Weighted Buffered Voronoi Cells for Distributed Semi-Cooperative Behavior

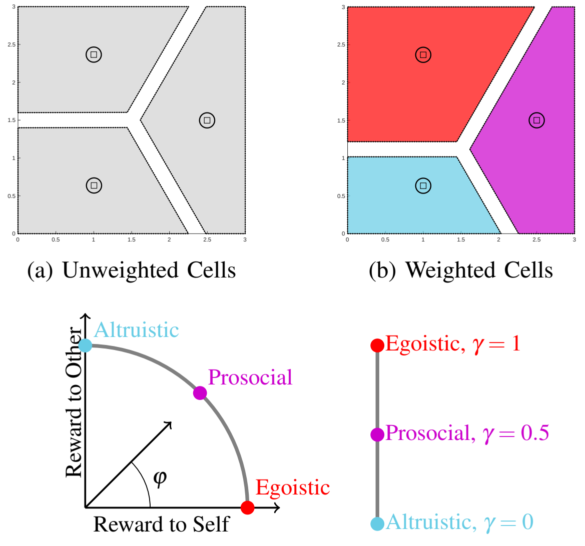
Alyssa Pierson[1], Wilko Schwarting[1], Sertac Karaman[2], and Daniela Rus[1]

***Abstract*— This paper introduces the Weighted Buffered Voronoi tessellation, which allows us to define distributed, semi-cooperative multi-agent navigation policies with guarantees on collision avoidance. We generate the Voronoi cells with dynamic weights that bias the boundary towards the agent with the lower relative weight while always maintaining a buffered distance between two agents. By incorporating agent weights, we can encode selfish or prioritized behavior among agents, where a more selfish agent will have a larger relative cell over less selfish agents. We consider this semi-cooperative since agents do not cooperate in symmetric ways. Furthermore, when all agents start in a collision-free configuration and plan their control actions within their cells, we prove that no agents will collide. Simulations demonstrate the performance of our algorithm for agents navigating to goal locations in a position-swapping game. We observe that agents with more egoistic weights consistently travel shorter paths to their goal than more altruistic agents.**

I. Introduction

We wish to consider multi-robot systems where different agents have different social preferences, ranging from egoistic to altruistic. One solution is to assign each agent preference using Social Value Orientation (SVO), a metric from social psychology that relates how an individual weights their reward to self versus the reward to others in social dilemmas. By assigning agents varying social preferences, we can design systems that are heterogeneous and incorporate semi-cooperative agents. In this paper, we encode these social preferences into the definition of a Voronoi tessellation, such that more egoistic and selfish agents have larger relative cells than the more altruistic agents in the environment.

For distributed multi-agent systems operating in cluttered environments, it is important to design control policies that allow for collision avoidance and scale with the number of agents. Voronoi cells can guarantee collision-free maneuvers of the group when all agents restrict planning to within their cell. To encode varying levels of cooperation, we present the Weighted Buffered Voronoi Cell (WBVC), which generates buffered cells with asymmetric boundaries depending on the relative cooperation between the agents. Using these weighted cells, we allow for semi-cooperative planning, where some agents may be more selfish than other agents. In this weighted tessellation, a more selfish agent has a larger cell than its less selfish neighbors. We present a control policy wherein agents navigate towards their goal location by choosing the closest point within their cell, with a right-handed heuristic for avoiding deadlock. Under this policy, the selfish agent's larger cell creates a larger available space for maneuvers, thus it is less likely to yield. We demonstrate this behavior in a position-swapping game between multiple agents with varying cooperation. Consistently, we observe more selfish agents take shorter paths to their goal than less selfish agents.



(a) Unweighted Cells

(b) Weighted Cells

(c) Social Value Orientation

Fig. 1: We encode Social Value Orientation (SVO) preferences in into a Voronoi tessellation using linear weights. (a) Unweighted vs (b) Weighted Buffered Voronoi tessellation, with egoistic (red), prosocial (purple), and altruistic (blue) agents. By encoding SVO into the Voronoi tessellation, egoistic agents receive a larger relative piece of the environment, which translates to priority when navigating.

Our solution is relevant to many robotic applications, including autonomous navigation, transportation, delivery services, tracking and surveillance, as well as systems with heterogeneous or semi-cooperative agents. Consider, for example, a fleet of package delivery robots working in a crowded environment. The robots do not necessarily have a centralized coordinator, and each robot may have a different priority service, akin to "overnight" vs "ground" shipping. Robots with the relative higher-priority should be given the right of way, while lower-priority robots may be expected to yield. By navigating with WBVC, agents with a higher-priority have larger cells and are more likely to not yield when traversing through an environment.

[1]Computer Science & Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139, USA [apierson, wilkos, rus]@mit.edu

[2]Laboratory of Information and Decision Systems (LIDS), MIT, Cambridge, MA, USA sertac@mit.edu

This work supported in part NSF Grant 1723943, the Office of Naval Research (ONR) Grant N00014-18-1-2830, and Toyota Research Institute (TRI). This article solely reflects the opinions and conclusions of its authors and not TRI, Toyota, or any other entity. Their support is gratefully acknowledged.

