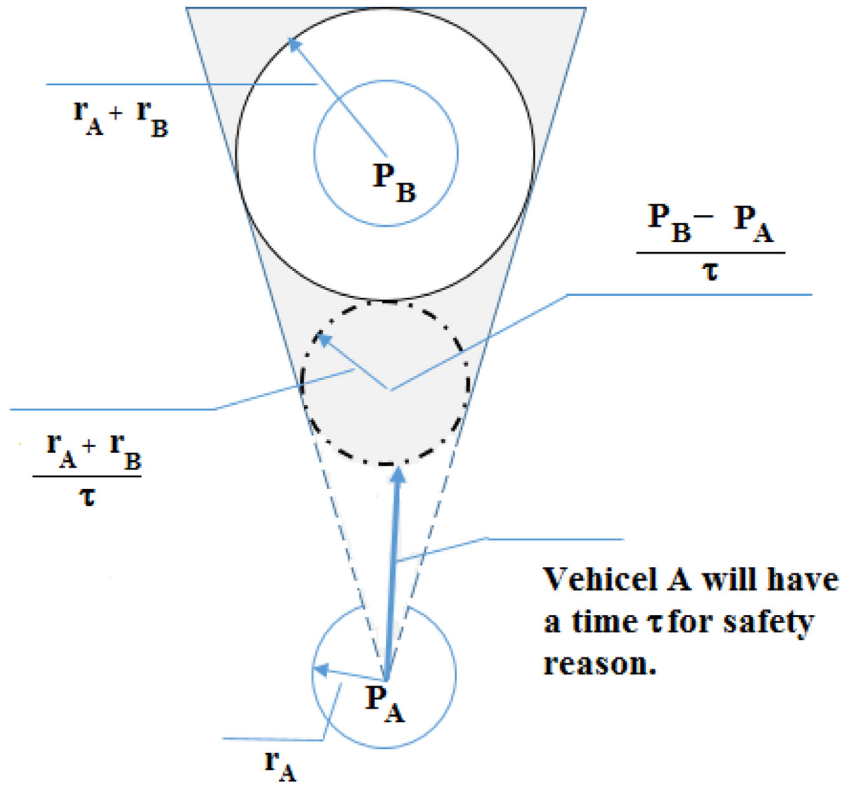

Fig. 10. Optimal collision reciprocal avoidance (ORCA) (van den Berg et al., 2011). Assume that we have two vehicles A and B with the respective radius r_A and r_B at the respective positions p_A and p_B. The velocities of vehicles A and B are v_A and v_B, respectively. The grey collision cone stands for the ORCA cone.

the feedback control is optimal based on a quadratic cost function. Considering the sensing uncertainty, in Hennes, Claes, Tuyls, and Meeussen (2012), an adaptive Monte-Carlo localization is used to estimate the position of the vehicle and ORCA approach uses the estimated information to select the velocity. In Jenie et al. (2014), the authors also consider the sensing uncertainty and assume that the velocity direction is known. But the velocity is uncertain and its range is known. Thus, a velocity obstacle set is obtained. The collision avoidance problem is to avoid this velocity obstacle set. The velocity obstacle approach presented can be extended to handle with uncertainties in shape and velocity of the obstacles, this is called the probabilistic velocity obstacles (Kluge & Prassler, 2006). This approach allows to reflect the limitations of real sensors and object tracking techniques.

On the other hand, design and implementation of velocity obstacle algorithm using a real team of multi-UAVs are a challenging work. In Huang, Teo, Liu, and Dymkou (2017b), the VO algorithm with the passing on the right rule is applied to eight UAVs for real flight test. The results verify that the VO algorithm can avoid both moving and static obstacles. In Huang, Teo, and Liu (2019), the VO algorithm is tested in a Hardware-in-the-loop simulation, where the eight UAVs go through a four-way intersection. In the simulation, we want to see the collision avoidance for crossing the UAVs and the result shows that the VO algorithm can guarantee the safety.

The above-mentioned approaches can be used in 2-dimensional plane. In 3-D case, these results no longer hold and it is necessary to develop new ideas for achieving the avoidance. The foundation of the 3D VO approach is introduced in Chakravarthy and Ghose (2012) and the authors extend their collision cone concept (Chakravarthy & Ghose, 1998) from 2-dimensional plane to 3-dimensional space. However, they only give some theoretical analysis and no simulation is given. In Zhang, Yang, and Zhou (2014), the authors consider the horizontal plane Oxy and verical plane Orz for checking the collision using the VO approach and 3-D feasible trajectories are estimated by using quaternion polynomial for a future time. However, this result considers only two planes and reduces the design freedom for avoiding the obstacles. In Alonso-Mora, Naegeli, Siegwart, and Beardsley (2015), the authors use an optimization algorithm to obtain a feasible solution subject to linear constraint which is a predefined side of the VO cone when vehicles are moving towards each other. However, the collision considers only the predefined side which limits the finding of the feasible trajectories in a 3D space. An elegant 3D velocity obstacle approach is developed in Jenie et al. (2016). The key point in this approach is to design avoidance planes which are the rotation of a plane around the vehicle $X-axis$. Each avoidance plane intersects with a 3D VO cone and form a 2-D cross-sectional shape which is collision area as shown in Fig. 11. The reachable velocity can be selected by finding the intersection points between the circle of the current velocity and 2-D cross-sectional area. Using this approach, the 3-D moving obstacles can be handled, but the static obstacle such as cylinder and cube etc., cannot be handled directly. The work in Tan et al. (2018) extends the result of Jenie et al. (2016) and develops a velocity obstacle algorithm against 3-D cylinder obstacles. In Conroy et al. (2014), the authors use two real Parrot Drone quadrotors to examine ORCA approach in a 3-D space and show that ORCA can avoid the moving obstacle well. However, this experiment does not use ORCA to test the case where there are many UAVs (>10) flying in 3D space.

4.6. Motion planning of teams of multi-UAVs

The geometric guidance is a reactive control strategy. The main advantage of this scheme is that it has a low computational load for multi-UAV control. However, as indicated in

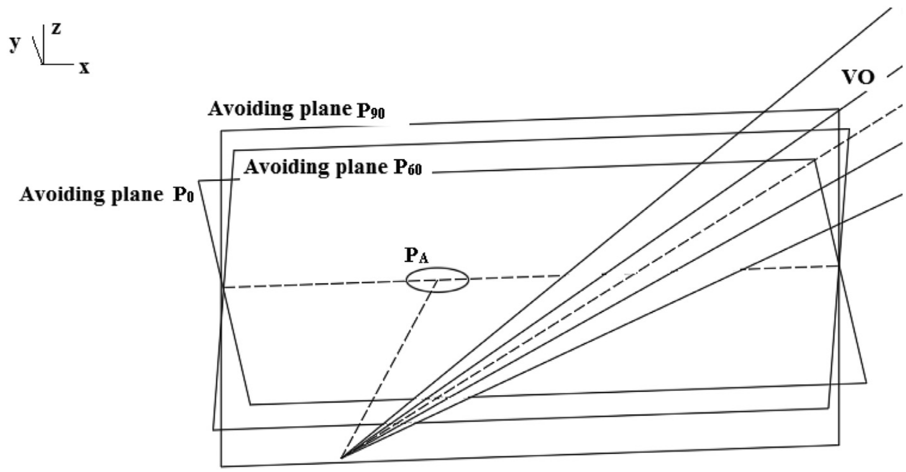

Fig. 11. Avoidance planes intersecting the 3D VO Jenie et al. (2016).

