
White Rose Research Online

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk
Universities of Leeds, Sheffield and York

Deposited via The University of Sheffield.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/141072/

Version: Accepted Version

Article:
Douthwaite, J., Zhao, S. and Mihaylova, L.S. (2019) Velocity obstacle approaches for multi-agent collision avoidance. Unmanned Systems, 7 (1). pp. 55-64. ISSN: 2301-3850

https://doi.org/10.1142/S2301385019400065

Electronic version of an article published in Unmanned Systems, 2019, https://doi.org/10.1142/S2301385019400065 © [copyright World Scientific Publishing Company] [https://www.worldscientific.com/worldscinet/us]

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

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


UNIVERSITY OF LEEDS
University of Sheffield
UNIVERSITY of York

Unmanned Systems, Vol. 0, No. 0 (2013) 1–9
© World Scientific Publishing Company

Velocity Obstacle Approaches for Multi-Agent Collision Avoidance

James A. Douthwaite, Shiyu Zhao, Lyudmila S. Mihaylova

Department of Automatic Control and Systems Engineering,
The University of Sheffield, Mappin Street, Sheffield, S1 3JD
E-mails: [jadouthwaite1,szhao,l.s.mihaylova]@sheffield.ac.uk *

This paper presents a critical analysis of some of the most promising approaches to geometric collision avoidance in multi-agent systems, namely, the velocity obstacle (VO), reciprocal velocity obstacle (RVO), hybrid-reciprocal velocity obstacle (HRVO) and optimal reciprocal collision avoidance (ORCA) approaches. Each approach is evaluated with respect to increasing agent populations and variable sensing assumptions. In implementing the localised avoidance problem, the author notes a problem of symmetry not considered in the literature. An intensive 1000 cycle Monte Carlo analysis is used to assess the performance of the selected algorithms in the presented conditions. The ORCA method is shown to yield the most scalable computation times and collision likelihood in the presented cases. The HRVO method is shown to be superior than the other methods in dealing with obstacle trajectory uncertainty for the purposes of collision avoidance. The respective features and limitations of each algorithm are discussed and presented through examples.

Keywords: Velocity Obstacles, Collision Avoidance, Multi-agent Systems, Unmanned Aerial Vehicles (UAVs)

1. Introduction

Collision avoidance within systems composed of multiple physical agents has been a challenge since their conception. From both a safety and system preservation perspective, collision prevention must be considered a fundamental aspect of its autonomy. The application of robotics in an increasingly wide range of civilian use cases has meant that new systems are being designed to navigate in cluttered, highly dynamic environments with an emphasis on reliability and safety. Furthermore, guaranteeing a systems reliability in scenarios where communication infrastructure is inconsistent, disrupted or malicious, present significant challenges to upcoming systems.

There are two main types of collision avoidance approaches: *i*) *cooperative*, *ii*) *non-cooperative* [2, 3]. Both types of approaches typically assume that the information about the object geometry is known. *Cooperative collision avoidance approaches* operate on the assumption of an implicit, unilateral communication layer through which the intent of any agent can be freely communicated to another. *Non-cooperative* approaches, however, operate with limited knowledge of the agent's objective, only the kinematic parameters that can be deduced by its own onboard tracking equipment. This reduces the situation to a localised sense, detect and avoidance (SDA) problem [4,5]. Robust non-cooperative methods are highly desirable in novel environments, or in the event communication is limited or obstructed. For this reason more sophisticated systems tend to combine aspects of cooperative techniques with SDA approaches as a fail-safe.

Previous approaches to solving the SDA problem include probabilistic modelling [6], conflict resolution interval and agent trajectory optimisation [5, 7–9]. Classical approaches include the application of potential fields as seen in [10] and numerous geometry based avoidance techniques [11,12]. The concept of the collision cone (CC) and the velocity obstacle (VO) is introduced in [13], and has grown in popularity within the multi-agent community. This is partly due to the geometric construction of velocity constraints being intuitive, but also allows a resolution velocity to be found quickly and with minimal obstacle information [14].

Iterations of the VO concept include the reciprocal velocity obstacle (RVO) [14–17], which has been shown to reduce oscillatory agent paths by considering the reactive nature of avoiding agents. Obstacles with variable acceleration are addressed under the notion of acceleration velocity obstacles (AVO) in [18]. Hybrid-reciprocal velocity obstacles (HRVO) are introduced in an effort to

*The authors gratefully acknowledge the support from the UK EPSRC under grant number EP/M506618/1. The Matlab® Open-source Multi-agent Simulator (OpenMAS) source code is available at `https://github.com/douthwja01/OpenMAS` [1]

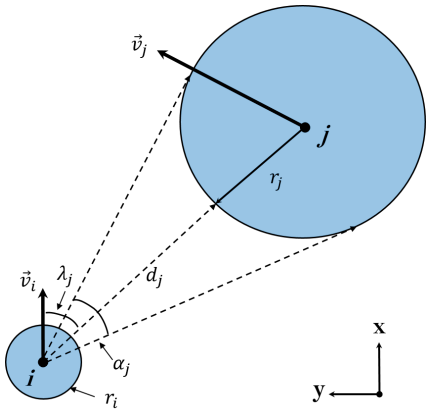

Figure 1. A description of the adopted sensor model defining the spherical position of obstacle j, at time k, as its position in the azimuth λ_j, distance $d_{j,k}$ and angular width $\alpha_{j,k}$.

eliminate direction ambiguity in [12, 19] and with it; the phenomenon known as the *reciprocal dance*. Despite producing smoother trajectories, the HRVO is not capable of guaranteeing that trajectories will be smooth. A method proposed to address this is the Optimal Reciprocal Collision Avoidance method, by adopting the concept of half-planes as linear constraints [20,21]. The kinematic velocity obstacles (KVO) presented in [22] demonstrate how additional agent kinematic assumptions and constraints might also be used to better approximate the escape velocity search volume. Similar methods demonstrate consideration for non-linear obstacle motion in [23] using a non-linear description of the agent and obstacle motion. The breadth of study within geometry based avoidance is substantial having been applied in various contexts, namely; pedestrian modelling, small robotic systems, unmanned aerial systems (UAS) and artificial intelligence (AI).