Related Work

To characterize the personality of agents and how that relates to varying degrees of cooperation, we draw inspiration from Social Value Orientation (SVO). SVO relates how an individual weights their personal reward against the reward to others in social dilemmas. Typically represented as an angle φ within a ring [1], [2], the angular preference indicates if a person is egoistic ($\varphi \approx 0$), prosocial ($\varphi \approx \frac{\pi}{4}$), or altruistic ($\varphi \approx \frac{\pi}{2}$), as illustrated in Figure 1a. We map this angular preference to a linear weight γ_i, which is then incorporated into our definition of the Voronoi tessellation. Social Value Orientation helps predict how others will behave and cooperate in social dilemma games, such as the prisoner's dilemma [3]. In [4], [5], authors show among humans, trust-based cooperation emerges as the expected utility-maximizing strategy when both social preferences and beliefs are taken into account. For robotics, SVO can both serve as a method for interacting with human agents, as well as assigning varying personalities to autonomous agents. As prior work, the authors incorporated SVO to estimate vehicle trajectories and predict human driver behavior for autonomous vehicles [6], as well as locally-optimize intersection traffic flow [7]. Here, we assign behaviors to autonomous agents from SVO definitions to encode which agents are naturally prioritized within our system.

Distributed navigation and collision avoidance is of great interest to multi-robot systems. One common approach for collision avoidance is Reciprocal Velocity Obstacles (RVO), which create velocity obstacles and assumptions on symmetric decision-making to ensure collision-free navigation for agents [8], [9], [10]. Other approaches include model predictive control (MPC) or receding horizon approaches to planning, where all other agents are treated as dynamic obstacles which an agent plans around [11], [12]. Recently, control barrier functions have been shown to define safe planning regions for agents, both for individual agents [13], [14] and swarm control [15]. Within distributed multi-agent systems, Voronoi cells are handy in a wide range of applications in coverage control [16], [17], [18], [19], [20], pursuer-evader dynamics [21], [22], [23], [24], and tracking [25], [26]. In [27], the agents navigate using the edges of the Voronoi tessellation created by obstacles. We extend the definition of Buffered Voronoi Cells (BVC) presented in [28], [29] to incorporate weighted cells. Weighted Voronoi cells allow us to encode a proxy of trust and cooperation [30], [31], [32], [33] among agents. To our knowledge, this is the first paper incorporating both agent social preferences and boundary safety buffers in the Voronoi tessellation definition.

The remainder of the paper is organized as follows: Section II defines the Weighted Buffered Voronoi tessellation and key properties. Section III contains our proof collision-free navigation and control algorithm for the simulations of the position-swapping game in Section IV. We state our conclusions in Section V.

II. Problem Formulation

Consider a bounded, convex environment $Q \subset \mathbb{R}^N$, with individual points in Q denoted q. Within the environment, there are n agents, with the positions of each agent denoted $p_i \in Q$ for $i \in \{1, ..., n\}$. Each agent has a radius r_i. We assume all agents have integrator dynamics,

$$\dot{p}_i = u_i, \tag{1}$$

where u_i is the desired control input for agent i. For agents navigating through an environment, we define $g_i \in Q$ as its goal location. Each agent also has a social preference $\gamma_i \in [0, 1]$, which represents their Social Value Orientation, shown in Figure 1a. We can convert between the angular notation and the linear weight using the following relationship:

$$\gamma_i = 1 - \frac{2\varphi_i}{\pi}, \tag{2}$$

such that egoistic agents are defined to have $\gamma_i = 1$, prosocial agents have $\gamma_i = 0.5$, and altruistic agents have $\gamma_i = 0$. While these values are assigned to specific definitions, γ_i can take any value within this range. These definitions map personalities to the robots, which in turn defines their desired level of cooperation. Next, we incorporate γ_i into the definition of the Voronoi cell weight to adapt the size of the cell based on relative differences, such that more egoistic agents will have larger cells than more altruistic agents.

A. Weighted Buffered Voronoi Cells

To weight Voronoi cells, we modify the definition of the Voronoi partition. Consider the standard Voronoi partition,

$$V_i = \{q \in Q | \ \|q - p_i\|^2 \leq \|q - p_j\|^2, \forall j \neq i, \quad i, j \leq n\}.$$

In the standard partition, the boundary between agents is the perpendicular bisector to the line between neighboring agents, and two agents i and j are equidistant to their shared Voronoi boundary. To move the location of the boundary, we add weights to the definition:

$$W_i = \{q \in Q \mid \|p_i - q\|^2 - w_i \leq \|p_j - q\|^2 - w_j\}, \tag{3}$$

where w_i and w_j are the weights of each agent. These weights adjust the cell boundaries between the agents, such that agents with larger relative weights will have larger cells. This is useful in sensor coverage, where it is desirable to assign a larger cell to agents with higher-quality sensors. Weights can also approximate trust between agents [33].

For robots with extent, buffered Voronoi cells ensure the spacing between cells allows for robots sitting on their Voronoi boundary to not collide with any other agent. To achieve this safety buffer, we define the weighted Voronoi cell with an asymmetric weight,

$$\mathcal{W}_i = \{q \in Q \mid \|p_i - q\|^2 \leq \|p_j - q\|^2 - w_{ij}\}, \tag{4}$$

where w_{ij} defines the cell weightings between agent i and neighbors $j \in \mathcal{N}_i$. The choice of w_{ij} weighting policy dictates the behavior of the system. For buffered Voronoi cells as defined in [28], [29], the weighting is

$$w_{ij} = (r_i + r_j) \|p_i - p_j\|, \tag{5}$$

which creates a gap of $(r_i + r_j)$ between agent boundaries as a safety radius buffer. Buffered Voronoi Cells guarantee agents with extent will not collide when situated on their boundary, where the buffer is generated around the original Voronoi boundary.