Pallottino et al. (2007b), it is a local planning and may lead a deadlock, where more UAVs block each other in a way such that none of them is able to continue going to their corresponding goals. This situation becomes worse when multi-UAVs perform their mission in a crowded environment.

Many approaches have been reported to solve this issue. Several widely used schemes include conflict-based search algorithms, multi-UAV trajectory planning of large UAV teams, prioritized planning, and formal verification.

4.6.1. Conflict-based search for multi-agent systems

Conflict-Based Search(CBS) is developed by Sharon, Stern, Felner, and Sturtevant (2015). It is a two-level search-based multi-agent path finding (MAPF) algorithm. The MAPF problem is specified by an undirected graph $G = (V, E)$ whose vertices $v \in V$ correspond to locations agents can occupy and edges $e \in E$ correspond to straight-line trajectories the agents traverse when moving from one vertex to the other one. A set of k agents $\{a_1, a_2, \ldots a_k\}$ is given, where a_i has start vertex $s_i \in V$ and goal vertex $g_i \in V$. Time is discretized into time steps and agent a_i is in s_i at time t_0. Between successive time steps $[t_i, t_{i+1}]$, every agent can either move to an adjacent vertex or wait at its current vertex. A path of a_i is a sequence of move and wait actions that lead a_i from s_i to g_i. A solution is referred to a set of k paths for the given set of k agents. A conflict between two paths is a tuple $< a_i, a_j, v, t >$ where agent a_i and agent a_j are planned to occupy vertex v at time t. The objective is to find a conflict-free solution.

In CBS, agents are associated with constraints. A constraint for agent a_i is a tuple $< a_i, v, t >$ where agent a_i is prohibited from occupying vertex v at time step t. CBS runs both high and low levels of search algorithms. The low-level search in CBS is to find an optimal plan for each individual agent given constraints provided by the high-level. The high-level search in CBS works on a constraint tree, which is a binary tree, in which each node represents a set of CBS constraints imposed on the agents and a joint plan consistent with these CBS constraints.

There are two versions of CBS: one is the discrete-time CBS (Sharon et al., 2015), and the other is the continuous-time CBS (Andreychuk, Yakovlev, Atzmon, & Stern, 2019). A number of the improved CBS versions have been reported (see Cao et al., 2019; Cohen et al., 2018; Hönig, Kiesel, Tinka, Durham, & Ayanian, 2018; Li et al., 2019). A survey in Felner et al. (2018) summarizes a line of work on CBS algorithms. Most of the developed conflict-based search algorithms emphasize algorithm development independent of real applications and some can be used in many UAV applications. This is the reason why CBS is briefly described here. However, it still has the gap between the algorithm and the real-world in some applications of multi-UAVs. In the next section, we introduce multi-UAV trajectory planning schemes which are practical in multi-UAV system, even though some works have similar idea as in the CBS algorithm.

4.6.2. Multi-UAV trajectory planning of large UAV teams

Multi-UAV trajectory planning is stated as: Given a set of UAVs with starting points and their destinations, the task is to find a set of globally coordinated collision-free trajectories. This topic has been studied extensively. Most researches focus on the trajectory generation on single UAV. The literature discussed in this section concentrate on multi-UAV trajectory planning. Some works of the literature borrow the CBS to generate trajectories of multi-UAVs in large teams of UAVs. In Augugliaro, Schoellig, and D'Andrea (2012), the authors present a collision-free trajectory planning approach in 3D space for multi-UAVs. The problem is treated as a sequential convex programming that converts non-convex constraints into convex ones. The programming is solved iteratively. However, it is performed in a centralized form. For large teams of UAVs, the optimization for generating all UAVs' trajectories is inefficient. In Cole and Wickenheiser (2018), the authors propose a reactive trajectory planning approach. It considers unknown environments with wind disturbances and uses UAV on-board sensors and communication with other UAVs within a finite region to compute heading changes to avoid obstacles by a prescribed distance. To guarantee smoothness of the trajectory, it uses a continuous fourth derivative algorithm to generate a collision-free trajectory. However, the proposed approach using only local information is suitable for smaller teams and does not scale up to the large size of UAV teams. In Hönig, Preiss, Kumar, Sukhatme, and Ayanian (2018), the authors present a formation-change trajectory planning approach for large UAV teams. The proposed approach uses two stages: discrete planning and continuous refinement. Discrete planning calculates a sequence of waypoints for each UAV on an approximated graph, while satisfying collision constraints. This planner is expressed as an integer linear programming problem. Based on the discrete plan, continuous refinement converts the waypoints obtained into a set of smooth trajectories which are represented by a polynomial Bezier basis within a short time interval, where the polynomial coefficients are obtained from solving a quadratic programming. Since the smooth trajectories are local solution, the authors further develop an iterative refinement scheme to improve the trajectory planning. This approach is demonstrated on 32 quadrotors navigating in an indoor environment. However, this approach is not robust enough for other issues, for example, persistence problem. In Hönig, Kiesel, Tinka, W. Durham, and Ayanian (2019), the authors use an action dependency graph (ADG) to find the action-precedence relationships of multi-UAV path solution. Thus, the proposed approach can execute multi-UAV plans on actual UAVs persistently and robustly.

4.6.3. Prioritized planning

The idea of prioritized planning is first introduced by Erdmann and Lozano-Pérez (1987). In a prioritized planning, each UAV is given a priority according to its operation. According to the priority assignment, each UAV provides a solution of the coordinated trajectories, which can avoid the collision and deadlock. The main benefit of this approach is that it is fast in a complex environment, especifially if large teams of multi-UAVs are considered. A easy way to implement prioritized planning is to design a centralized algorithm, where all UAVs can broadcast their objectives to a centralized planner (or a UAV) which finds a solution for each UAV. In Silver (2005), the author presents a cooperative A^* based on the prioritized planning scheme and finds a joint path for all the UAVs. In Zhang, Wang, Fu, and Su (2018), the authors present a prioritized planning with an improved ant algorithm to find a solution for all UAVs, where the priority of each UAV is assigned according to its remained battery charge. Such algorithms rely on point-to-point communication network connectivity for all time. While the size of the joint space, for example, the number of UAVs, is increased significantly, maintaining network connectivity can restrict the UAVs to perform their missions. Therefore, a preferred solution is to allow each UAV to communicate with its neigbouring UAVs and give a distributed solution. A large body of work has been reported on the distributed prioritized planning schemes (Čáp, Novák, Kleiner, & Selecký, 2015; Čáp, Novák, Selecký, Faigl, & Vokffnek, 2013; Ma, Jiao, Wang, & Panagou, 2018; Velagapudi, Sycara, & Scerri, 2010). In Velagapudi et al. (2010), the authors present a decentralized algorithm of the prioritized planning approach, where UAVs execute the algorithm in a synchronized iteration. In every iteration, each UAV checks if its current path is consistent with paths of higher priority UAVs. If yes, then the UAV keeps its current motion; otherwise, it needs to find a new path. However, it is difficult to implement the synchronized prioritized planning algorithm due to imperfect networks. To overcome this drawback, in Čáp et al. (2015, 2013), the authors propose an asynchronous decentralized algorithm of the prioritized planning scheme, where each UAV only handles the incoming messages and replaces the corresponding paths stored. The UAV checks if its current path is consistent with the new contents of its paths stored. If yes, then the UAV keeps its current path; otherwise, it needs to trigger replanning and broadcasts its new path to other UAVs. Furthermore, a 3D decentralized prioritized planning algorithm for multi-UAVs moving among 3D obstacles is proposed (Ma et al., 2018). It is a two-level control, where a prioritized A^* path planner at high level provides waypoints, while a barrier function-based control provides a safe direction of motion for each UAV. The proposed approach is not required to have a fully connected network, but the prioritized A^* path planner is triggered whenever UAVs are connected. However, these approaches needs to trigger replanning every time when an unexpected event affects the motion of one UAV. In Gregoire, Cap, and Frazzoli (2018), based on homotopy theory, the authors design a motion coordination scheme. In this approach, it is not necessary to do replanning even when an exogenous disturbance enters the team of UAVs. This approach requires to have initial coordinated trajectories which can be obtained from any motion planner, and then they are used to construct a maximal set of homotopic solutions which includes all paths in a collision-free and deadlock-free coordination space. A robust trajectory tracking control law is also proposed, guaranteeing that the trajectory in the coordination space remains in the homotopy class which can be encoded by priority graphs as proved in Gregoire (2014).