This paper presents some of the most promising approaches for collision avoidance in communication denied environments. These approaches are studied over a range of scenarios with increasing complexity in order to examine their effectiveness over increasing obstacle populations. In addition, we demonstrate how uncertainty in the observation of obstacles effects the performance of each algorithm in the presented cases. In an intensive 1000 cycle Monte Carlo analysis, we observe several key performance parameters, namely; rate of collision, computation time and minimum maintained separation. The pros and cons of each algorithm are summarised and presented for the reader. In computing agent trajectories locally, the author notes a problem of symmetry not considered in the literature. The findings from this study can be applied in a number of fields, from swarm systems to air traffic control. Two dimensional collision avoidance is considered, although the extension to the three dimensional case is natural.

The structure of the paper is as follows. Section 2 introduces the problem of interest, the agent and sensor constraints. Section 3 presents the mainstream velocity obstacle approaches to collision avoidance and their principle differences. Section 4 presents performance evaluation and validation of the considered algorithms. They are assessed and compared over several scenarios. Finally, Section 5 summarises the results.

2. Problem Description

We begin by considering the interactions of one agent i moving through an environment where numerous obstacles are moving freely. Both agents are moving through two dimensional (2D) Cartesian space. The agents velocities are denoted by $\vec{v}_i \in \mathbb{R}^{2\times 1}$ and $\vec{v}_j \in \mathbb{R}^{2\times 1}$, with representative radii r_i and r_j, respectively. The position of agent i at time t_{k+1} is defined as $\vec{p}_{i,k+1} = \vec{p}_{i,k} + \Delta t \cdot \vec{v}_{i,k}$ where Δt is the sampling time. We define a maximum speed constraint v_{max} that limits the velocities available to the avoidance routine, represented simply as $\|\vec{v}_i\| \leq v_{\max}$, where $||.||$ denotes the Euclidean norm. We assume that each agent is able to make its own observations of its surrounding using on board sensors such as a camera and range finder. The resulting measurements represent the spherical position of agent j in the form of an azimuth position, range and width, denoted by $\lambda_j \in [-\pi, \pi]$, $d_j \in [0, d_{max}]$ and $\alpha_j \in [-\pi, \pi]$, respectively. The parameter d_{max} is used here to describe the maximum visual range of agent i. Agent i observes agent j in its body axes as seen in Figure 1 [5].

Agent i measures the spherical position, $\lambda_{j,k}$, range $d_{j,k}$, and angular width $\alpha_{j,k}$ in the body axes of i. The agent then computes its equivalent Cartesian position $\vec{p}_{j,k} = [x_{j,k}, y_{j,k}]^T$ and radius estimate $r_{j,k}$ at time k. The Cartesian velocity of the obstacle is then calculated from successive position samples $\vec{v}_{j,k} = \frac{1}{\Delta t}(p_{j,k} - p_{j,k-1})$. The agents current knowledge of the obstacle j at time t_k, is then defined by the state vector $\vec{X}_{j,k} = [p_{j,k}, v_{j,k}, r_j]^T$.

All agents are assumed capable of retaining a time-variant set of states that correspond to all obstacles in a neighbourhood local to agent i. The limits of this neigh-

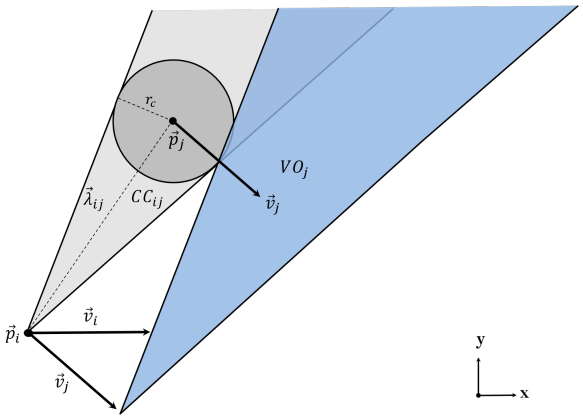

Figure 2. The VO_j (shaded blue) from the initial CC_{ij}. Here the VO_j defined in the configuration space of i, from the relative position λ_{ij}, configuration radius $r_c = r_i + r_j$ and velocity $\vec{v}_j$.

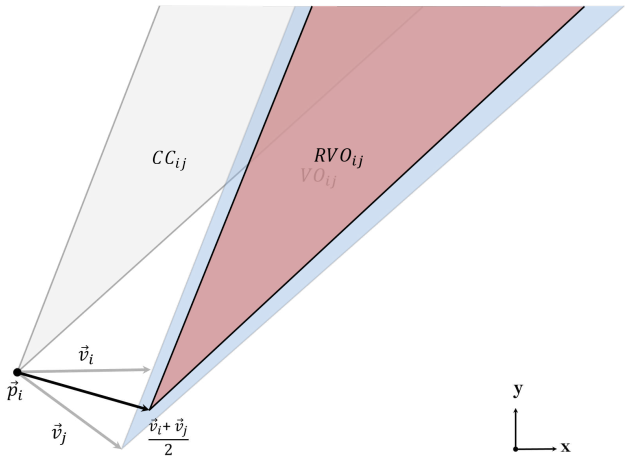

Figure 3. The construction of RVO_j by averaging the velocities of the agent $\vec{v}_i$ and obstacle $\vec{v}_j$.