Here, we wish to define weighted cells that give more egoistic agents larger relative cells compared to more altruistic agents while incorporating the safety radius buffer. Recall the social preferences of each agent, γ_i, defines the relative cooperation with other agents, with values of $\gamma_i = 1$ corresponding to a purely egoistic agent, while a purely altruistic agent has $\gamma_i = 0$. We incorporate the social preferences into the Voronoi weighting as

$$\begin{aligned} w_{ij} = \left[1 + \frac{1}{2}(\gamma_i - \gamma_j)\right](r_i + r_j)\|p_i - p_j\| \\ - \frac{1}{2}(\gamma_i - \gamma_j)\|p_i - p_j\|^2. \end{aligned} \tag{6}$$

Note that when $\gamma_i = \gamma_j$, the weighting in (6) reduces to (5). Furthermore, when the agents are exactly a distance $d = (r_i + r_j)$ apart, the boundaries of the Voronoi cells are tangential to the agents. As the distance increases between the agents, a gap of $(r_i + r_j)$ remains between the two boundaries, but the agent with the higher relative γ has a larger cell and the boundary is not centered at the midpoint between agents. Figure 1 illustrates this point.

B. Properties of WBVC

We propose that agents can navigate through environments with guaranteed collision avoidance utilizing WBVC. To establish our collision avoidance guarantees, we first state several key properties of Weighted Buffered Voronoi Cells.

Definition 1 (Collision-Free Configuration). *For n agents with positions p_i and radius of extent r_i, let $d_{ij} = \|p_i - p_j\|$ denote the distance between agents. The configuration of agents is collision free if all agents satisfy*

$$d_{ij} \geq (r_i + r_j), \quad \forall i, j \in \{1, 2, ..., n\}, i \neq j.$$

For the group of agents in a collision-free configuration, we present the following three Lemmas that define key properties of the WBVC tessellation. First, Lemma 1 proves the WVBC $\mathcal{W}_i$ is always non-empty. Next, Lemma 2 proves the distance between two points in two different cells is always at least $(r_i + r_j)$. Finally, Lemma 3 proves that Weighted Buffered Voronoi Cells are non-overlapping when the agents are in a collision-free configuration.

Lemma 1 (Non-Empty Cells). *For each agent i with position p_i and cell $\mathcal{W}_i$, $\mathcal{W}_i \neq \emptyset$ when the agents are in a collision-free configuration.*

Proof. From the WBVC definition (4) with weights defined by (6), first substitute $q = p_i$ to find the definition of the cell at agent p_i.

$$0 \leq d_{ij}^2 - \left[1 + \frac{1}{2}(\gamma_i - \gamma_j)\right](r_i + r_j)d_{ij} + \frac{1}{2}(\gamma_i - \gamma_j)d_{ij}^2, \tag{7}$$

where $d_{ij} = \|p_i - p_j\|$. For agents to be collision-free, by Definition 1, $d_{ij} \geq (r_i + r_j)$. At the minimum spacing between agents, $d_{ij} = (r_i + r_j)$, and the right hand side of (7) reduces to zero. Since (7) remains true, we know that at this minimum spacing, the Voronoi cell $\mathcal{W}_i$ is non-empty and contains the position of the agent at p_i. For all $d_{ij} > (r_i + r_j)$, we can also see (7) remains true and $\mathcal{W}_i \neq \emptyset$. □

Lemma 2 (Minimum Distance). *For agents i and j with cells $\mathcal{W}_i$ and $\mathcal{W}_j$, let $\bar{p}_i$ denote a random point in $\mathcal{W}_i$ and $\bar{p}_j$ denote a random point in $\mathcal{W}_j$. The distance between any two of these points is $\|\bar{p}_i - \bar{p}_j\| \geq (r_i + r_j)$.*

Proof. Consider the point $\bar{p}_i$ in the definition of $\mathcal{W}_i$ from (4) and $\bar{p}_j$ in $\mathcal{W}_j$,

$$\|p_i - \bar{p}_i\|^2 \leq \|p_j - \bar{p}_i\|^2 - w_{ij},$$

where w_{ij} is defined by (6). Similarly,

$$\|p_j - \bar{p}_j\|^2 \leq \|p_i - \bar{p}_j\|^2 - w_{ji}.$$

By adding the two equations, we have

$$\begin{aligned} &\|p_i - \bar{p}_i\|^2 + \|p_j - \bar{p}_j\|^2 \\ &\qquad \leq \|p_j - \bar{p}_i\|^2 - w_{ij} + \|p_i - \bar{p}_j\|^2 - w_{ji}. \end{aligned} \tag{8}$$

