# BRaVO: Biased Reciprocal Velocity Obstacles Break Symmetry in Dense Robot Populations

Seyed Abbas Sadat
*Autonomy Lab, School of Computing Science*
*Simon Fraser University*
*Burnaby, Canada*
*Email: sas21@sfu.ca*

Richard T. Vaughan
*Autonomy Lab, School of Computing Science*
*Simon Fraser University*
*Burnaby, Canada*
*Email: vaughan@sfu.ca*

***Abstract*—We present an extension to the Reciprocal Velocity Obstacles (RVO) approach to multi-robot collision avoidance with the aim of alleviating the problem of congestion caused by symmetrical situations in dense conditions. We show that in a resource transportation task RVO robots are unable to make progress due to crowds of robots with opposing navigation goals at source and sink. We introduce Biased Reciprocal Velocity Obstacles (BRVO), which breaks the symmetry among robots by giving priority to the robots leaving a task-related place of interest. BRVO is compared to RVO in two experiments and it is shown that BRVO is able to resolve the congestion much more quickly than RVO.**

***Keywords*-Collision Avoidance, Interference Reduction, Multi-Robot Systems**

## I. INTRODUCTION

Interference between robots is a common problem in multi-robot systems, particularly those with no centralized control. Interference in general can be described as competition for resources, for example access to a loading zone or a charging station. Usually robots simply get in each other's way during normal navigation in the environment. An acute version of the problem is seen when robots have opposing navigation goals, for example a robot leaving a shared recharging station interferes with robots arriving at that station. In these situations, some symmetry-breaking mechanism is needed to decide which robot has the priority and should go first. Such mechanisms have proved to increase the effectiveness of the multi-robot system [1].

Collision Avoidance (CA) is also a crucial part of most mobile robot systems. If robots fail to avoid obstacles (either stationary or mobile obstacles) physical damage might occur to them or to the environment and sometimes the whole system may fail. Researchers have proposed a number of approaches to collision avoidance in multi-robot systems with no centralized control. van den Berg *et al.* [2] introduced *Reciprocal Velocity Obstacles* method which has been demonstrated to be very effective for safe navigation in systems with multiple mobile robots. However, in dense situations, robots interfere with each other and there is no mechanism in RVO to break the symmetry among them. Hence those robots will move with the flow of the surrounding agents which might not be aligned with their goal direction. For example, consider the transportation task in which a group of autonomous robots should pick up items from a source position and then deliver them to a sink (Fig. 1). With RVO as the collision avoidance method, a robot that receives an item at source is stuck in the crowd of robots that want to pick up an item and because of the symmetrical situation, it can not continue its way to the sink. This situation frequently occurs in this task.

In this paper we present a modification to RVO that breaks the symmetry among robots in dense situations, thereby improving performance by reducing spatial interference.

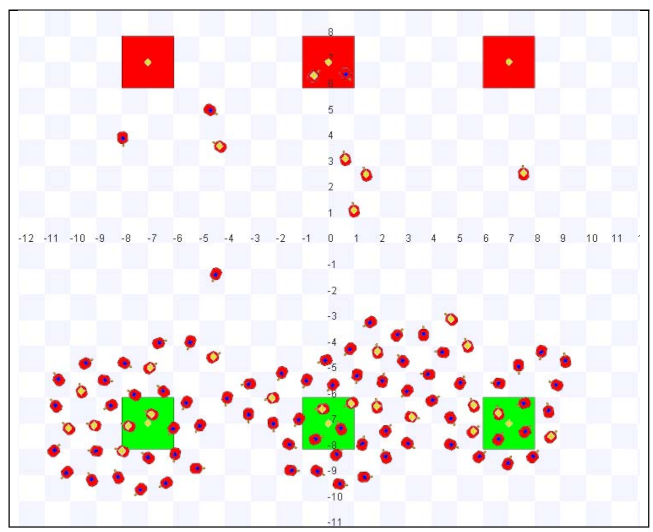


Figure 1: Stage simulation of transportation task. Robots (red octagons) must visit a source (bottom/green square) to collect a widget (yellow diamond) and deliver it to a particular sink (red/top square). Congestion around the sources has slowed task progress.