bourhood is defined by a unilateral maximum neighbourhood separation $d_{j,k} \leq d_{\max}$ shown in Table 1. The agent population is assumed to be coordinated via some unspecified global objective. This objective is represented as a sequence of way-points in the global space; between which collisions may occur [24]. As an agent achieves a way-point, the target way-point is immediately reallocated as the agent moves into the next segment of the objective. To guarantee that collisions will otherwise occur, the way-points are selected to induce a conflict. In accordance with the SDA concept, the global position of the target way-point is presented locally to the agent by transforming into its body axes $\vec{p}_i^{\text{wp}} = \boldsymbol{R}_{i,k}^G(\vec{p}^{\text{wp}} - \vec{p}_i)$. Here $\boldsymbol{R}_{i,k}^G$ defines the global to body rotation matrix of agent i at time k. At all times, the position of agent i's way-point $\vec{p}_i^{\text{wp}}$ is assumed observable to agent i from its current position $\vec{p}_i$. The preferred velocity is that in the direction of $\vec{p}_i^{\text{wp}}$; expressed as $\vec{v}_i^{\text{pref}} = \frac{\vec{p}_i^{\text{wp}} - \vec{p}_i}{||\vec{p}_i^{\text{wp}} - \vec{p}_i||} \cdot v^{\text{pref}}$ where v_{pref} is the preferred speed.

3. Velocity Obstacle Methods

3.1. *The Velocity Obstacle*

The concept of the velocity obstacle (VO), based on the geometric assembly of the collision cone (CC), was first presented in [13]. Obstacles are observed in the agents local horizontal plane (XY) as their planar cross-section centred at $\vec{p}_j$ as seen in Figure 2. Here, the collision cone for obstacle j is defined as CC_{ij} from the geometric properties of the obstacles relative position $\vec{\lambda}_{ij}$, configuration radius r_c and velocity $\vec{v}_j$.

The VO is defined as a region containing the set of velocity vectors at t_k that will increase the chance of collision with obstacle j. It is assembled by translating CC_{ij} via the Minkowski sum $VO_{ij} = CC_{ij} \oplus \vec{v}_j$. In the consideration of multiple obstacles, the union of multiple $VO_{1:n}$ is taken. Agent velocities are therefore considered valid if $\vec{v}_{i,k+1} \notin VO_k = \cup_{j=1}^n VO_{j,k}$ [13]. Velocities satisfying this constraint describe a collision free trajectory for agent i in the presence of obstacles $VO_{j=1:n}$ for time t_k.

In practice, oscillatory trajectories are often observed in instances where two agents attempt to resolve a conflict with one another using the VO method. This often propagates until the point of collision occurs; as the two agents repeatedly resolve velocities $\vec{v}_{i,k+1}$ that imply a new conflict at t_{k+1} [15]. Obstacles that are static, or moving with constant velocity can otherwise be handled using the VO approach.

3.2. *The Reciprocal Velocity Obstacle*

An iteration of the conventional VO method, known as the RVO [15], attempts to consider the reciprocal motion of the second decision making agent j in order to produce smoother avoidance trajectories. The agent generates a VO with an apex augmented by the average of the two object velocities $\vec{v}_{i,k+1} \notin CC_{ij} \oplus (\vec{v}_{i,k} + \vec{v}_{j,k})/2$. This concept effectively allows the agent to mediate its correction trajectory $\vec{v}_{i,k+1}$ in accordance with $\vec{v}_j$. At time t_k, the RVO contains represents the region of velocities for i that are the average of both the velocity of agent i and the velocity of obstacle j.

The RVO is shown to eliminate the VO oscillation mentioned in Section 3.1 [15], and the resultant resolution trajectories are seen to be smoother. While this is the case, agent i and obstacle j do not explicitly agree on which sides they will approach each other. This can lead to scenarios where agents will mirror the trajectories of their respective obstacles in an attempt to avoid them. The oscillations induced by this behaviour, distinct from

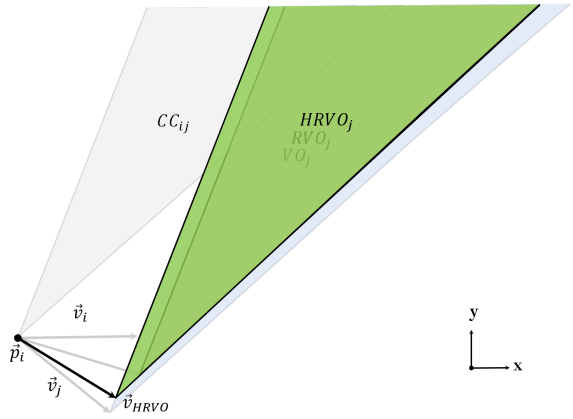

Figure 4. The relation of the hybrid reciprocal velocity obstacle HRVO to the initial VO_j and the RVO_j for a given obstacle B.