From $\|p_i - p_j\| = \|p_j - p_i\|$, the sum of the weightings $w_{ij} + w_{ji} = 2(r_i + r_j)\|p_i - p_j\|$. Substituting this expression into (8), this reduces to

$$(\bar{p}_i - \bar{p}_j)^T (p_i - p_j) \geq (r_i + r_j)\|p_i - p_j\|.$$

For two vectors a and b, it is known that $\|a\|\|b\| > \|a^T b\|$. Let $a = (\bar{p}_i - \bar{p}_j)$ and $b = (p_i - p_j)$. We can then state

$$\|\bar{p}_i - \bar{p}_j\| \geq \frac{(\bar{p}_i - \bar{p}_j)^T (p_i - p_j)}{\|p_i - p_j\|} \geq \frac{(r_i + r_j)\|p_i - p_j\|}{\|p_i - p_j\|}, \tag{9}$$

thus proving that $\|\bar{p}_i - \bar{p}_j\| \geq (r_i + r_j)$ for any two points $\bar{p}_i \in \mathcal{W}_i$ and $\bar{p}_j \in \mathcal{W}_j$, for all $i \neq j$. □

Following Lemma 2, we derive our final Lemma stating that two Voronoi cells $\mathcal{W}_i$ and $\mathcal{W}_j$ are non-overlapping when the agent configuration is collision-free.

Lemma 3 (Non-Overlapping Cells). *For a point $\bar{p}_i$ belonging to $\mathcal{W}_i$, then $\bar{p}_i \notin \mathcal{W}_j$ for all $j \neq i$, and $\mathcal{W}_i \cap \mathcal{W}_j = \emptyset$.*

Proof. For a point $\bar{p}_i \in \mathcal{W}_i$, it must satisfy the cell definition in (4), or

$$\|p_i - \bar{p}_i\|^2 \leq \|p_j - \bar{p}_i\|^2 - w_{ij}.$$

Now we can plug $\|p_i - \bar{p}_i\|$ into the definition of $\mathcal{W}_j$ and check if it remains true,

$$\begin{aligned} \|p_j - \bar{p}_i\|^2 &\leq \|p_i - \bar{p}_i\|^2 - w_{ji}, \\ &\leq \|p_j - \bar{p}_i\|^2 - w_{ij} - w_{ji}, \\ 0 &\leq -2(r_i + r_j)\|p_i - p_j\|. \end{aligned}$$

From this contradiction, we know $\bar{p}_i \notin \mathcal{W}_j$. Since any point within a cell cannot belong to any other cell, we conclude $\mathcal{W}_i \cap \mathcal{W}_j = \emptyset$, $\forall i \neq j$. □

III. Collision-Free Navigation with WBVC

In this section, we present our algorithm for navigating through an environment with Weighted Buffered Voronoi Cells. We propose that all agents can navigate while avoiding collisions if they choose their control actions within their cell. Assumption 1 defines this constraint.

Assumption 1. *All agents choose their control policy $\dot{p}_i(t) = u_i$ such that their projected next position $\bar{p}_i = p_i(t) + u_i(t)\Delta t$ is within their Voronoi cell $\mathcal{W}_i$ at time t.*

This assumption is common in non-malicious multi-agent systems. Under this assumption, we are able to guarantee collision-free navigation of all agents in Theorem 1.

Theorem 1 (Collision-Free Navigation)**.** *For a group of n agents with initial positions $p_i(t_0)$ in a collision-free configuration, and all agents choose their control policy under Assumption 1, then all future configurations of agents $p_i(t > t_0)$ are collision free.*

Proof. For agents starting in a collision-free configuration, Lemma 1 states that all agents have a non-empty Voronoi cell, thus all agents have at least one point that can be chosen at the next time step. Let $\bar{p}_i$ be the targeted point of agent i at the next time step. Lemma 2 states that for any points $\bar{p}_i$ and $\bar{p}_j$, $\|\bar{p}_i - \bar{p}_j\| \geq (r_i + r_j)$. Thus, if any agent moves to any other point within its cell, the next configuration is by definition, also a collision-free configuration. We also know by Lemma 3 that no point $\bar{p}_i$ can belong to any other agent's cell, so no agents can target the same point and collide. □

In Theorem 1, we do not define the particular control policy of the agents, only that the choice of policy follows Assumption 1. For the simulations in Section IV, all agents move towards goal locations, outlined by Algorithm 1.

A. Algorithm Overview

Here, we present our algorithm for how agents can navigate towards their specified goal locations with their Weighted Buffered Voronoi Cell, summarized by Algorithm 1. For all agents to eventually reach their goal, we additionally assume that the goal locations are also a collision-free configuration of the agents.

Algorithm 1 Navigation in WBVC

```
Input: Collision-Free $p_i(t_0)$, $g_i \quad \forall\ i \in \{1,...,n\}$
while $\|g_i - p_i\| > 0, \quad \forall i \in \{1,...,n\}$ do
    Update weights $w_{ij}$ (6)
    Update tessellation $\mathcal{W}_i$ (4)
    Update control policy $u_i = -k(g_i - p_i)$
    Update $\bar{p}_i = p_i + u_i \Delta t$
    if $\bar{p}_i \in \mathcal{W}_i$ then
        Set $\dot{p}_i = u_i$ (goal is in cell)
    else
        Find intersection of $u_i$ and $\mathcal{W}_i$ boundary $\hat{p}_i$
        if $\|p_i - \hat{p}_i\| < \epsilon$ then
            Resolve $\hat{p}_i$ with deadlock heuristic
        end if
        Move towards boundary point $\dot{p}_i = -k(\hat{p}_i - p_i)$
    end if
end while
```

First, an agent checks if the goal location is contained within their cell $g_i \in \mathcal{W}_i$. If not, the agents then find the intersection between the line defined by $(g_i - p_i)$ and their Voronoi cell boundary, $\hat{p}_i$, and target this point. We know the closest point in their cell is on the Voronoi boundary [29]. To avoid deadlock, we also define an ϵ parameter to check the distance from the boundary of each agent. When agents are within a distance $\|\hat{p}_i - p_i\| < \epsilon$ from their boundary, we check for deadlock and if detected, choose a new target point $\hat{p}_i$ with a right-hand rule.