The rest of this paper is organized as follows. In section II, we summarize some of the main work in collision avoidance

978-0-7695-4683-4/12 $26.00 © 2012 IEEE
DOI 10.1109/CRV.2012.65

and spatial interference. Then we go over the reciprocal velocity obstacle approach in section III. we discuss the problem of RVO in dense situations and propose our solution to it (section IV). The experiments and results are explained in section V followed by a short discussion.

## II. Related Work

### A. Collision Avoidance

Various autonomous local collision avoidance methods have been described, differing in their approach to environment modeling and control functions. Among the most successful is the *Nearness Diagram* (ND), a reactive approach in which a motion heading is picked using a local model of the environment which is generated in form of a polar distance histogram [3]. By analyzing different possible situations to select motion direction, ND reduces the chances of deadlocks and oscillations, undesirable problems that are commonly seen in reactive navigation approaches. Also popular is the *Dynamic Window* (DW), a velocity-based reactive collision avoidance method with two phases. First it constructs the search space by only considering circular trajectories represented by $(\upsilon, \omega)$. A control command $(\upsilon, \omega)$ is discarded if the robot can not stop before it reaches the closest obstacle on the resulting curvature. Then, the velocities that can be achieved within a short time period are maintained in the search space, given the limited accelerations of the robot. In the second phase an objective function is maximized which has components of heading alignment with the preferred direction, distance to the closest obstacle on the trajectory and the magnitude of the velocity of the robot.

Both ND and DW consider moving entities as stationary obstacles and do not take into account their future behavior. In contrast, the recently developed *Velocity Obstacle (VO)* approach, explicitly considers the velocity of the moving objects in its model which makes this approach more suited for multi-robot collision avoidance [4]. VO operates in velocity space in which every object introduces a set of forbidden velocities whose shape is that of a cone. The robot chooses a velocity that is outside of these velocity obstacles to avoid collision. van den Berg *et al.* introduced Reciprocal Velocity Obstacles (RVO) which improves on VO by reducing an undesirable oscillation problem that occurs in multi-robot VO [2]. Recently, they proposed an optimal method of RVO for systems with multiple mobile robots [5]. We will describe the latter in more detail in Section III.

### B. Reducing Spatial Interference

In a mathematical model of robot foraging, Lerman *et al.* show that adding more robots to the system improved the group performance while decreasing individual robot performance [6]. Interference between robots reduces the marginal benefit of adding additional robots. Based on that model, an optimal group size was found that maximizes the group performance. Schneider-Fontan *et al.* [7] proposed territorial division to keep robots away from each other's work site and thus reduce interference. Stegaard *et al.* [8] suggested bucket brigading in multi-robot foraging, in which each forager restricts itself to a specific area and relies on other workers to deliver the resource to the destination.

Vaughan *et al.* studied explicit anti-interference strategies both in simulations [1] and real [9] robots to increase performance in the transportation task. Aggressive display behaviors were used as a mechanism to resolve conflicts in constrained parts of the environment. Based on the amount of work they had invested up to that point, robots selected an aggression level and the difference in aggression levels between interfering robots was used to break the symmetry. Scheidler *et al.* [10] proposed and studied several methods to reduce and control the emergent congestion in a dense ant like moving agent system. The asymmetries that resolve conflicts are introduced by modifying either the environment or the robot controllers. In addition to reducing congestion (and therefore improving the performace), the fairness of the proposed method is shown to be no lower than before. Also, in [11] the effect of restricting waypoint detection in the sensor's field of view is studied in an ant-like trail-following system. It is shown that a narrower field of view will result in multiple trails formed between places and interest and therefore the interference in each trail is reduced.

## III. Reciprocal Velocity Obstacles

In this section we briefly describe the Reciprocal Velocity Obstacles method introduced in [5].