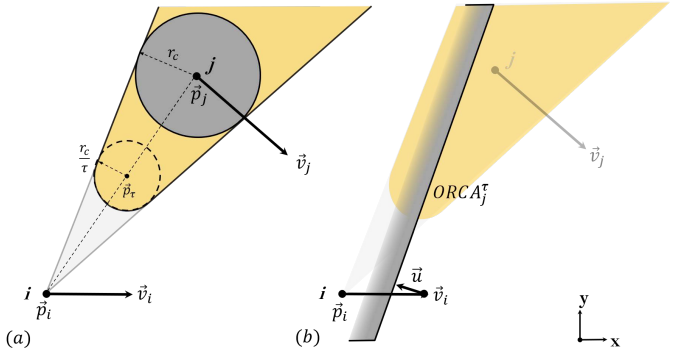

Figure 5. a) The geometric description of the truncated VO for obstacle j, defined by the truncation parameter τ, relative position $(\vec{p}_j - \vec{p}_i)$ and configuration radius $r_c = r_i + r_j$. b) The assembled ORCA obstacle and velocity correction $\vec{u}$ as a result of obstacle j.

those of the VO, are often referred to as a *reciprocal dance*.

3.3. *The Hybrid Reciprocal Velocity Obstacle*

An advancement on the VO problem has been proposed to negate the causes of reciprocal dance by augmenting the VO and RVO regions. The HRVO, shown in Figure 4, alters the apex of the HRVO in order to example different behaviour depending on the relative motion of the obstacle $\vec{v}_j$. The centerline of VO_j and RVO_j are co-linear in nature, therefore if the obstacle is moving right, the agent should resolve a trajectory $\vec{v}_{i,k+1}$ to pass the obstacle on the left and vice versa. Failure to do so brings about the phenomena of the reciprocal dance. Although the method is shown to improve the generation of smooth avoidance trajectories, it cannot guarantee it theoretically [12]. In the example given in Figure 4, directional bias is established by adjusting the apex of the $HRVO_j$ to be the intersection of the leading edge of RVO_j the trailing edge of VO_j (i.e. $HRVO_{ij} = CC_{ij} \oplus \vec{v}_{HRVO}$. The resulting constraint set imposed upon agent i at time t_k is then written $\vec{v}_{i,k+1} \notin HRVO_k = \cup_{j=1}^{n} HRVO_{i,k}$ [12].

Typically the RVO and HRVO are only necessary in the computation of inter-agent avoidance trajectories. The global VO set for agent i can instead be written as the union of the reciprocal variants (RVO or HRVO) for surrounding agents A_j and the VO for obstacles O_j: $\vec{v}_{i,k} \notin HRVO_k = \bigcup_{A_j=1}^{n} \mathrm{HRVO}_{A_j} \cup \bigcup_{O_j=1}^{n} \mathrm{VO}_{O_j}$.

3.4. *Optimal Reciprocal Collision Avoidance*

The RVO concept has be extended more recently in a method termed optimal reciprocal collision avoidance (ORCA). The ORCA approach is described well in [25], demonstrating how the ORCA velocity obstacle is formulated for a given reciprocally collision avoiding agent pair i and j. The resultant trajectory is not only smooth but, for small time steps, can be seen as continuous in the velocity space. The truncation parameter, τ, represents the time window for which a collision free trajectory should be guaranteed, i.e the agent can move at its new velocity for τ seconds.

If we assume that $\vec{v}_i$ and $\vec{v}_j$ are those that will bring about a collision in the future, then we define $\vec{u}$ as the vector to the point closest to the boundary of VO_j: $\vec{u} = (\arg\min_{\vec{v} \in \delta \mathrm{VO}^\tau} ||\vec{v} - (\vec{v}_i - \vec{v}_j)||) - (\vec{v}_i - \vec{v}_j)$(see Figure 5). Using the "outward" facing normal $\vec{n}$ of the boundary at the point $(\vec{v}_i - \vec{v}_j) + \vec{u}$ and the assumption that the responsibility that the avoidance is shared equally, the formulation for the $ORCA_j$ constraint can be written as $ORCA_k^\tau = \vec{v} | \vec{v} - (\vec{v}_i + \frac{1}{2}\vec{u}).\vec{n} \geq 0$. The geometric representation of $\vec{v}$ is given in Figure 5(b). Here it is represented as a "half-plane" with normal $\vec{n}$, with the initial point at $\vec{p} = \vec{v}_i + \frac{1}{2}\vec{u}$ [25].

The ORCA lines themselves allow the scenario to be described using only linear constraints. In addition, representation of the RVO as half-planes allows for simplification of the constraint set by eliminating those already covered by other ORCA lines, whilst guaranteeing continuously smooth agent trajectories.

3.5. *Trajectory Selection*

How the optimal resolution velocity is determined from the constraint sets defined in Sections 3.1-3.4, is also subject to strategy [13]. In the literature this is typically determined by considering the minimum deviation from a desired trajectory $\vec{v}_i^{\mathrm{pref}}$ subject to the union of the VO_k set. In such cases the optimal velocity can then be expressed as $\vec{v}_i^* = \arg\min_{\vec{v} \notin VO}(||\vec{v} - \vec{v}_i^{\mathrm{pref}}||)$. In this paper,

the optimal resolution velocity is determined similarly, using the *clear path* method [25], subject to the global constraint set of a given algorithm.

4. Experimental Results

4.1. *Experimental Conditions*

In this section we demonstrate the conflict resolution methods outlined in Section 3. The agent population is initialised with the parameters defined in Table 1. The noise parameters are applied to better represent sensor-derived measurement of the obstacle trajectory. Agents are designated a target way-point at the antipodal position of a concentric circle with a radius of 20m. The agents are tasked with crossing the circle to reach their way-point positions $\vec{p}_i^{\text{wp}}$ whilst avoiding collision. In Figure 6 the agent initialise at their origins (circles) and move through the collision centre to reach their respective way-points (triangles). Events such as collisions or way-point incidence are said to occur when the following condition is violated $||\vec{p}_i - \vec{p}_i^{\text{wp}}|| < (r_i + r_i^{\text{wp}}) - \gamma$, where the parameter γ is a condition tolerance that aims to eliminate ambiguity between collisions and narrow-misses caused by the nature of discrete simulation. The agent and scenario parameters are otherwise explicitly stated in Table 1.