4.6.4. Formal verification

A major difficulty for path planning is to integrate task planning with continuous motion planning (Plaku & Karaman, 2015). One way to solve this is to use formal verification techniques which can be helpful in proving the correctness of systems. The verification of these systems is done by a formal proof on an abstract mathematical model of the system. These mathematical model systems include finite state machines, labelled transition systems, Petri nets, timed automata and temporal logic etc. For example, in Wan et al. (2018), the authors use a Petri Net to design a control approach to solve the collision problem in multi-vehicles; a labeled Petri Net is proposed to represent the unpredictable events by a set of transitions; the whole net is partitioned into a set of forbidden zones; the proposed approach computes the consistent marking set for a given sequence, and use it to design a collision-free action.

A typical representation of formal verification is linear temporal logic (LTL). LTL was first introduced by Pneuli Pnueli (1977). It is used for reasoning about computer programs. Later, Lamport (1994) developed LTL theory whose principle is based on the action-as-relation rule. For a system, it can be expressed by LTL in the form of Boolean and temporal operators. The Boolean logic is a form of algebra, including $\neg$(negation), $\vee$ (disjunction), $\wedge$ (conjunction), and $\rightarrow$(implication), while the temporal operators use, X (next), U(until), and R (release). LTL describes an infinite sequence of states, where each state has a unique transition to its time successor.

The problem of multi-UAV motion planning from LTL specifications can be stated as: Given the UAV model and initial state, an obstacle-free workspace, and a task specification as an LTL formula φ, the objective of LTL motion planning is to design collision-free motion trajectories for all UAVs that satisfy φ. One approach on LTL motion planning is to design the high-level planner by using a mixed integer linear programming (MILP) with LTL specifications (Karaman & Frazzoli, 2011; Wolff, Topcu, & Murray, 2014). Although this approach can handle both the discrete and continuous constraints, it is not practical when having more obstacles or increasing UAV number. An efficient approach for multi-UAV motion planning from LTL specifications is based on the coordination of Satisfiability Modulo Convex (SMC) programming (Saha, Ramaithitima, Kumar, Pappas, & Seshia, 2014; Shoukry et al., 2017), where the problem is decomposed into small sub-problems using SMC. Therefore, multi-UAV motion planning can be efficiently solved. Alternative approach is to use a hierarchical framework for planning of arbitrarily large teams of UAVs with obstacles (Kloetzer & Belta, 2007). In this approach, collision avoidance is expressed by LTL specifications φ and the paths that satisfy φ can be obtained. However, these approaches are only applicable for the systems where UAVs can change their paths. For UAVs which have fixed paths, a distributed algorithm to give the solution of collision and deadlock avoidance, is proposed in Zhou, Hu, Liu, and Ding (2017). It is assumed that each path is static obstacle-free. The motion of each UAV is modeled as a labeled transition system which is a primary form of LTL. The proposed algorithm is executed by repeatedly stopping and resuming UAVs whose next movement can result in collisions or deadlocks. However, this algorithm cannot work in an environment with moving obstacles such as humans on ground or moving balloons in a sky. In Alonso-Mora et al. (2018), the authors develop an integration of a high-level mission planner with a local planner in 3D spaces, guaranteeing collision and deadlock avoidance, where a moving obstacle blocks the UAV temporally. The high-level planning is an off-line algorithm for synthesizing the mission which is implemented in a centralized form, where LTL is used to design an automaton, while an on-line planning executes the automaton, where the optimization is used to obtain a collision-free velocity reference. Specifically, the proposed approach can handle walking object moving in an environment.

Multi-UAV motion planning considers multi-UAVs as a system and designs paths for all UAVs. Several different approaches have been proposed for avoiding collision. Specifically, these approaches can solve the deadlock issue such that the multi-UAVs can reach their goals or destinations successfully.

5. Summary and discussion

In this section, we first summarize the collision avoidance approaches considered in this paper. Table 1 is given to show the basic information about every approach: the approach, velocity constraints, robustness, both static and dynamical obstacles, disturbance, 3D space, deadlock, and large teams of UAVs.

When applying a collision avoidance approach, it needs to check if its control signal is reasonable for any real vehicle. This implies that an infinite control signal is not acceptable. In the table, it is observed that some potential function and conflict resolution approaches have this limitation. In most existing geometric guidance approaches, this term is satisfied, since in general, the reachable velocity set is always selected within an allowable range. This is an distinct advantage of the geometric guidance approach, especially for the velocity obstacle approaches.

We also need to consider how your approach is robust against unpredictable disturbances. If the planner without such robust function, multi-UAV motions will be affected seriously by an unpredictable event. For example, when UAVs have created motion

Table 1
Summary of collision avoidance approaches:VC–velocity constraint;R–Robustness;SDO–static and moving obstacles; 3DS–3 dimensional space;LT–Large Teams of UAV .