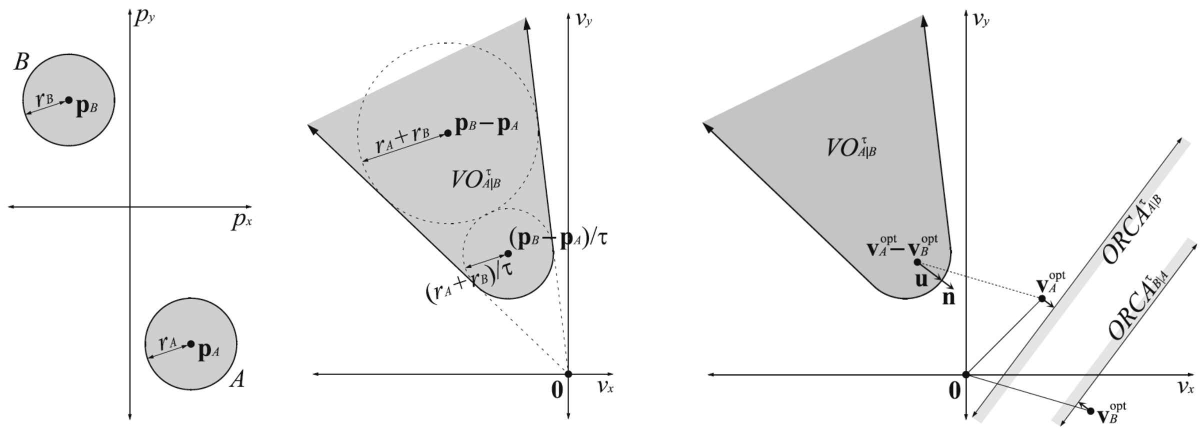
Let $D(\mathbf{p}, r) = \{\mathbf{q} \mid \| \mathbf{q} - \mathbf{p} \| \leq r\}$ in which $\mathbf{p}$ and $\mathbf{q}$ are 2D vectors and $r$ is a scalar value. Then for robots $A$ and $B$, the *velocity obstacle* $VO^{\tau}_{A|B}$ is defined as:

$$VO^{\tau}_{A|B} = \{\mathbf{v} \mid \exists t \in [0, \tau] :: t\mathbf{v} \in D(\mathbf{p}_B - \mathbf{p}_A, r_A + r_B)\}$$

($\mathbf{p}_A$ and $\mathbf{p}_B$ denote the positions and $\mathbf{v}_A$ and $\mathbf{v}_B$ denote the velocity of the robots $A$ and $B$). The velocity obstacle for robot $A$ induced by $B$, $VO^{\tau}_{A|B}$, is the set of all velocities relative to robot $B$ that will lead to collision of A and B at a time before time $\tau$ (see figure 2b). Now let $X \oplus Y = \{x + y \mid x \in X, y \in Y\}$ (Minkowski sum), then the set of collision-avoiding velocities for robot $A$ given that $B$ selects its velocity from $V_B$ is defined as:

$$CA^{\tau}_{A|B}(V_B) = \{\mathbf{v} \mid \mathbf{v} \notin VO^{\tau}_{A|B} \oplus V_B\}$$

$V_A$ and $V_B$ are called *reciprocally collision-avoiding* when $V_A \subseteq CA^{\tau}_{A|B}(V_B)$ and $V_B \subseteq CA^{\tau}_{B|A}(V_A)$ and they are called *reciprocally maximal* when $V_A = CA^{\tau}_{A|B}(V_B)$ and $V_B = CA^{\tau}_{B|A}(V_A)$. Here, the goal of the algorithm is to find sets of *permitted velocities* $V_A$ and $V_B$ such that they are reciprocally collision-avoiding and maximal and thus guarantee that $A$ and $B$ are collision-free for at least $\tau$ time. As there are many pairs of sets $V_A$ and $V_B$ that satisfy this requirement, the pair that maximizes the amount of




(a) (b) (c)

Figure 2: Reciprocal Velocity Obstacles. $(a)$ Robots are considered as disk-shaped holonomic agents with radius $r$ and position $p$. $(b)$ The $velocity\ obstacle$ for robot $A$ induced by $B$ is the set of all relative velocities for robot $A$ with respect to robot $B$ that will result in a collision at some time before $\tau$. $(c)$ RVO computes the minimum change $u$ to velocity of $A$ that will prevent a collision between $A$ and $B$. Figure reproduced from (J. van den Berg, S. Guy, M. Lin, and D. Manocha, *Reciprocal n-body collision avoidance*, Springer Tracts in Advanced Robotics, vol. 70, pp. 319, 2011.)

permitted velocities close to *optimization velocities* $\mathbf{v}_A^{opt}$ for $A$ and $\mathbf{v}_B^{opt}$ for $B$ are selected. These sets are denoted by $ORCA^{\tau}_{A|B}$ for $A$ and $ORCA^{\tau}_{B|A}$ for $B$. In [5], $\mathbf{v}_A^{opt}$ is set to $\mathbf{v}_A$ (i.e. the current velocity) for all robots $A$.