For our simulations in Section IV, we create a position-swapping game, where the goal location of the agent, g_i is set to the initial location of another agent $p_j(t_0)$, which guarantees that the initial and goal configurations are collision-free. Furthermore, these agents swap in pairs, so $g_i = p_j(t_0)$ implies $g_j = p_i(t_0)$. We choose pairwise swapping, as it encourages interactions between agents in our simulations.

B. Deadlock Heuristic

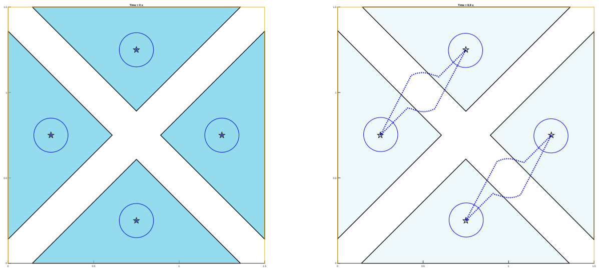
While we cannot guarantee that deadlock never occurs in a distributed system, we can design heuristics for the agents to avoid and resolve potential deadlock scenarios. We choose a right-handed rule, such that when an agent encounters a deadlock situation, it explores and yields to its right. Let b_i be the boundary edge segment of the cell $\mathcal{W}_i$ to the right of the intersection $\hat{p}_i$. For this paper, we find the closest point in b_i to the goal. If the agent is already at the closest point, we move $\hat{p}_i$ to the right by a fraction of the length of b_i.

IV. Simulations

Here, we present our Matlab simulations of a position-swapping game, which demonstrate how agents can move through cluttered environments to desired goal locations. First, we start with a four-agent swap to illustrate how the weighted cells affect yielding and trajectories. Next, we present a circle-swap game, where all agents start on the radius of a circle and must swap positions with the agent directly across from them. Finally, we present simulations for random initial positions and random target assignments. Across all simulations, we observe that agent with a more egoistic SVO take shorter paths to their goal than agents with more altruistic SVOs. Videos of all simulations are included with the submission of this paper.

A. Four-Agent Comparison

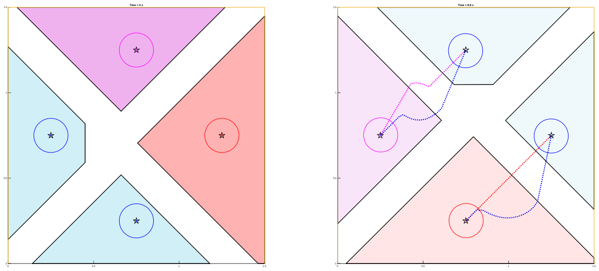
To illustrate how an agent's SVO preference translates to priority in navigation, we start with a four-agent swapping example. Figure 2 shows the initial and final positions of four agents with equal weightings. The agents navigate from their initial positions to their goal locations using Algorithm 1, shown in Figure 2b. We plot the trajectories of each agent on the final configuration, and note their inherent symmetry. This symmetry emerges from the same deadlock-avoidance maneuver, which both agents enact when they reach their boundary. Figure 3 shows this same four-agent position swap, but now one agent is egoistic ($\gamma_1 = 1$, red), while one agent is prosocial ($\gamma_2 = 0.5$, purple). The egoistic agent has a much larger WBVC area, and note the boundary between the agents is closer to the altruistic agents. Again, each agent navigates to their goal location with Algorithm 1, but the trajectories are no longer symmetric. Since the altruistic agent has a smaller cell, it reaches its boundary first, thus yielding first. The egoistic agent then does not have to yield, and can directly navigate along the shortest path to its goal location. Between the prosocial and altruistic agents, the prosocial agent yields slightly, but the altruistic still takes the larger deviation by yielding first.



(a) Initial Configuration

(b) Final Configuration

Fig. 2: For $n = 4$ agents with equal SVO preferences, the Voronoi tessellation reduces to the BVC. In a position swap game, the agents take symmetric paths.