Approach	Literature	VC	R	SDO	3DS	Deadlock	LT
Path selection	Nikolos et al. (2003)	Yes	No	No	Yes	No	No
	Yang et al. (2013)	Yes	No	No	Yes	No	No
	Bellingham et al. (2003)	Yes	No	No	No	No	No
Graph path selection	Lozano-Pérez and Wesley (1979)	Yes	No	No	Yes	No	No
	El. Khaili (2014)	Yes	No	Yes	Yes	No	No
Conflict resolution	Pallottino et al. (2007b); Stanley (2005)	Yes	No	Yes	Yes	No	No
	Liu and Hwang (2014); Tomlin et al. (1998)	No	No	Yes	No	No	No
	Borrelli et al. (2006)	Yes	No	Yes	Yes	No	No
	Delsart et al. (2009)	Yes	No	No	No	No	No
	Gerdts et al. (2012)	Yes	No	No	Yes	No	No
	Matsuno and Tsuchiya (2014a,b)	Yes	NO	Yes	No	No	No
	Hwang and Tomlin (2002); Stroe and Andrei (2017)	No	No	Yes	No	No	No
MPC	Richards et al. (2003); Richards and How (2003)	Yes	Yes	No	Yes	No	No
	Alrifaee et al. (2016)	Yes	Yes	Yes	No	No	No
	Carvalho et al. (2013); Chao et al. (2012); Nikou et al. (2017)	Yes	Yes	No	Yes	No	No
	Hang et al. (2020)	Yes	Yes	Yes	No	No	No
	Zhu and Alonso-Mora (2019)	Yes	Yes	Yes	Yes	No	Yes
Potential function	Borenstein and Koren (1989); Koren and Borenstein (1991)	No	Yes	Yes	No	No	Yes
	Arambula Cosio and Padilla Castaẽda (2004)	Yes	Yes	No	No	No	Yes
	Connolly et al. (1990); Daily and Bevly (2008); Kim and Khosla (1992); Panati et al. (2015); Shi et al. (2007)	No	Yes	Yes	No	No	Yes
	Ge and Cui (2000)	No	Yes	No	No	No	Yes
	Tingbin and Qisong (2013)	No	Yes	Yes	No	No	Yes
	Fakoor et al. (2015)	Yes	Yes	Yes	No	No	Yes
	Rahmani et al. (2008); Rimon and Koditschek (1992); Roelofsen et al. (2015); Valbuena and Tanner (2012)	No	Yes	Yes	No	No	Yes
Geometric guidance	Fiorini and Shiller (1998); Jenie et al. (2015b)	Yes	Yes	Yes	No	No	Yes
	Anderson (2011); Lalish (2009)	Yes	Yes	No	No	No	Yes
	van der Berg et al. (2008)	Yes	Yes	Yes	No	No	Yes
	Allawi and Abdalla (2014)	Yes	Yes	Yes	No	No	Yes
	Alonso-Mora et al. (2012); Conroy et al. (2014); Snape et al. (2011); van den Berg et al. (2011)	Yes	Yes	Yes	No	No	Yes
	Jenie et al. (2014)	Yes	Yes	No	No	No	Yes
	Bareiss and van den Berg (2013); van den Berg et al. (2012)	No	Yes	Yes	Yes	No	Yes
	Kluge and Prassler (2006)	Yes	Yes	No	Yes	No	Yes
	Alonso-Mora et al. (2015); Zhang et al. (2014)	Yes	Yes	Yes	Yes	No	Yes
	Jenie et al. (2016); Tan et al. (2018)	Yes	Yes	Yes	Yes	No	Yes
Motion planning of teams of multi-UAVs	Andreychuk et al. (2019); Sharon et al. (2015)	No	Yes	Yes	Yes	No	No
	Cohen et al. (2018); Hönig, Kiesel et al. (2018); Li et al. (2019)	No	Yes	Yes	Yes	Yes	No
	Augugliaro et al. (2012)	Yes	No	Yes	Yes	Yes	No
	Cole and Wickenheiser (2018)	Yes	Yes	Yes	Yes	No	Yes
	Hönig, Preiss et al. (2018)	No	No	Yes	Yes	Yes	Yes
	Hönig et al. (2019)	No	Yes	Yes	No	Yes	Yes
	Silver (2005)	No	No	Yes	No	Yes	No
	Zhang et al. (2018)	No	No	Yes	No	Yes	No
	Ma et al. (2018)	Yes	No	Yes	Yes	Yes	No
	Velagapudi et al. (2010)	No	No	Yes	No	Yes	Yes
	Čáp et al. (2015, 2013)	No	Yes	Yes	No	Yes	Yes
	Gregoire (2014); Gregoire et al. (2018)	No	Yes	Yes	Yes	Yes	Yes
	Karaman and Frazzoli (2011); Saha et al. (2014); Wolff et al. (2014)	Yes	Yes	Yes	Yes	Yes	No
	Alonso-Mora et al. (2018); Shoukry et al. (2017)	Yes	Yes	Yes	Yes	Yes	Yes
	Kloetzer and Belta (2007)	Yes	Yes	Yes	Yes	Yes	No
	Zhou et al. (2017)	Yes	Yes	No	Yes	Yes	Yes
	Wan et al. (2018)	No	No	Yes	Yes	Yes	No

plans for a team of UAVs, even one UAV is blocked by an unpredictable balloon in a sky, all the other UAVs may be stopped. From the table, most existing approaches have no such function.

Subsequently, we consider if the proposed approach can handle both static and dynamical obstacles. For a real world, this is a basic requirement, for we face with various obstacles including static or dynamical obstacles. Fortunately, most existing approaches except of path planning approaches satisfy this term.

Next, we are concerning about the 3D issue. As multi-vehicle autonomous systems are required to support a potential move to free flight in sky, the collision avoidance should guarantee the safety in a 3-D space. From the table, it is observed that most existing approaches cannot satisfy this term.

After that, we check the term “Deadlock”. This is to see if an algorithm can continue to achieve its mission without a collision when a UAV is blocked temporally by a dynamic obstacle which includes both the moving UAV and obstacles (for example, hot air balloons are the moving obstacles). This is an important requirement for large teams of UAVs. Unfortunately, most existing approaches cannot satisfy this term.

Finally, we check if the proposed approach is suitable for large teams of UAVs. This is because swarm UAVs are becoming a hot point in the research of coordinating multiple UAVs as a system. It is observed that some works can be applied to large teams of UAVs.

From Section III and Table 1, it is observed that the formal verification approach shows its advantages compared with other approaches. The formal verification approach has the following features:

- The deadlock issue can be avoided;
- The approach is robust against unpredictable events;
- It is also suitable for large teams of UAVs if incorporating with reactive control scheme.

6. Challenges of collision avoidance

It is forseen that in the near future, there will be a demand for large numbers of UAVs to cooperatively perform complex tasks. Collision avoidance needs to be developed for achieving and performing such tasks. Though the survey above has listed many approaches that have been proposed and studied, there are still some challenges in collision avoidance control that are not addressed in the previous sections. Here, we summarize the main challenges that we have found in this area.