In order to construct $ORCA^{\tau}_{A|B}$ and $ORCA^{\tau}_{B|A}$, suppose that $A$ and $B$ are moving with velocities $\mathbf{v}_A^{opt}$ and $\mathbf{v}_B^{opt}$ respectively and that $\mathbf{v}_A^{opt} - \mathbf{v}_B^{opt} \in VO^{\tau}_{A|B}$ (i.e. they are on a collision course). Let $\mathbf{u}$ be the smallest change that is required to the relative velocity of $A$ and $B$ to prevent collision within $\tau$ time. Each robot take half the responsibility ($\frac{1}{2}\mathbf{u}$) to avoid collision. Then $ORCA^{\tau}_{A|B}$ is constructed as follows:

$$ORCA^{\tau}_{A|B} = \{\mathbf{v} \mid (\mathbf{v} - (\mathbf{v}_A^{opt} + \tfrac{1}{2}\mathbf{u})).\mathbf{n} \geq 0\}$$

in which $\mathbf{n}$ is the outward normal of the boundary of $VO^{\tau}_{A|B}$ at point $(\mathbf{v}_A^{opt} - \mathbf{v}_B^{opt}) + \mathbf{u}$. As it is shown in figure 2c, $ORCA^{\tau}_{A|B}$ is a half-plane in the velocity space.

For collision avoidance, each robot computes the half-plane of permitted velocities with respect to each other robot and the intersection of these sets is the permitted velocity for the robot with respect to all other robots:

$$ORCA^{\tau}_{A} = D(0, v_A^{max}) \cap \textstyle\bigcap_{B \neq A} ORCA^{\tau}_{A|B}$$

(the speed limit of the robot is also considered above). If the robot $A$ adopts any velocity from $ORCA^{\tau}_{A}$, it will prevent collision with other robots for at least $\tau$ time. In order to progress in the preferred direction, the robot selects a new velocity $V_A^{new}$ that is closest to its preferred velocity $\mathbf{v}_A^{pref}$. All robots repeat this cycle of sensing others' speed and position, computing permitted velocity set and finding the best new velocity to avoid collision with other robots.

In dense situations, the intersection of the permitted half-planes might be empty (i.e. $ORCA^{\tau}_{A} = \emptyset$), and thus there is no possible velocity by which the robot can prevent collision. For these situations van den Berg [5] proposes a way of computing a new velocity that is the "safest possible" and can gradually make a way out of the constraints imposed by other robots. The new velocity is selected such that it minimizes the maximum distance to any of the permitted half-planes induced by the other robots. This new velocity is not influenced by the preferred velocity and is dependent on the behavior of the surrounding robots.

## IV. Biased Reciprocal Velocity Obstacles

Consider a task in which the robots should visit a service station (e.g. charging station or the "sink" or the "source" in a transportation task) for a very short time and then leave it so that the other robots can do the same. With a large population of robots the area at the station may be packed by robots. Even with careful coordination, this occurs frequently in practice. In this dense situation, the set of permitted velocities $ORCA^{\tau}_{A}$ will be empty and RVO will choose the safest possible velocity which is independent of $\mathbf{v}_A^{pref}$ and thus the robots may not progress in the preferred direction. In practice we observe that it sometimes takes a long time for a robot to leave the crowd. This problem becomes worse given the kinematic and dynamic constraints of the non-holonomic robots (see Figure 1).

We simply extend Reciprocal Velocity Obstacles to deal with the congestion problem by breaking the symmetry among the robots in such a way that the priority is given to the robots that are leaving the place. Here, priority can be viewed as how close the new velocity is to the preferred velocity: if they are aligned, the robot is moving towards its goal, if they are different, the robot is making some space for others to pass. Therefore, when finding the set of permitted velocities for the leaving robots only, we compute the minimum change that should be applied to the $preferred$ velocity (rather than $current$ velocity as in [5]) so that the new velocity is out of the velocity obstacle induced by other robot. However, there is no change in the way that the entering robots compute their set of permitted velocities. Consequently robots that seek to leave the crowd will pick velocities that are close to their preferred velocity and continue their path (out of the crowd) to the goal, whereas other robots show more flexibility in their movement direction to avoid collisions. We call this new method Biased Reciprocal Velocity Obstacles (BRVO).