(a) Initial Configuration

(b) Final Configuration

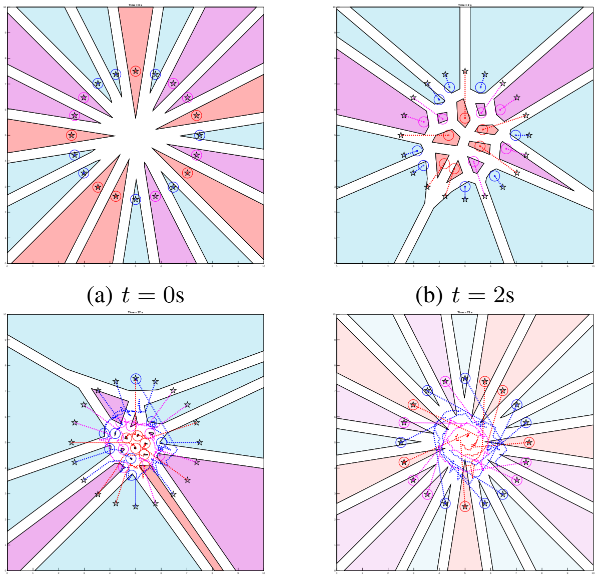
Fig. 3: Here, two agents are altruistic (blue), one agent is prosocial (purple), and one agent is egoistic (red). The altruistic agents yield to both the prosocial and egoistic agent. While the egoistic agent does not yield, the prosocial agent yields slightly but less than the altruistic agent.

B. Circle-Swap Game

In these simulations, we initialize all agents on a circle, and assign position swaps with the agent directly across the circle, which maximizes interactions and potential gridlock of all agents. We performed $m = 100$ simulations of $n = 20$ agents, all with radius $r_i = r = 0.2m$ and SVO preferences assigned randomly among the group. In each simulation, approximately $1/3$ of the agents are given an egoistic SVO of $\gamma_i = 1$ (red), $1/3$ are assigned a prosocial SVO of $\gamma_i = 0.5$ (purple), and the final third assigned an altruistic SVO of $\gamma_i = 0$ (blue). Figure 4 illustrates initial and final configurations during the circle swap. Note at the initial configuration, the agents with the more egoistic SVO have larger cells than the prosocial and altruistic agents.

Figure 5 plots both the distance to goal of each agent, as well as the distance to the nearest neighbor. All agents reach the goal, while maintaining a distance of at least $2r$ from any other agent. Across all trials, we did not observe deadlock, with all agents reaching their goal.

We propose that by planning with WBVC, more egoistic agents will have priority over more altruistic agents in path planning, as illustrated in Figure 3b. Figure 6a plots the additional distance agents travel compared to the shortest path, categorized by SVO preferences, for all agents over the $m = 100$ trials. Here, we classify additional distance as a percentage of the shortest path length. From Figure 6a, we note that egoistic agents have a lower additional distance and tighter distribution than prosocial and altruistic agents. Over the trials, the median distance traveled by egoistic agents was 20% over the shortest path length, 40% for prosocial agents, and 70% for altruistic agents.



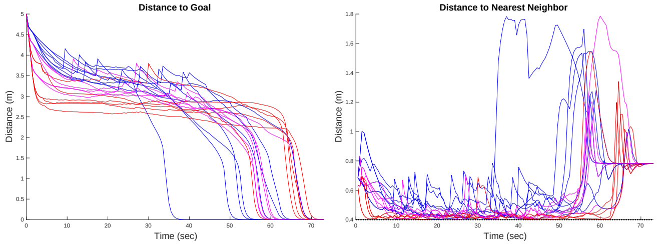
(a) $t = 0$s

(b) $t = 2$s

(c) $t = 37$s

(d) $t = 73$s

Fig. 4: Circular swap for $n = 20$ agents. The cell color indicates its SVO: egoistic agents ($\gamma_i = 1$) in red, prosocial agents ($\gamma_i = 0.5$) in purple, and altruistic agents ($\gamma_i = 0$) in blue. Note the egoistic agents start with slightly larger cells at $t = 0$s, which allows them to reach the center first. All agents reach their goal locations by $t = 73$s.



(a) Distance to Goal

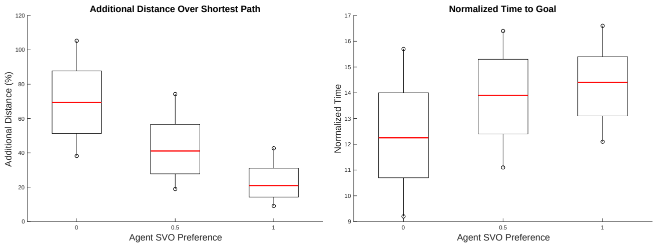
(b) Nearest Neighbor

Fig. 5: (a) Distance to goal over time. The color of the line indicates the agents' SVO. All agents reach the goal by t_f. (b) Minimum distance to neighbor of all agents, with $2r$ plotted as the dashed line, verifying no collisions occur.

Interestingly, for the circle game, the shortest path did not correspond to the shortest time, shown in Figure 6b. In fact, altruistic agents seem to reach their goal first. This occurs because egoistic agents, with the larger cells, are able to reach the center of the circle first (also shown in Figure 5), but then get stuck in gridlock. The altruistic agents remain as the outer agents, and are forced to yield around the outside of the group. As a result, the inner egoistic agents are effectively trapped until the outer agents move away from the center.