- Avoiding conditions: The UAV control should analyze the conditions that ensure the avoidance against obstacles, especially in dealing with the uncertainty. This is because sensors (GPS or LiDAR) always have position and velocity errors. Although several literature have given collision avoidance conditions, the robust analysis is not discussed. It is important to consider the effects of the sensor errors in collision avoidance.
- Motion planning in uncertain environments: Most existing motion planning techniques for multi-UAVs assume that the accurate knowledge of the environment is known. This is not true in a practical situation due to the changing environment. Motion planning must be capable of replanning the path while detecting unanticipated obstacles.
- Deadlock issue in multi-UAV systems: When multi-UAVs share the same environment, we face collision avoidance, and the other is deadlock avoidance. In fact, the deadlock issue is another type of collision avoidance. This is because when more UAVs block each other, resulting in that they are not able to continue their motions without collision. Most existing reactive control strategies, for example, velocity obstacle approaches, can solve the first problem, but fail to have the solution of the second problem. Although some approaches described in Section 4.6 have been conducted in dealing with the deadlock issue, it is still open when having large teams of UAVs in constrained and uncertain environments.
- Safe distance for worst case: In general, the minimum safe distance is a compulsory design parameter in collision avoidance algorithm. This requires to know the safe condition for worst case, i.e., UAV is able to come to a full stop before colliding with obstacles. Furthermore, the distance error caused by communication delays should also be considered in the safe distance.
- Distributed collision avoidance: Centralized approach has limited scalability and often require a global resource to perform collision avoidance. While this approach is suitable for small multi-UAV system, it does not scale for swarm control. We believe that the multi-UAV network will continue to grow in the future. Thus, the development of the distributed collision avoidance strategy will be essential for swarm control.
- 3-dimensional (3D) collision avoidance: The 2-dimensional collision avoidance problem has been studied intensively. However, the 3D collision avoidance is much more complex than that of 2D form. Existing 3D collision avoidance algorithm in Jenie et al. (2016) has presented a significant result. However, the computational load is large, especially in high accuracy requirement. This is not practical for a real-time UAV control. It is necessary to further develop a simplified 3D collision avoidance algorithm.
- Implementation issues: There are practical issues when implementing multi-UAV collision avoidance control. These include the following aspects.
  - Imperfect communication network. Messages from communication may be lost or delayed.
  - Real-time capability. Motion planning must respond within a real-time requirement to take a action.
  - UAV team. In a UAV team, one UAV may be lost and new UAV may be inserted into the team. How to replan the motion in the team will be a challenging work.
  - Cooperative intersection control. In a practical situation, an intersection point may not be simple like a cross road, and it may be one point of intersection of many lines (>4).

7. Conclusions

This paper has summarized collision avoidance approaches for multi-UAV systems. These approaches are mainly divided into path planning, conflict resolution, potential function, geometric guidance, and motion planning of teams of UAVs. The detailed comments have been given to show the main features of the proposed approaches. A table has also been given to summarize these works. In future research, we will implement the 3D velocity obstacle approach in several real vehicles and test their performance. It is also interesting to develop a multi-UAV motion planning algorithm for the fixed wing drones. Moveover, we will incorporate the current air traffic rules with 3D environments to develop a more realist collision avoidance algorithm.

Declaration of Competing Interest

The authors declare that they do not have any financial or non-financial conflict of interests.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.arcontrol.2019.10.001.

References

Alonso-Mora, J., Breitenmoser, A., Beardsley, P., & Siegwart, R. (2012). Reciprocal collision avoidance for multiple car-like robots. In *Proceedings of IEEE international conference on robotics and automation* (pp. 360–366). Saint Paul, MN, USA.

Alonso-Mora, J., DeCastro, J. A., Raman, V., Rus, D., & Kress-Gazit, H. (2018). Reactive mission and motion planning with deadlock resolution avoiding dynamic obstacles. *Autonomous Robots, 42*(4), 801–824.

Alonso-Mora, J., Naegeli, T., Siegwart, R., & Beardsley, P. (2015). Collision avoidance for aerial vehicles in multi-agent scenarios. *Autonomous Robots, 39*(1), 101–121.

Alrifaee, B., Kostyszyn, K., & Abel, D. (2016). Model predictive control for collision avoidance of networked vehicles using lagrangian relaxation. In *14th IFAC symposium on control in transportation systems* (pp. 430–435). Istanbul, Turkey.

Allawi, Z. T., & Abdalla, T. Y. (2014). An ABC-optimized reciprocal velocity obstacle algorithm for navigation of multiple mobile robots. In *The second engineering conference of control, computers and mechatronics engineering* (pp. 231–238).

Anderson, E. (2011). *Quadrotor Implementation of the Three-Dimensional Distributed Reactive Collision Avoidance Algorithm*. University of Washington. Master of Science in Aeronautics and Astronautics.

Andreychuk, A., Yakovlev, K., Atzmon, D., & Stern, R. (2019). Multi-agent pathfinding with continuous time. ArXiv:1901.05506v3 [cs.AI] 14 Jun.

Arambula Cosio, F., & Padilla Castañeda, M. (2004). Autonomous robot navigation using adaptive potential fields. *Mathematical and Computer Modelling, 40*(910), 1141–1156.

Augugliaro, F., Schoellig, A. P., & D'Andrea, R. (2012). Generation of collisionfree trajectories for a quadrocopter fleet: A sequential convex programming approach. In *Proc. IEEE/RSJ int. conf. intell. robots syst.* (pp. 1917–1922).

Bareiss, D., & van den Berg, J. (2013). Reciprocal collision avoidance for robots with linear dynamics using LQR-obstacles. In *Proc.of IEEE international conference on robotics and automation (ICRA)* (pp. 3832–3838). Karlsruhe, Germany.

Bellingham, J., Tillerson, M., Richards, A., & How, J. P. (2003). Multi-task allocation and path planning for cooperating UAVs. In S. Butenko, R. Murphey, & P. M. Pardalos (Eds.), *Chapter 2, in the book: Cooperative control: Models, applications and algorithms*. B.V. US: Springer-Science+Business Media.

Borenstein, J., & Koren, Y. (1989). Real-time obstacle avoidance for fast mobile robots. *IEEE Transctionson Systems, Man, and Cybernetics, 19*, 1179–1187.

Borenstein, J., & Koren, Y. (1991). The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation, 7*(3), 278–288.

Borrelli, F., Subramanian, D., Raghunathan, A., & Biegler, L. (2006). MILP and NLP techniques for centralized trajectory planning of multiple unmanned airvehicles. In *Proceedings of the american control conference* (pp. 5763–5768). Piscataway, NJ: IEEE Publ..

Cao, J., Teo, R. S. H., Huang, S., & Ren, Q. (2019). A fast, robust and decentralized approach for altitude de-confliction of multiple UAVs. In *Proc. of international conference on unmanned aircraft systems (ICUAS)* (pp. 79–85). Atlanta, GA, USA.

Čáp, M., Novák, P., Kleiner, A., & Selecký, M. (2015). Prioritized planning algorithms for trajectory coordination of multi-ple mobile robots. *IEEE Transactions on Automation Science and Engineering, 12*(3), 835–849.

Čáp, M., Novák, P., Selecký, M., Faigl, J., & Vokffnek, J. (2013). Asynchronous decentralized prioritized planning for coordination in multi-robot system. In *Proc. intell. robot. syst.* (pp. 3822–3829). Tokyo, Japan.

Carvalho, A., Gao, Y., Gray, A., Tsengy, H. E., & Borrelli, F. (2013). Predictive control of an autonomous ground vehicle using an iterative linearization approach. In *Proceedings of the 16th international IEEE annual conference on intelligent transportation systems* (pp. 2335–2340). The Hague, The Netherlands.

Chakravarthy, A., & Ghose, D. (1998). Obstacle avoidance in a dynamic environment: Acollision cone approach. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 28*(5), 562–574.

Chakravarthy, A., & Ghose, D. (2012). Generalization of the collision cone approach for motion safety in 3-d environments. *Autonomous Robots, 32*(3), 243–266.