Asymmetry is achieved by defining a safety distance $d \geq 0$ such that if the served robots are within the radius $d$ of the service station, they use their preferred velocity as the optimization velocity and otherwise act the same as standard RVO:

$$\mathbf{v}_d^{opt}(k) = \begin{cases} \mathbf{v}^{pref} & \text{if } k \leq d \\ \mathbf{v}^{current} & \text{else} \end{cases}$$

In the above function, $k$ is the distance of the robot from the last service station and $d$ is the safety distance of that station. In this paper, the safety distance is the same for all stations in each experiment.

## V. Experiments and Results

We performed two experiments to examine the effectiveness of BRVO. The first one is a modifed version of the original RVO demonstration contrived to produce extreme congestion; the second examines performance in a more realistic task with emergent congestion.

### A. Experiment 1: Antiopodal with center visit

Experiments are perfomed using the freely available Antix simulator[1]. Disk-shaped, holonomic and homogenous mobile robots of radius $0.5m$ are arranged on the circumference of a circle, evenly spaced, and must move to their respective antipodal positions with the condition that each robot has to visit within radius $2.5m$ of the center of the circle at least once (Figure 4). The test is run with population sizes of 100, 200 and 300 robots, and repeated for RVO and BRVO controllers. This is a challenging task since the first robots that arrive at the center point are completely surrounded by incoming robots. We measured the time that each robot arrives at its goal. Figure 3 shows the result. It indicates that the robots can reach the final destination more quickly using BRVO than RVO.

[1]Antix is a simple kinematic simulator that supports large populations, see *http://github.com/rtv/Antix*

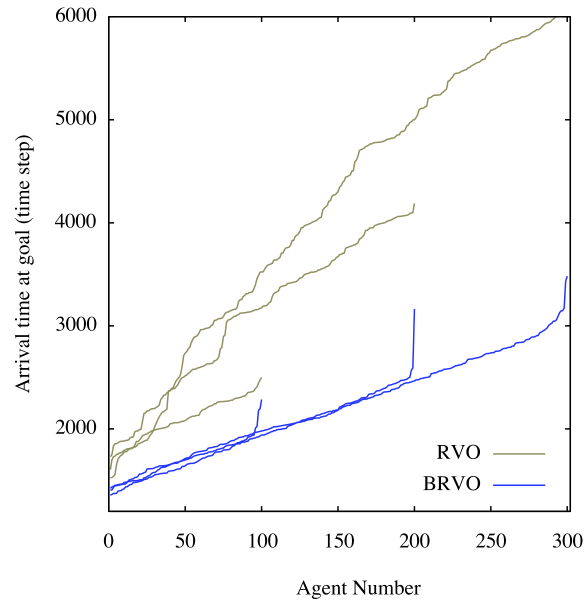


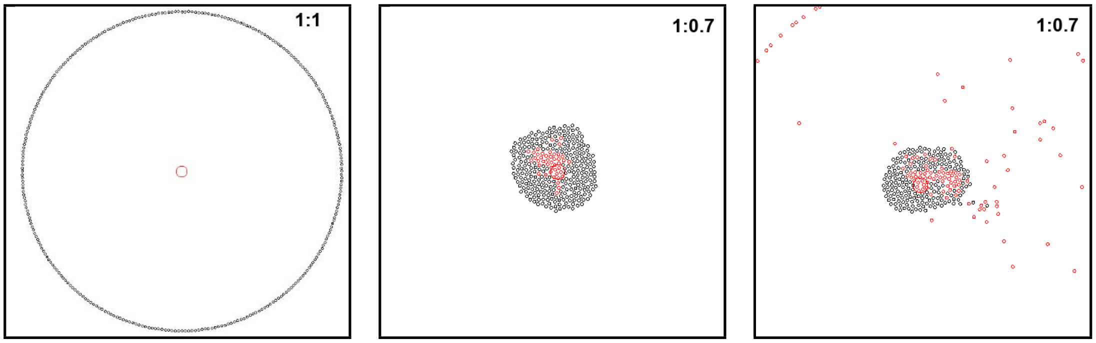

Figure 3: The time of arrival at the antipodal point. The experiment is performed with population sizes of 100, 200 and 300 with RVO and BRVO separately. The figure clearly indicates that the agents get to their goal quicker with BRVO compared to RVO.