C. Random Navigation

As a final demonstration, we ran $m = 100$ trials with $n = 50$ agents with randomized initial positions and SVO preferences performing swaps with a random agent in the



(a) Additional Distance (b) Time to Goal

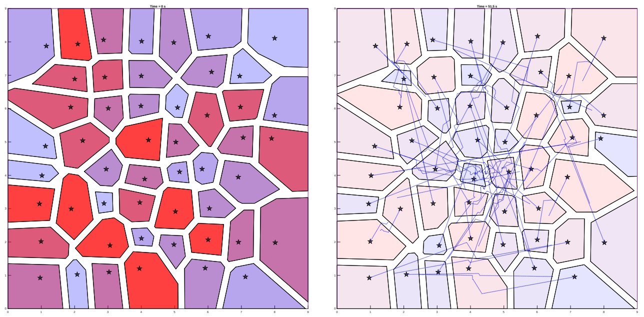
Fig. 6: (a) Distance traveled as compared to the shortest path over $m = 100$ trials for $n = 20$ agents in circle swap. Consistently, more altruistic agents travel longer paths than the more egoistic agents. (b) Time to reach the goal, by SVO. While the altruistic agents take longer paths, they also reach their goals faster. We mark the median, box the interquartile range, and the whiskers denote the $9th$ and $91st$ percentiles.

environment. While the previous circle swap simulations specified three possible values for SVO, here, agents are assigned an SVO value within $\gamma = [0,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1]$ for a finer discretization of preferences. Overall, we ran $m = 100$ trials of these randomized simulations, with all agents successfully reaching their goal in each trial (no observed deadlock). The agents are initialized in a $9m\ x\ 9m$ environment, with radius $r = 0.1m$, shown in Figure 7. Each initial configuration is collision-free, satisfying Definition 1.

Figure 8 plots the distance to goal and distance to nearest neighbor for all agents over time. We see all agents reach their goal locations, and the distance between agents is always greater than $2r$. Figure 9 shows the the distributions of additional distance and time to goal versus SVO for $m = 100$ trials. The additional distance is defined as additional percentage of the shortest path length the agent travels. Again, we see that as the SVO preference γ_i increases, agents deviate less from the shortest possible direct path, with more altruistic agents travelling longer paths. However, unlike the circle game, the more egoistic agents also tend to arrive in a shorter time. Due to the random configurations and swapping, we do not have the effect of all agents crowding a single area, as the egoistic agents are less likely to be trapped in this randomized environment.

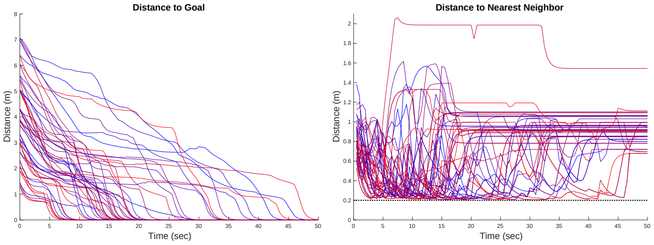
V. Conclusions

This paper presents the Weighted Buffered Voronoi tessellation, which allows us to encode social preferences and varying degrees of cooperation into a buffered Voronoi cell. For agents navigating through an environment, we guarantee collision avoidance when all agents choose control policies within their cells. Additionally, agents with more egoistic SVO preferences travel shorter paths when navigating towards their goal than more altruistic agents. We avoid deadlock in our position swapping games by employing a right-hand rule, consistent with other distributed multi-agent policies. Future work will example how the social preferences of the individual might be optimized to achieve optimal group performance.



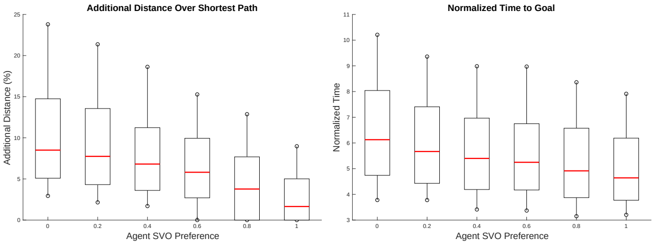
(a) Initial Configuration (b) Final Configuration

Fig. 7: Initial and final configuration for $n = 50$ agents performing position swaps with a random agent in the environment. Agents are initialized with random SVO preferences. The color scale indicates SVO preferences, with egoistic agents in red, and altruistic agents in blue.



(a) Distance to Goal (b) Nearest Neighbor

Fig. 8: (a) Distance to goal of all agents over time. The color of the line indicates the agents' SVO preference. All agents reach the goal by the end of the simulation. (b) Minimum distance to neighbor of all agents. Each agent has a radius of $r_i = 0.1m$. We verify no collisions occur by noting the minimum spacing is always greater than $2r_i$ (plotted in dashed line).



(a) Additional Distance (b) Time to Goal

Fig. 9: (a) Additional distance traveled over shortest path for $n = 50$ agents over $m = 100$ randomized trials. More altruistic agents travel longer paths compared to the more egoistic agents. (b) Time to goal, normalized by the shortest time to goal. Unlike the circle swap example, egoistic agents also gain a time advantage compared to more altruistic agents. We mark the median, box the interquartile range, and the whiskers denote the $9th$ and $91st$ percentiles.

References

[1] W. B. G. Liebrand, "The effect of social motives, communication and group size on behaviour in an n-person multi-stage mixed-motive game," *European Journal of Social Psychology*, vol. 14, no. 3, pp. 239–264, 1984. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ejsp.2420140302

[2] R. O. Murphy, K. A. Ackermann, and M. Handgraaf, "Measuring social value orientation," *Judgment and Decision Making*, vol. 6, no. 8, pp. 771–781, 2011.

[3] A. Rapoport and A. M. Chammah, *Prisoner's dilemma; a study in conflict and cooperation*. University of Michigan Press, 1965.