Chao, Z., Zhou, S. L., Ming, L., & Zhang, W. G. (2012). UAV formation flight based on nonlinear model predictive control. *Mathematical Problems in Engineering, 2012*, 1–16. ID 261367.

Cohen, L., Wagner, G., Chan, D., Choset, H., Sturtevant, N., Koenig, S., & Kumar, S. (2018). Rapid randomized restarts for multi-agent path finding solvers. In *The eleventh international symposium on combinatorial search* (pp. 1909–1911). Stockholm, Sweden.

Cole, K., & Wickenheiser, A. M. (2018). Reactive trajectory generation for multiple vehicles in unknown environments with wind disturbances. *IEEE Transactions on Robotics, 34*(5), 1333–1348.

Conroy, P., Bareiss, D., Beall, M., & van den Berg, J. (2014). 3-d reciprocal collision avoidance on physical quadrotor helicopters with on-board sensing for relative positioning. ArXiv:1411.3794v1 [cs.RO] 14 Nov.

Connolly, C. I., Bums, J. B., & Weiss, R. (1990). Path planning using laplace's equation. In *IEEE conf. on robotics and aitromation* (pp. 2102–2106).

Cui, J., Zhang, Y., Ma, S., Yi, Y., Xin, J., & Liu, D. (2017). Path planning algorithms for power transmission line inspection using unmanned aerial vehicles. In *Proc.of control and decision conference* (pp. 2304–2309).

Daily, R., & Bevly, D. M. (2008). Harmonic potential field path planning for high speed vehicles. In *American control conference (2008), westin seattle hotel, Seattle, Washington, USA June 11–13* (pp. 4609–4614).

Dalamagkidis, K., Valavanis, K. P., & Piegl, L. A. (2009). On integrating unmanned aircraft systems into the national airspace system. In *International series on intelligent systems, control, and automation: Science and engineering: vol. 54* (pp. 43–59). New York: Springer Science + Business Media.

Delsart, V., Fraichard, T., & Martinez-Gomez, L. (2009). Real-time trajectory generation for carlike vehicles navigating dynamic environments. In *IEEE int. conf. on robotics and automation* (pp. 3401–3406). Kobe, Japan.

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

El. Khaili, M. (2014). Visibility graph for path planning in the presence of moving obstacles. *International Journal of Engineering Science and Technology, 4*(4), 118–123.

Erdmann, M., & Lozano-Pérez, T. (1987). On multiple moving objects. *Algorithmica, 2*, 1419–1424.

Fakoor, M., Kosari, A., & Jafarzadeh, M. (2015). Revision on fuzzy artificial potential field for humanoid robot path planning in unknown environment. *International Journal of Advanced Mechatronic Systems, 6*(4), 174–183.

Felner, A., Stern, R., Shimony, S., Boyarski, E., Goldenberg, M., Sharon, G., … Surynek, P. (2018). Search-based optimal solvers for the multi-agent pathfinding problem: Summary and challenges. In *The tenth international symposium on combinatorial search* (pp. 20–37). Pittsburgh, Pennsylvania, USA.

Fiorini, P., & Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research, 17*(7), 760–772.

Fox, D., Burgard, W., & Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Magazine, 4*(1), 23–33.

Ge, S. S., & Cui, Y. J. (2000). New potential functions for mobile robot path planning. *IEEE Transactions on Robotics and Automation, 16*(5), 615–620.

Gerdts, M., Henrion, R., Hömberg, D., & Landry, C. (2012). Path planning and collision avoidance for robots. *Numerical Algebra, Control and Optimization, 2*(3), 437–463.

Gregoire, J. (2014). Priority-based coordination of mobile robots. ArXiv preprint arXiv:14100879.

Gregoire, J., Cap, M., & Frazzoli, E. (2018). Locally-optimal multi-robot navigation under delaying disturbances using homotopy constraints. *Autonomous Robots, 42*, 895–907.

Hang, P., Huang, S., Chen, X., & Tan, K.K. (2020). Path planning and collision avoidance control of unmanned aerial systems in uncertain environment. ISA Transactions (under review).

Hennes, D., Claes, D., Tuyls, K., & Meeussen, W. (2012). Multi-robot col-lision avoidance with localization uncertainty. In *Proceedings of the intl. conf. on autonomous agents and multiagent systems* (pp. 147–154). Valencia, Spain.

Hönig, W., Preiss, J. A., Kumar, T. K. S., Sukhatme, G. S., & Ayanian, N. (2018). Trajectory planning for quadrotor swarms. *IEEE Transactions on Robotics, 34*(4), 856–869.

Hönig, W., Kiesel, S., Tinka, A., Durham, J. W., & Ayanian, N. (2018). Conflict-based search with optimal task assignment. In *Proceedings of the 17th international conference on autonomous agents and multiagent systems* (pp. 757–765). Stockholm, Sweden.

Hönig, W., Kiesel, S., Tinka, A., W. Durham, J., & Ayanian, N. (2019). Persistent and robust executionof MAPF schedules in warehouses. *IEEE Robotics and Automation Letters, 4*(2), 1125–1131.

Huang, S., Tan, K. K., & Lee, T. H. (2002). *Applied predictive control*. UK: Springer -Verlag.

Huang, S., & Teo, R. S. H. (2019). Computationally efficient visibility graph-based generation of 3d shortest collision-free path among polyhedral obstacles for unmanned aerial vehicles. In *2019 international conference on unmanned aircraft systems (ICUAS)* (pp. 1218–1223). Atlanta, GA, USA, June 11–14.

Huang, S., Teo, R. S. H., Leong, W. L., Martinel, N., Forest, G. L., & Micheloni, C. (2018). Coverage control of multiple unmanned aerial vehicles: A short review. *Unmanned Systems, 6*(2), 131–144.

Huang, S., Teo, R. S. H., & Liu, W. (2019). Distributed cooperative collision avoidance control for multi-unmanned aerial vehicles. *Actuators, 8*(1), 1–25.

Huang, S., Teo, R. S. H., Liu, W., & Dymkou, S. M. (2017a). Agent model for multi-UAV control via protocol designs. *International Journal of Intelligent Computing and Cybernetics, 10*(4), 412–429.

Huang, S., Teo, R. S. H., Liu, W., & Dymkou, S. M. (2017b). Distributed cooperative collision avoidance control and implementation for multi-unmanned aerial vehicles. In *Proc.of 11th asian control conference (ASCC)* (pp. 222–227). Australia: Gold Coast Convention Centre.

Huang, S., Teo, R. S. H., Liu, W., & Dymkou, S. M. (2019). Distributed UAV loss detection and auto-replacement protocol with guaranteed properties. *Journal of Intelligent & Robotic Systems, 93*(12), 303–331.

Hwang, I., & Tomlin, C. (2002). Protocol-based conflict resolution for finite information horizon. In *Proc. IEEE Amer. control conf.: 1* (pp. 748–753).

Jenie, Y. I., Kampen, E. J. V., Visser, C. C. d., Ellerbroek, J., & Hoekstra, J. M. (2015b). Selective velocity obstacle method for deconflicting maneuversapplied to unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics, 38*(6), 1140–1146.