When the robots are using RVO, those that have visited the center can not leave the area easily. This situation goes on until a 'bubble' of robots can escape out of the crowd and reach their goals. This phenomenon can be seen in the results graph: the curves for RVO have changes in gradient. The BRVO controller produces a more constant stream of robots out of the center (given the appropriate $d$ parameter), which leads to the more linear curves for BRVO. The results also demonstrate that the population size has very little influence on the congestion-resolving property of BRVO, i.e. the rate of the robots reaching their goal remains approximately constant when adding more robots.

Figure 5 shows the distance to goal over time for 300 agents. This figure indicates that with RVO robots spend more time at the circle center compared to BRVO. Additionally, the white spaces are the result of discontinuities in the stream of robots reaching their goal. This figure also shows a peculiarity in the behavior of BRVO (which is also evident in figure 3): an agent is sometimes backed off by two or more aggressive robots and becomes further away from its goal. This happens only occasionally, but it increases the task completion time for the last few robots (see the near-vertical lines near task completion for BRVO in Figure 3).

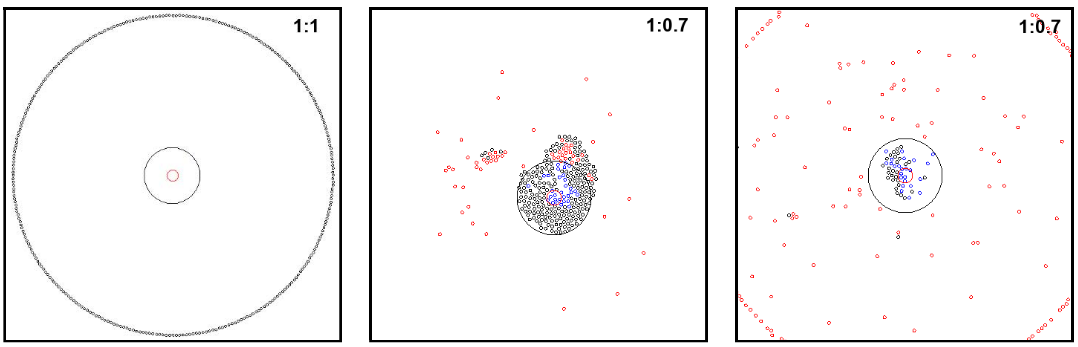
### B. Experiment 2

In a second experiment, we ran Stage [12] simulations with the task environment shown in Figure 1. The robots




initial configuration | time step 1000 | time step 2000

(a) RVO




initial configuration | time step 1000 | time step 2000

(b) BRVO

Figure 4: Experiment 1. Antix simulation screenshots. Starting with a circle formation, the robots visit the small center circle and then navigate to their respective antipodal positions. The top row shows the results using an RVO controller; the bottom row uses a BRVO controller. In the second row, the bigger circle in the middle shows the 'safety zone': after reaching the center, robots modify their velocity optimization function until they leave this zone.

(Stage's Pioneer 3DX models) should pick up a unit of resource at a source location and deliver it to a sink. Each unit is addressed to one of three sinks. The bottom (green) squares are the sources; top (red) squares are sinks. In the screenshots, robots (red octagons) are shown with yellow diamonds to indicate they are carrying a unit of resource. The robots start each trial at positions chosen from a random uniform distribution, but the sequence of sources and sinks that each robot visits is random but remains the same in all trials. Each trial runs for 30 simulated minutes, and the total number of resources delivered at the end of the trial is our performance metric. 10 trials are performed for each population size $p$.

The results of the simulation are summarized shown in Figure 6. Hypothesis testing using a T-test shows that the probability $P_{t-test}$ that the results are not from significantly different distributions for each population sizes of $p = 10, 20, 30, 40$ is 0.97, 0.022, 0.015 and 0.012 respectively. All population sizes above 40 are from significantly different distributions ($P_{t-test} << 0.001$). We conclude that in these tests BRVO performs similarly to RVO in populations up to 40, and better in populations above 40 up to 200.