Table 1. A table of the assumptions and agent parameters used in the preceding example scenarios, including the sensor uncertainties used in the representative sensing condition.

Parameter	Value
Maximum speed (v_{max})	2 m/s
Preferred speed (v_{pref})	1m/s
Agent critical radius (r_i)	0.5 m
Neighbour horizon (d_{max})	15 m
Camera standard deviation (σ_α)	5.208×10^{-5} rad
Range-finder standard deviation (σ_r)	0.5 m
Airspeed standard deviation (σ_s)	0.5 m/s
Position standard deviation (σ_p)	0.5 m
Agent orbital radius	10 m
Way-point orbital radius	20 m
Cycles	1000
Sampling rate (Δt)	0.25 s
Way-points & collisions tolerance (γ)	1×10^{-3} m

4.2. *Performance Evaluation*

The selected algorithms presented in Section 3 are validated over scenarios with an increasing number of agents. We examine the ten agent scenario and discuss the principle difference in the algorithms' performance. Figure 6 demonstrates the trajectories generated by the VO algorithm. When compared to the RVO in Figure 7 the trajectory adjustments can be seen to be abrupt, with greater oscillation throughout, until all conflicts are resolved. The compensation for obstacle movement is clearly seen in Figure 7 as the trajectories are shown more gradual. This indicative of the adjustment of the RVO in response to the movement of the obstacles; leading to fewer instances of harsh correction. Oscillation in the form of reciprocal dance can still be observed however as the direction of pass is resolved. In comparing the RVO trajectories to that of the HRVO in Figure 8; their is a clear reduction in the oscillation as the agents initially determine their direction of pass. The HRVO directional bias can also be observed from the agent trajectories, indicated by the emergent spiral behaviour around the conflict centre.

The representation of the VO as ORCA constraints is shown to produce trajectories similar to that the of HRVO in Figure 9. The linearity of the of the constraints however is shown to create smooth trajectories throughout the conflict scenario, resulting in a smaller overall course deviations. The selected algorithms were exampled in scenarios with 2, 5, 10 and 20 agents and their performance measured over 1000 Monte Carlo independent iterations. In addition to this, two sensor conditions were observed; A) Ideal Sensing; the agents are given perfect knowledge of the surrounding obstacles B) Representative Sensing; the agents adopt the sensor properties defined in Table 1.

Table 2. Algorithm performance of in the same 10 agent scenario. Condition A) Sensing capabilities are assumed ideal, Condition B) Assuming representative sensing. Each value represents the mean across 1000 independent Monte Carlo iterations.

Algorithm Condition	Mean Collisions	Mean Minimum Separation (m)	Mean Computation Time (ms)
Condition A			
VO	9.203	0.581	2.000
RVO	3.140	0.831	2.100
HRVO	0.053	0.996	2.400
ORCA	0.038	1.000	0.460
Condition B			
VO	7.749	0.624	2.000
RVO	9.380	0.577	2.100
HRVO	2.878	0.836	2.600
ORCA	6.881	0.757	0.463

The mean behaviour of the presented approaches are shown in Table 2, where a clear difference can be seen between the *Ideal* and *Representative* sensing conditions during the 10 agent example scenario. Under the assumptions of ideal obstacle telemetry, the compensative nature of the RVO is shown to reduce the mean number of collisions to 31.40%. This is a significant reduction from the 92.03% achieved in same scenario using the original VO method. The innate directional bias in the formation of the HRVO is shown to be a clear advantage over the

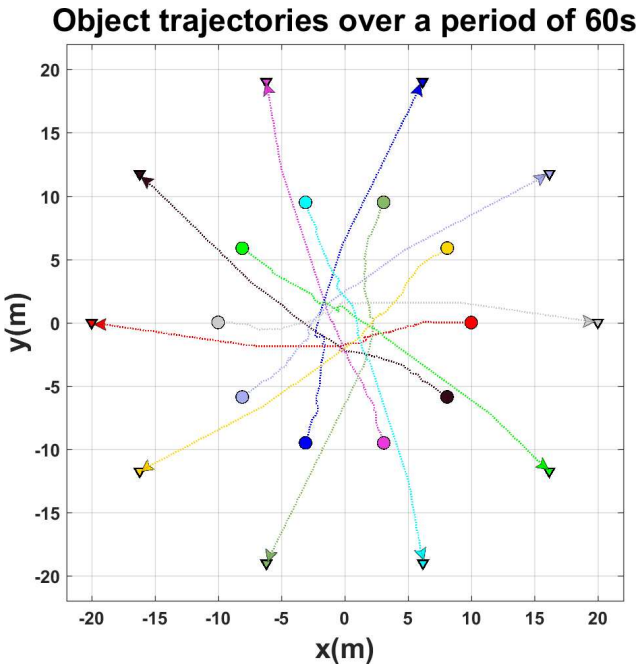

Figure 6. A depiction of ten agents using the VO based reactive avoidance in a concentric collision scenario. The oscillations due to obstacle compensative motion can be clearly observed as the agent progress towards the collision centre.