Jenie, Y. I., Kampen, E. J. V., Visser, C. C. d., & Chu, Q. P. (2014). Velocity obstacle method for non-cooperative autonomous collision avoidance system for UAVs. In *AIAA guidance,navigation, and control conference* (pp. 1–12). National Harbor, Maryland, USA.

Jenie, Y. I., Kampen, E. J. V., Visser, C. C. D., Ellerbroek, J., & Hoekstra, J. M. (2016). Three-dimensional velocity obstacle method for uncoordinated avoidance maneuvers of unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics, 39*(10), 2312–2323.

Kochenderfer, M. J., Holland, J. E., & Chryssanthacopoulos, J. P. (2012). Next generation airborne collision avoidance system. *Lincoln Laboratory Journal, 19*(1), 55–71.

Kuchar, J. K., & Yang, L. C. (2000). A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems, 1*(4), 179–189.
Karaman, S., & Frazzoli, E. (2011). Linear temporal logic vehicle routing with applications to multi-UAV mission planning. *International Journal of Robust and Nonlinear Control, 21*(12), 1372–1395.
Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research, 5*(1), 90–99.
Kim, J. O., & Khosla, P. K. (1992). Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotic and Automation, 8*(3), 338–349.
Kloetzer, M., & Belta, C. (2007). Temporal logic planning and control of robotic swarms by hierarchical abstractions. *IEEE Transactions on Robotics, 23*(2), 320–330.
Kluge, B., & Prassler, E. (2006). Recursive probabilistic velocity obstacles for reflective navigation. field and service robotics. In S. Yuta, H. Asama, E. Prassler, T. Tsubouchi, & S. Thrun (Eds.), *Springer tracts in advanced robotics: vol. 24* (pp. 71–79). Heidelberg: Springer.
Koren, Y., & Borenstein, J. (1991). Potential field methods and their inherent limitations for mobile robot navigation. In *Proc. IEEE conf. robotics and automation* (pp. 1398–13404). Sacramento, CA.
Kuwata, Y., Wolf, M. T., Zarzhitsky, D., & Huntsberger, T. L. (2011). Safe maritime navigation with colregs using velocity obstacles. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 4728–4734). San Francisco, CA,USA.
Lalish, E. (2009). Distributed reactive collision avoidance for multivehicle systems. Phd thesis University of Washington. Seattle, WA, USA.
Lamport, L. (1994). The temporal logic of actions. *ACM Transactions on Programming Languages and Systems, 16*(3), 872–923.
Li, J., Harabor, D., Stuckey, P., Felner, A., Ma, H., & Koenig, S. (2019). Disjoint splitting for multi-agent path finding with conflict-based search. In *Proceedings of the twenty-ninth international conference on automated planning and scheduling (ICAPS)* (pp. 279–283). Berkeley, California USA.
Liu, W., & Hwang, I. (2014). Probabilistic aircraft mid-air conflict resolution using stochastic optimal control. *IEEE Transaction on Intelligent Transportation Systems, 15*(1), 37–46.
Lozano-Perez, T. (1983). Spatial planning: A configuration space approach. *IEEE Transactions on Computers, C-32*(2), 108–120.
Lozano-Pérez, T., & Wesley, M. A. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM, 22*(10), 560–570.
Ma, X., Jiao, Z., Wang, Z., & Panagou, D. (2018). 3-d decentralized priori-tized motion planning and coordination for high-density operations of micro aerial vehicles. *IEEE Transactions on Control Systems Technology, 26*(3), 939–953.
Matsuno, Y., & Tsuchiya, T. (2014a). Probabilistic conflict detection in the presence of uncertainty, electronic navigation research institute (ENRI). *Air Traffic Management and Systems, Lecture Notes in Electrical Engineering, 290*, 17–33.
Matsuno, Y., & Tsuchiya, T. (2014b). Stochastic 4d trajectory optimization for aircraft conflict resolution. In *IEEE aerospace conference* (pp. 1–10). Big Sky, MT, USA.
Minguez, J., & Montano, L. (2004). Nearness diagram (ND) navigation:collision avoidance in troublesome scenarios. *IEEE Transactions on Robotics and Automation, 20*(1), 45–49.
Mujumdar, A., & Padhi, R. (2011). Reactive collision avoidance of using nonlinear geometric and differential geometric guidance. *Journal of Guidance, Control, and Dynamics, 34*(1), 303–311.
Mujumdar, A., & Padhi, R. (2012). Reactive collision avoidance of using nonlinear geometric and differential geometric guidance. *Journal of Guidance, Control, and Dynamics, 34*(1), 303–311.
Muske, K. R., & Rawlings, J. B. (1993). Model predictive control with linear models. *A.I.Ch.E. Journal, 39*(2), 262–287.
Nakhaeinia, D., Tang, S. H., Noor, S. B. M., & Motlagh, O. (2011). A review of control architectures for autonomousnavigation of mobile robots. *International Journal of the Physical Sciences, 6*(2), 169–174.
Nikolos, I., Valavanis, K., Tsourveloudis, N., & Kostaras, A. (2003). Evolutionary algorithm based offline/online path planner for UAV navigation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B:Cybernetics, 33*(6), 898–912.
Nikou, A., Verginis, C., Heshmati-alamdari, S., & Dimarogonas, D. V. (2017). A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance. [cs.RO] 12 Nov. arXiv:1705.01426v2
Nguyen, M. H.-T., Tan, K. K., & Huang, S. (2011). Enhanced predictive ratio control of interacting systems. *Journal of Process Control, 21*(7), 1115–1125.
Pallottino, L., Scordio, V. G., Bicchi, A., & Frazzoli, E. (2007b). Decentralized cooperative policy for conflict resolution in multivehicle systems. *IEEE Transactions on Robotics, 23*(6), 1170–1183.
Panati, S., Baasandorj, B., & Chong, K. T. (2015). Autonomous mobile robot navigation using harmonic potential field. In *The 3rd international conference on manufacturing, optimization, industrial and material engineering (MOIME)* (pp. 1–7). Sheraton Kuta, Bali, Indonesia, vol.83.
Petti, S., & Fraichard, T. (2005). Safe motion planning in dynamic environments. In *Proc. IEEE RSJ int. conf. intell. robot. syst.* (pp. 2210–2215). Edmonton, AB, Canada.
Plaku, E., & Karaman, S. (2015). Motion planning with temporal-logicspecifications: Progress and challenges. *AI Communications, 29*(1), 151–162.
Pnueli, A. (1977). The temporal logic of programs. In *Proceedings of the 18th annual symposium on foundations of computer science (FOCS)* (p. 4657). doi:10.1109/SFCS.1977.32.
Rahmani, A., Kosuge, K., Tsukamaki, T., & Mesbahi, M. (2008). Multiple UAV deconfliction via navigation functions. In *AIAA guidance, navi-gation and control conference* (pp. 1–8). Honolulu, HI, USA.
Richards, A., Kuwata, Y., & How, J. P. (2003). Experimental demonstrations of real-time MILP control. In *AIAA guidance, navigation, and control conference and exhibit* (pp. 2003–5802). Austin, Texas,USA, pp. AIAA.