Consistent with previous work examining spatial interference, we find that overall performance of the team increases with population size initially, until it reaches a maximum, after which the performance drops. Both RVO and BRVO have the same performance with $p \leq 30$. However, with more robots, BRVO gives superior average performance to RVO. The high variance observed with RVO is because robots often create a congested area around the sinks or sources which results in a degradation in performance. But, sometimes they manage to distribute themselves in time and space so that no large crowd of robots are present at the same time near the sinks or sources and thus there is no drop in performance. In case of BRVO, even if the source or the sink is congested, the leaving robots can easily navigate through the crowd and progress in the task and which results in the low variance and high performance.

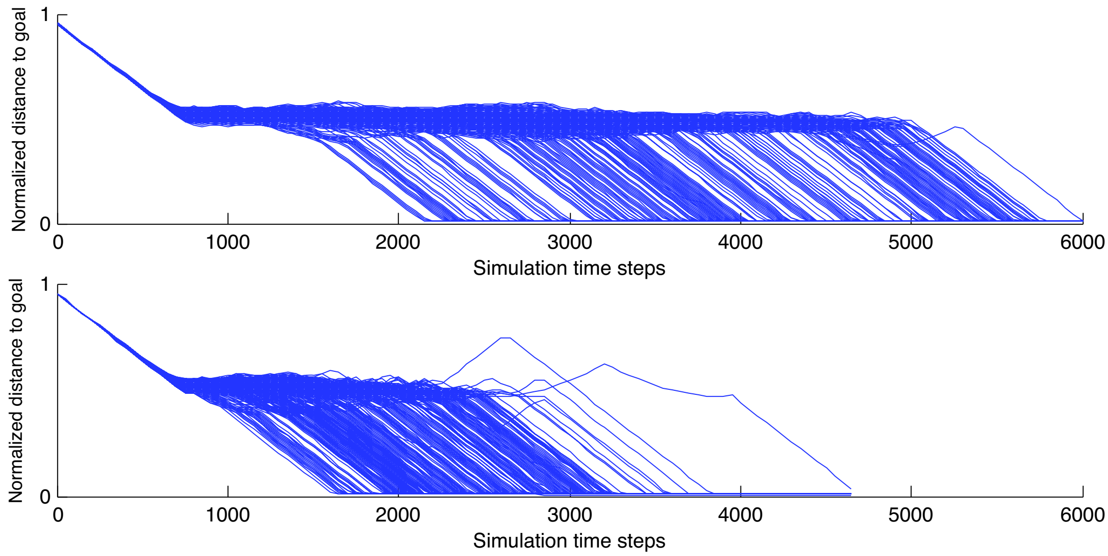



Figure 5: The normalized distance to goal over time for 300 agents using RVO (top) and BRVO (bottom). Robots spend more time at the center when using RVO.

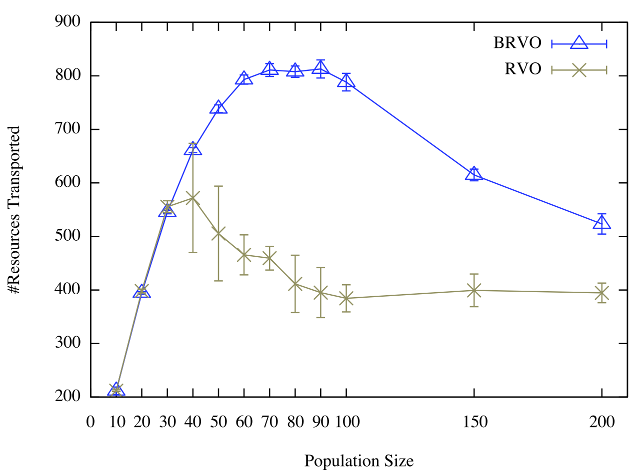



Figure 6: The number of resources transported vs. the robot population size. The figure indicates that with more robots BRVO is more successful in dealing with congestion and produces results with low variance whereas RVO is not able to generate satisfactory behavior.

## VI. Discussion

The Biased Reciprocal Velocity Obstacle acts effectively in densely packed situations where RVO is not able to generate satisfactory behavior. In RVO, those reciprocal collision avoiding sets are considered that are close to optimization velocities $\mathbf{v}_A^{opt}$ and $\mathbf{v}_B^{opt}$, and both robots are optimizing for the same optimization velocities, i.e. the current velocities. But in BRVO, this is invalidated since the leaving robot ($A$) uses its own preferred velocity ($\mathbf{v}_A^{pref}$) as $\mathbf{v}_A^{opt}$ (and still $\mathbf{v}_B^{opt} = \mathbf{v}_B$), whereas the other robot ($B$) uses the current velocities of $A$ and $B$ as the optimization velocities. This introduces a bias towards the preferred velocity to the new velocity of robot $A$ and, acting as a symmetry-breaking mechanism, it makes the robot more aggressive for getting out of the dense situation.