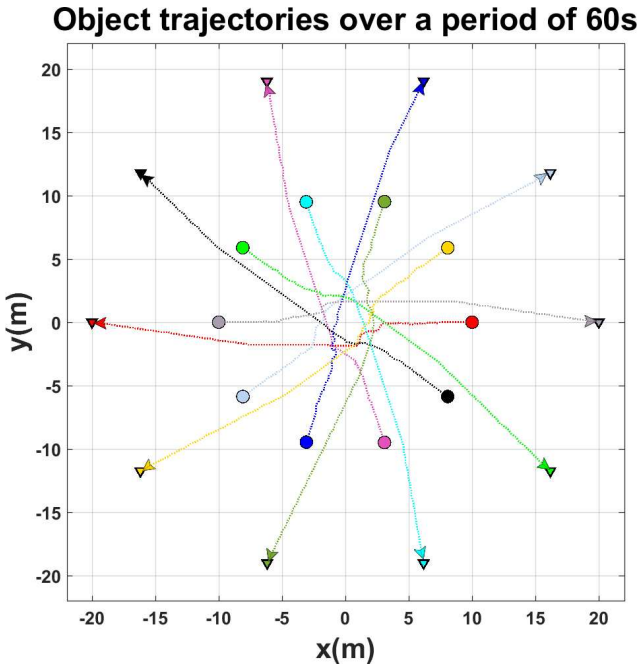

Figure 7. A depiction of the ten agent concentric scenario and applying the RVO based avoidance method. Abrupt trajectory changes can be seen observed, with distinct oscillations as new agents j enter the visual horizon of agent i.

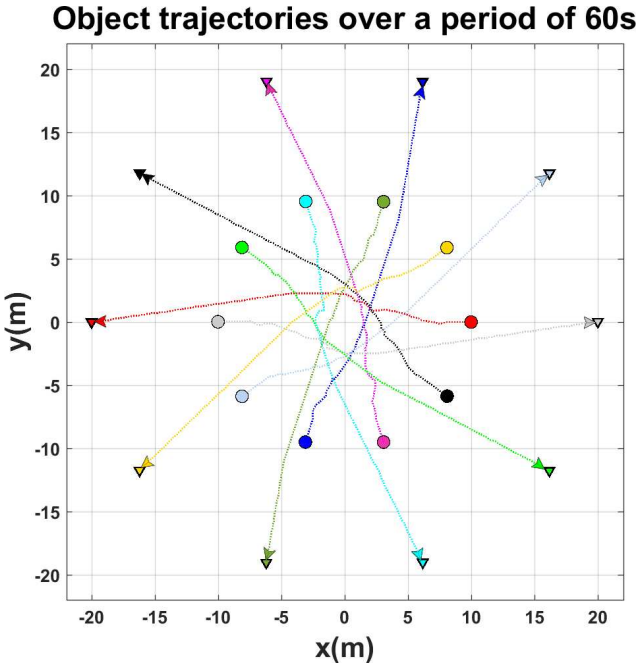

Figure 8. The ten agent concentric scenario repeated with the HRVO obstacle generation method applied. Oscillations can be observed as the procedure begins, however shown to be near linear as the direction of pass is resolved.

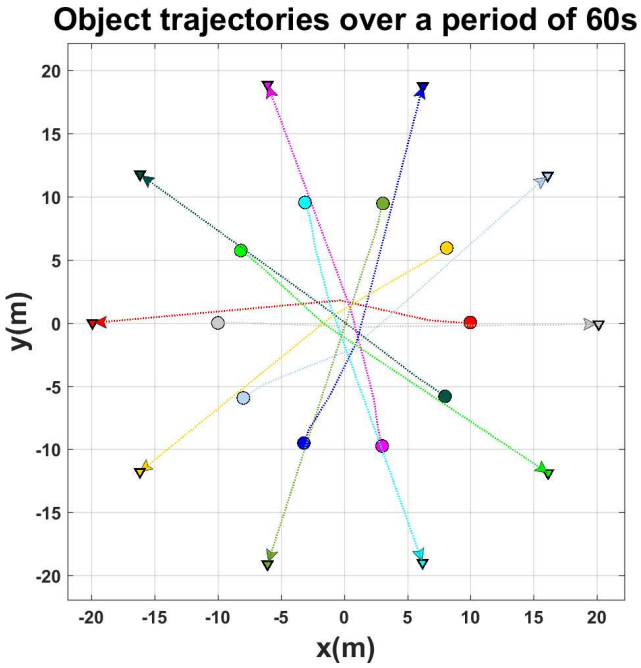

Figure 9. The same 10 agent scenario repeated under the ORCA obstacle generation method. The resultant trajectories appear as smoother, more gradual adjustments than the previous methods.

VO and RVO methods, resulting in a mean collision rate of 0.53%. The lowest mean number of collisions however was found using the ORCA method; averaging 0.38% over 1000 iterations.

Observing the behaviour of the algorithms in the presence of sensor uncertainty demonstrated a 5.08% mean increase in computation time. This can be seen more clearly in Figure 10. A disadvantage of the RVO method is shown here; with a factor of 3 increase in mean collision likelihood across the 1000 iterations. This may be due the aggravation of the reciprocal corrections (reciprocal dance) by the uncertainty in obstacle trajectory. Similar behaviour can also be observed for the ORCA algorithm, as the sensor uncertainty is shown to significantly reduced its effectiveness under this regime also. The mean minimum separation achieved by th ORCA approach was shown to be the closest to the $1m$ boundary condition. This suggests a clear benefit of the ORCA method to be its consistency in achieving safe separation in ideal conditions. Although, in considering uncertainty resulted in a mean collision likelihood 40.03% higher than the similarly effective HRVO approach.

Studying Figure 10, we observe an exponential relationship between the size of the agent population and the mean algorithm computation time for the VO, RVO and HRVO methods. The ORCA approach however, with its linear representation of the constraint set, is shown to yield computation times that scale linearly with increasing agent number. The relationship between the performance reduction rate $r_{ORCA} = 3.4 \times 10^{-5} s/n$ is shown to be distinctly lower than the other presented approaches. The ORCA algorithm therefore has a clear advantage when considering scalability for larger multi-agent systems, albeit more susceptible to uncertainty than the HRVO. All analyses were completed using an Intel Core i7-6600HQ quadcore (@2.8GHz) CPU. Code for the presented algorithms and scenarios are also available on Github [1].