Richards, A., & How, J. P. (2003). Model predictive control of vehicle maneuvers with guaranteed completion time and robust feasibility. In *Proceedings of the american control conference* (pp. 4034–4040). Denver, Colo, USA.
Rimon, E., & Koditschek, D. E. (1992). Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation, 8*(5), 501–518.
Roelofsen, S., Gillet, D., & Martinoli, A. (2015). Reciprocal collision avoidance for quadrotors using on-board visual detection. *IEEE/RSJ Interna-tional Conference on Intelligent Robots and Systems*, 4810–4817.
Saha, I., Ramaithitima, R., Kumar, V., Pappas, G. J., & Seshia, S. A. (2014). Automated composition of motion primitives for multirobot systems from safe LTL specifications. In *Int. conf. intelligent robots and systems* (pp. 1525–1532).
Sariff, N., & Buniyamin, N. (2006). An overview of autonomous mobile robot path planning algorithms. In *Proc of the 4th student conference on research and development* (pp. 183–188).
Sharon, G., Stern, R., Felner, A., & Sturtevant, N. (2015). Conflict-based search for optimal multiagent pathfinding. *Artificial Intelligence, 219*, 40–66.
Shi, C., Zhang, M., & Peng, J. (2007). Harmonic potential field method for autonomous ship navigation. In *The 7th int. conf. ITS telecommun.* (pp. 1–6).
Shoukry, Y., Nuzzo, P., Balkan, A., Saha, I., Sangiovanni-Vincentelli, A. L., A. Seshia, S., … Tabuada, P. (2017). Linear temporal logic motion planning forteams of underactuated robots using satisfiability modulo convex programming. In *56th IEEE annual conference on decision and control (CDC)* (pp. 1132–1137).
Silver, D. (2005). Cooperative pathfinding. In *Proceedings of the 1st artificial intelligence and interactive digital entertainment conference (AIIDE)* (pp. 117–122). Marina del Rey, CA; United States.
Snape, J., van den Berg, J., Guy, S., & Manocha, D. (2011). The hybrid reciprocalvelocity obstacle. *IEEE Transactions on Robotics, 27*(4), 696–706.
Stanley, A. (2005). Flight path deconfliction of autonomous UAVs. *Infotech@Aerospace Conferences*, 1–21. Arlington, Virginia, USA.
Stroe, G., & Andrei, I. C. (2017). Automated conflict resolution in air traffic management. *INCAS BULLETIN, 9*(1), 91–104.
Tan, C. Y., Huang, S., Tan, K. K., Teo, R. S. H., Liu, W., & Lin, F. (2018). Collision avoidance design on unmanned aerial vehicle in 3d space. *Unmanned Systems, 6*(4), 277–295.
Tingbin, C., & Qisong, Z. (2013). Robot motion planning based on improved artificial potential field. In *The 3rd international conference on computer science and network technology* (pp. 1208–1211). Dalian, China.
Tomlin, C., Pappas, G. J., & Sastry, S. (1998). Conflict resolution for air traffic management: A study in multiagent hybrid systems. *IEEE Transactions on Automatic Control, 43*(4), 509–521.
Valbuena, L., & Tanner, H. G. (2012). Hybrid potential field based control of differential drive mobile robots. *Journal of Intelligentand Robotic Systems, 68*(3–4), 307–322.
Velagapudi, P., Sycara, K. P., & Scerri, P. (2010). Decentralized prioritized planning in large multirobot teams. In *Proc. IEEE/RSJ int. conf. intell. robot.* (pp. 4603–4609).
van den Berg, J., Guy, S., Lin, M., & Manocha, D. (2011). Reciprocal n-body collision avoidance. In C. Pradalier, R. Siegwart, & G. Hirzinger (Eds.), *Robotics research: the 14th international symposium ISRR, number 70 in springer tracts in advanced robotics* (pp. 3–19). Berlin, Heidelberg, Germany: Springer.
van den Berg, J., Wilkie, D., Guy, S., Niethammer, M., & Manocha, D. (2012). LQG-obstacles: Feedback control with collision avoidance for mobile robots with motion and sensing uncertainty. In *IEEE int. conf. on robotics and automation* (pp. 346–353). Saint Paul, Minnesota, USA.
van den Berg, J., Lin, M., & Manocha, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. In *Proceedings of the IEEE international conference on robotics and automation (ICRA 2008)* (pp. 1928–1935). Piscataway, NJ: IEEE Publ..
Wan, Y., Luo, J., Zhang, Q., Wu, W., Huang, Y., & Zhou, M. (2018). Controller design for avoiding collisions in automated guided vehicle systems via labeled petri nets. *IFAC PapersOnLine, 51*(7), 139–144.
Warren, C. W. (1993). Fast path planning method using modified a^* method. In *Proc.of IEEE international conference on robotics and automation* (pp. 662–667). Atlanta, USA.
Watanabe, Y., Calise, A., Johnson, E., & Evers, J. (2006). Minimum-effort guidance for vision-based collision avoidance. In *AIAA atmospheric flight mechanics conference and exhibit* (pp. 2006–6641). AIAA Paper.
Watanabe, Y., Calise, A. J., & Johnson, E. N. (2006). Minimum effort guidance for vision-based collision avoidance. *AIAA atmospheric flight mechanics conference and exhibit*. Pp.AIAA Paper 2006–6641, Keystone, CO, USA.
Wolff, E. M., Topcu, U., & Murray, R. M. (2014). Optimization-based trajectory generation with linear temporal logic specifications. In *IEEE int. conf. robotics and automation* (pp. 5319–5325).
Yang, L. C., & Kuchar, J. K. (1997). Prototype conflict alerting system for free flight. *Journal of Guidance, Control, and Dynamics, 20*(4), 768–773.
Yang, X., Mejias, L., & Bruggemann, T. (2013). A 3d collision avoidance strategy for uavs in a non-cooperative environment. *Journal of Intelligent Robotic Systems, 70*, 315–327.
Zhang, X., Huang, S., Liang, W., & Tan, K. K. (2019). HLT*: Real-time and any-angle path planning in 3d environment. In *Proc of the 45th annualconference of the IEEE industrialelectronics society (IES)* (pp. 14–17). Lisbon, Portugal, October.

Zhang, X., Ma, J., Huang, S., Cheng, Z., & Lee, T. H. (2019). Integrated planning and control for collision-free trajectory generation in 3d environment with obstacles. In *Proc.of the 19th international conference on control, automation and systems* (pp. 15–18). Jeju, South Korea.

Zhang, Y., Wang, F., Fu, F., & Su, Z. (2018). Multi-AGV path planning for indoor factory by using prioritized planning and improved ant algorithm. *Journal of Engineering and Technological Sciences, 50*(4), 534–547.

Zhang, Y., Yang, X., & Zhou, W. (2014). Flyable path planning for multiple UAVs in complicated threat environment. In *2014 international conference on multisensor fusion and information integration for intelligent systems (MFI)* (pp. 1–5). Beijing, China.

Zhou, Y., Hu, H., Liu, Y., & Ding, Z. (2017). Collision and deadlock avoidance in multirobot systems: A distributed approach.IEEE transactions on systems. *Man, and Cybernetics: Systems, 47*, 1712–1726.

Zhu, H., & Alonso-Mora, J. (2019). Chance-constrained collision avoidancefor mavs in dynamic environments. *IEEE Robotics and Automation Letters, 4*(2), 776–783.