In this paper we assumed that the radii of the safety areas with high congestion of robots are provided *a priori*. This will impose a limitation on our method so that it cannot be used in situations that regions with potential high density are not known in advance. The creation of a method to dynamically estimate the size of such regions will be left for future work. Also, neither RVO nor BRVO guarantee safe navigation in dense conditions, so an emergency-stop mechanism must also be used to guarantee collision-free behaviour. Neither RVO or BRVO are deadlock-free, but we observe them both being very good in practice with hundreds of robots and with real-world system stochasticity enough to make deadlocks unlikely.

## VII. Conclusion and Future Work

In this paper we studied the performance of RVO in densely packed situations. We showed that in tasks that have a strong conflict in robot goals, such as having a shared service location which is consequently surrounded by robots, RVO is not able to exhibit a satisfactory behavior, i.e. the robots progress in their path very slowly. We proposed BRVO, a small extension to RVO, to deal with this problem by breaking the symmetry among robots so that those that are leaving the place are more biased towards their goal and are effectively prioritized to leave the place. The effectiveness of BRVO is shown with two different experiments in simulation. The results demonstrate that BRVO is capable of handling very dense situations. BRVO is a coordination method that integrates collision avoidance and interference reduction in a multi-robot system.

The reader may felt that the extensions to RVO we have examined are modest. We agree absolutely. Our modest contributions are: (i) we identified a problem with the otherwise excellent RVO that makes it unsuitable in some situations due to symmetry effects; and (ii) we propose the BRVO extension that is *as similar to RVO as we could make it*, while substantially breaking symmetry and thus addressing the problem. Our method is identical to RVO when robots are away from the problematic points, and we have shown empirically that it has significant performance benefits in two scenarios that are challenging for RVO.

In future we will show BRVO working on a real multi-robot system. Also we will work on methods to estimate the radius of safety zones dynamically.

## REFERENCES


[1] R. T. Vaughan, K. Støy, G. S. Sukhatme, and M. J. Matarić, "Go ahead, make my day: Robot conflict resolution by aggressive competition," in *Proceedings of the 6th International Conference on Simulation of Adaptive Behaviour (SAB)*, (Paris, France), pp. 491–500, August 2000.

[2] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pp. 1928 –1935, may 2008.

[3] J. Minguez and L. Montano, "Nearness diagram (nd) navigation: collision avoidance in troublesome scenarios," *Robotics and Automation, IEEE Transactions on*, vol. 20, pp. 45 – 59, feb. 2004.

[4] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[5] J. van den Berg, S. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," *Springer Tracts in Advanced Robotics*, vol. 70, pp. 3–19, 2011.

[6] K. Lerman and A. Galstyan, "Mathematical model of foraging in a group of robots: Effect of interference," *Auton. Robots*, vol. 13, no. 2, pp. 127–141, 2002.

[7] M. Schneider-Fontn and M. J. Mataric, "Territorial multi-robot task division," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 815–822, 1998.

[8] E. H. stergaard, G. S. Sukhatme, and M. J. Mataric, "Emergent bucket brigading - a simple mechanism for improving performance in multi-robot constrained-space foraging tasks," in *In Autonomous Agents*, pp. 2219–2223, 2001.

[9] M. Zuluaga and R. Vaughan, "Reducing spatial interference in robot teams by local-investment aggression," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (Edmonton, Alberta), August 2005.

[10] M. M. Alexander Scheidler, Daniel Merkle, "Congestion control in ant like moving agent systems," *Biologically-Inspired Collaborative Computing*, vol. 268, pp. 33–43, 2008.

[11] A. Sadat and R. T. Vaughan, "Blinkered lost: Restricting sensor field of view can improve scalability in emergent multi-robot trail following," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2010.

[12] R. T. Vaughan, "Massively multi-robot simulations in stage," *Swarm Intelligence*, vol. 2, no. 2-4, pp. 189–208, 2008.