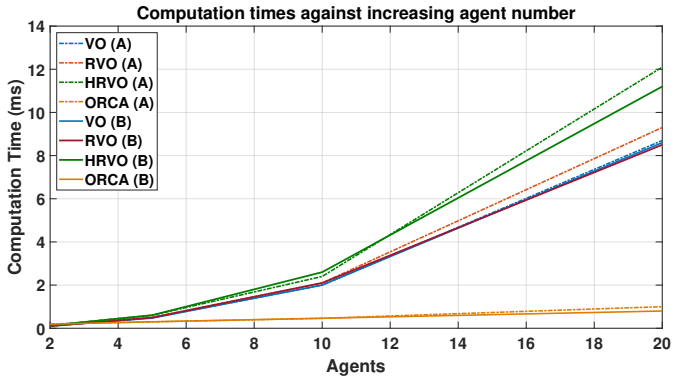

Figure 10. The mean algorithm computation times in both condition A) Ideal obstacle knowledge is assumed B) Obstacle telemetry data is subject to interference. Their effect on computation time is observed with an increasing number of obstacles.

The relation between the agent density and the number of collisions is shown in Figure 11. As expected, the addition of obstacle uncertainty is shown to generally induce a higher rate of collision in presented methods. This is with the exception of the original VO method; where the method is shown to be more effective with uncertainty. Methods considering both the velocity of i and j in the design of their constraints are shown to be more adversely effected by sensor noise. The most effective methods of avoiding collision are shown to be the HRVO and ORCA methods. However with representative sensor conditions the ORCA algorithm demonstrated a much higher rate of collision.

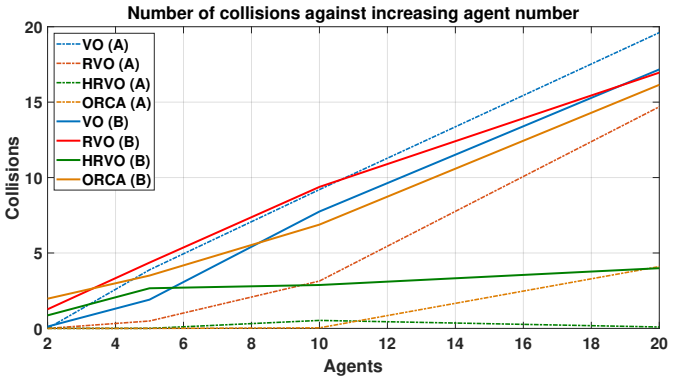

Figure 11. The mean rate of collision in both condition A) Ideal obstacle knowledge is assumed B) Obstacle telemetry data is subject to interference.

4.3. *A Problem of Symmetry*

In collision scenarios involving greater than two agents, there exists a problem of symmetry. While unlikely to occur in real systems, the situation may occur where an agent is presented with a constraint set that is symmetric about the forward direction $\vec{v}_i$ (as seen in Figure 12).

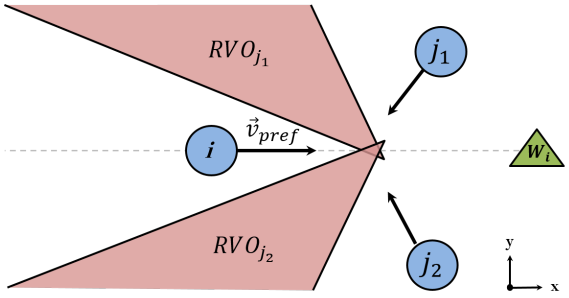

Figure 12. A depiction of the scenario where the symmetry of the constraint set will induce a *dead-lock* scenario.

As Figure 12 suggests, the agent will choose a velocity minimising the separation with the way-point W_i. Any velocity that acts away to alleviate the situation is less

optimal than the current preferred velocity. Unless a provision is made to allow the agent to violate a constraint momentarily, the agents velocity will tend to zero. This results in behaviour similar to the *Dead-lock* scenario in [12], where the density of the constraint sets mutually prevents any agent from progressing to their target positions.

In such scenarios a higher level strategy must be applied to intelligently preserve a collision-free trajectory by manipulating the constraint set or designing a new desired velocity ($\vec{v}^{\text{pref}}$). As part of the Monte Carlo analysis, the initial positions of the agents are perturbed by a small noise signal $\sigma_p = 0.5$m. This process also aids in the prevention of the dead-lock by ensuring that the scenario is asymmetrical.

5. Conclusions

In this paper several promising geometric approaches to the SDA problem are presented for use in multi-agent systems. Uncertainty in obstacle trajectory is shown to increase the mean computation time of all the proposed approaches without compensative measures.

The HRVO and ORCA methods are shown to be more effective in both negotiating obstacle cluttered environments whilst enduring uncertainty in obstacle trajectory. The ORCA method is also shown to generate both smoother resolution trajectories than the other presented methods despite low tolerance to obstacle uncertainty. The HRVO is shown to be statistically competitive with the ORCA in likelihood of collision, with higher tolerance to obstacle uncertainty. The benefit of the ORCA approach however, can clearly be seen in its scalability however, yielding computation times distinctly lower than the other methods.

The presented algorithms have shown that reactive collision avoidance can be sufficient to mitigate multiple collisions in a communication denied environment. Further work into inherent avoidance will examine such algorithms in the presence static and dynamic obstacles in more sophisticated coordinated tasks.