[4] J. Dijkstra and M. A. L. M. van Assen, "Explaining cooperation in the finitely repeated simultaneous and sequential prisoners dilemma game under incomplete and complete information," *The Journal of Mathematical Sociology*, vol. 41, no. 1, pp. 1–25, 2017.

[5] R. O. Murphy and K. A. Ackermann, "Social preferences, positive expectations, and trust based cooperation," *Journal of Mathematical Psychology*, vol. 67, pp. 45 – 50, 2015.

[6] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman, and D. Rus, "Social behavior for autonomous vehicles," *Proceedings of the National Academy of Sciences*, vol. 116, no. 50, pp. 24 972–24 978, 2019. [Online]. Available: https://www.pnas.org/content/116/50/24972

[7] N. Buckman, A. Pierson, W. Schwarting, S. Karaman, and D. Rus, "Sharing is Caring: Social Compliant Autonomous Intersection Negotiation," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

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

[9] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research*, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 3–19.

[10] A. Giese, D. Latypov, and N. M. Amato, "Reciprocally-rotating velocity obstacles," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 3234–3241.

[11] N. E. D. Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, Feb 2012.

[12] H. Zhu and J. Alonso-Mora, "Chance-constrained collision avoidance for mavs in dynamic environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 776–783, April 2019.

[13] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*, Dec 2014, pp. 6271–6278.

[14] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68 – 73, 2015, analysis and Design of Hybrid Systems ADHS. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S240589631502412X

[15] D. Panagou, D. M. Stipanovi?, and P. G. Voulgaris, "Distributed coordination control for multi-robot networks using lyapunov-like barrier functions," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 617–632, March 2016.

[16] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *Robotics and Automation, IEEE Transactions on*, vol. 20, no. 2, pp. 243–255, 2004.

[17] M. Schwager, D. Rus, and J.-J. Slotine, "Decentralized, adaptive coverage control for networked robots," *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 357–375, 2009.

[18] J. Cortés, "Coverage optimization and spatial load balancing by robotic sensor networks," *Automatic Control, IEEE Transactions on*, vol. 55, no. 3, pp. 749–754, 2010.

[19] S. G. Lee and M. Egerstedt, "Controlled coverage using time-varying density functions," in *Proc. of the IFAC Workshop on Estimation and Control of Networked Systems*, 2013.

[20] A. Sadeghi and S. L. Smith, "Coverage control for multiple event types with heterogeneous robots," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 3377–3383.

[21] H. Huang, W. Zhang, J. Ding, D. Stipanovic, and C. Tomlin, "Guaranteed decentralized pursuit-evasion in the plane with multiple pursuers," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, Dec 2011, pp. 4835–4840.

[22] S. Pan, H. Huang, J. Ding, W. Zhang, D. Stipanovic, and C. Tomlin, "Pursuit, evasion and defense in the plane," in *American Control Conference (ACC), 2012*, June 2012, pp. 4167–4173.

[23] Z. Zhou, W. Zhang, J. Ding, H. Huang, D. M. Stipanovic, and C. J. Tomlin, "Cooperative pursuit with voronoi partitions," *Automatica*, vol. 72, pp. 64 – 72, 2016.

[24] A. Pierson, Z. Wang, and M. Schwager, "Intercepting rogue robots: An algorithm for capturing multiple evaders with multiple pursuers," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 530–537, April 2017.

[25] L. Pimenta, M. Schwager, Q. Lindsey, V. Kumar, D. Rus, R. Mesquita, and G. Pereira, "Simultaneous coverage and tracking (scat) of moving targets with robot networks," in *Algorithmic Foundation of Robotics VIII*, ser. Springer Tracts in Advanced Robotics, G. Chirikjian, H. Choset, M. Morales, and T. Murphey, Eds. Springer Berlin Heidelberg, 2010, vol. 57, pp. 85–99.

[26] A. Pierson and D. Rus, "Distributed target tracking in cluttered environments with guaranteed collision avoidance," in *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, Dec 2017, pp. 83–89.

[27] O. Arslan and D. E. Koditschek, "Sensor-based reactive navigation in unknown convex sphere worlds," in *The 12th International Workshop on the Algorithmic Foundations of Robotics*, 2016.

[28] S. Bandyopadhyay, S. J. Chung, and F. Y. Hadaegh, "Probabilistic swarm guidance using optimal transport," in *2014 IEEE Conference on Control Applications (CCA)*, Oct 2014, pp. 498–505.

[29] D. Zhou, Z. Wang, S. Bandyopadhyay, and M. Schwager, "Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1047–1054, April 2017.

[30] L. Pimenta, V. Kumar, R. C. Mesquita, and G. Pereira, "Sensing and coverage for a network of heterogeneous robots," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, 2008, pp. 3947–3952.

[31] M. Pavone, A. Arsie, E. Frazzoli, and F. Bullo, "Equitable partitioning policies for robotic networks," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 2356–2361.

[32] J. S. Marier, C. A. Rabbath, and N. Lchevin, "Health-aware coverage control with application to a team of small uavs," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1719–1730, Sept 2013.

[33] A. Pierson, L. C. Figueiredo, L. C. Pimenta, and M. Schwager, "Adapting to sensing and actuation variations in multi-robot coverage," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 337–354, 2017. [Online]. Available: https://doi.org/10.1177/0278364916688103