Bibliography

[1] J. A. Douthwaite, S. Zhao and L. S. Mihaylova, An open source multi-agent simulator (openmas) for matlab ® https://github.com/douthwja01/openmas (2018).

[2] S. B. Hottman, K. R. Hansen and M. Berry, Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems, Tech. Rep. September, Federal Aviation Administration, U.S. Department of Transportation, Washington, DC (2009).

[3] B. M. Albaker and N. A. Rahim, A survey of collision avoidance approaches for unmanned aerial vehicles, *Proceedings of the International Conference for Technical Postgraduates*, (IEEE, Kuala Lumpur, 2009), pp. 1–7.

[4] A. Zhahir, A. Razali and M. Ajir, Current development of unmanned aerial vehicle sense and avoid system, *In Proceedings of the Conference on Materials Science and Engineering*, **152**, Universiti Putra Malaysia (2016).

[5] J. A. Douthwaite, A. D. Freitas and L. S. Mihaylova, An interval approach to multiple unmanned aerial vehicle collision avoidance, *In Proceedings of the 11th Symposium Sensor Data Fusion: Trends, Solutions and Applications*, Bonn, Germany (2017).

[6] S. Ramasamy and R. Sabatini, A unified approach to cooperative and non-cooperative sense-and-avoid, *In Proceedings of the International Conference on Unmanned Aircraft Systems*, (2015), pp. 765–773.

[7] J. Marin, R. Radtke, D. Innis, D. Barr and A. Schultz, Using a genetic algorithm to develop rules to guide unmanned aerial vehicles, *In Proceedings of the International Conference on Systems, Man, and Cybernetics*, **1** (1999), pp. 1055–1060.

[8] A. Richards and J. How, Aircraft trajectory planning with collision avoidance using mixed integer linear programming, *In Proceedings of the American Control Conference*, **3**(2) (2002), pp. 1936–1941.

[9] H. Chen, V. Jilkov and X. Li, On threshold optimisation for aircraft conflict detection, *In Proceedings of the 18th International Conference on Information Fusion*, (2015), pp. 1198–1204.

[10] M. Suzuki and K. Uchiyama, Three-dimensional formation flying using bifurcating potential fields, *In Proceedings of the International Conference on Guidance, Navigation and Control*, **22** (2009), pp. 1753–1759.

[11] D. Wilkie, J. van den Berg and D. Manocha, Generalized velocity obstacles, *In Proceedings of the International Conference on Intelligent Robots and Systems*, (2009), pp. 5573–5578.

[12] J. Snape, J. van den Berg, S. J. Guy and D. Manocha, The hybrid reciprocal velocity obstacle, *Transactions on Robotics* **27**(4) (2011) 696–706.

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

[14] J. Guzzi, A. Giusti, L. Gambardella and G. Di Caro, Local reactive robot navigation: A comparison between reciprocal velocity obstacle variants and human-like behavior, *IEEE International Conference on Intelligent Robots and Systems* **Nov** (2013) 2622–2629.

[15] J. van den Berg, M. Lin and D. Manocha, Reciprocal velocity obstacles for real-time multi-agent navigation, *In Proceedings of the International Conference on Robotics and Automation*, (2008), pp. 1928–1935.

[16] J. Alonso-Mora, A. Breitenmoser, P. Beardsley and R. Siegwart, Reciprocal Collision Avoidance for Multiple Mobile Robots, *IEEE International Conference on Robotics and Automation*, (IEEXplore, Saint Paul, Minnesota, 2012), pp. 1–16.

[17] V. Santos, M. F. Campos and L. Chaimowicz, On segregative behaviours using flocking and velocity obstacles, *Distributed Autonomous Robotic Systems*, eds. M. "Ani Hsieh and G. Chirikjian Berlin, Heidelberg (2014), pp. 121–133.

[18] J. van den Berg, J. Snape, S. Guy and D. Manocha, Reciprocal collision avoidance with acceleration-velocity obstacles, *In Proceedings of the International Conference on Robotics and Automation*, (2011), pp. 3475–3482.

[19] J. Snape, J. Van Den Berg, S. Guy and D. Manocha, Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles, *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, ed. IEEE St Louis, USA (2009), pp. 5917–5922.

[20] A. Levy, C. Keitel, S. Engel and J. McLurkin, The Extended Velocity Obstacle and applying ORCA in the real world, *IEEE International Conference on Robotics and Automation*, **June**(June), Washington, Seattle (2015), pp. 16–22.

[21] M. Kim and J. Oh, Study on optimal velocity selection using velocity obstacle (OVVO) in dynamic and crowded environments, *Autonomous Robots* **40**(8) (2016) 1459–1470.

[22] J. Wilkerson, J. Bobinchak, M. Culp, J. Clark, T. Halpin-Cha, K. Estabridis and G. Hewer, Two-dimensional distributed velocity collision avoidance, *Pilotless Aircraft, Navigation and Guidance*, (February) (2014), pp. 1–8.

[23] Z. Shiller, F. Large, S. Sekhavat and C. Laugier, Motion planning in dynamic environments: obstacles moving along arbitrary trajectories, *In Proceedings of the International Conference on Robotics and Automation*, **4**, Grenoble, France (2001), pp. 3716–3721.

[24] B. Zhu, L. Xie, D. Han, X. Meng and R. Teo, A survey on recent progress in control of swarm systems, *Science China Information Sciences* **60**(7) (2017).

[25] J. van den Berg, S. Guy, M. Lin and D. Manocha, Reciprocal n-body collision avoidance, *Robotics Research*, eds. R. Pradalier, C.and Siegwart and G. Hirzinger Berlin, Heidelberg (2011), pp. 3–19.