Comparison of Potential and Voronoi Techniques for Pursuit-Evasion

Jennifer Cheng
Department of Mechanical Engineering
Stanford University
Stanford, USA
jwycheng@stanford.edu

Rianna Jitosho
Department of Mechanical Engineering
Stanford University
Stanford, USA
rjitosho@stanford.edu

Aditi Kumar
Department of Mechanical Engineering
Stanford University
Stanford, USA
aditik10@stanford.edu

Christopher A. Strong
Department of Electrical Engineering
Stanford University
Stanford, USA
castrong@stanford.edu

***Abstract*—Pursuit-evasion is applicable in various robotic applications such as missile guidance, search-and-rescue, disaster management, and autonomous driving. However, guaranteeing capture and minimizing capture time of evaders remains an open problem due to the difficulty of extending current approaches to diverse environments and multiple evaders. Proposed solutions in literature include using potential fields, Voronoi cells, Apollonius circles, and Partially Observable Markov Decision Processes. We implement a potential field-based and a Voronoi cell-based control policy to solve the pursuit-evasion game in a bounded convex polytope environment. We then analyze the performance and effectiveness of each method and compare the two methods. Only the Voronoi based method guarantees capture for a bounded convex polytope. We observed limitations of both algorithms: the potential field approach requires hand-tuning and lacks guarantees on capture, while the Voronoi cell approach places limits on the environment and agent speeds to guarantee capture.**

I. Introduction

Pursuit-evasion is a class of problems where one group, called pursuers, tries to track down or move within some distance of another group, called evaders [1]. The goal of the pursuers is to minimize time to capture, while the goal of the evaders is to maximize time to capture. These pursuit-evasion games can occur in bounded or unbounded environments. Many different methods exist for solving pursuit-evasion games, for instance, casting the problem as a differential game, using geometric methods, or numerically solving Hamilton-Jacobi equations.

Research in this area is useful for many applications, such as search and rescue, surveillance, disaster management, and missile guidance [2]. In search and rescue scenarios, drones can be used to look for survivors using a search strategy that assumes the survivors' policies are not known. Pursuit-evasion games can also be applied to tackle surveillance problems, for instance preventing unwanted entities from entering a certain area or observing suspicious threats. It is also applicable to disaster management, for instance monitoring the spread of a wildfire or a chemical leak. In missile guidance, the missile tracks a moving target and attempts to intercept it [3]. Other applications of pursuit-evasion games exist in robotics, autonomous vehicles, and unmanned aerial vehicles as well.

In this paper, we are interested in exploring pursuit-evasion with multiple evaders and pursuers using potential field and Voronoi cell methods in a bounded, convex polytope. We present two methods based on existing research and compare their success rates, average capture time, and how each individual method performs based on the ratio of pursuers to evaders in ability to capture an evader(s) within 60 seconds.

The paper proceeds as follows. In Section II we present related work on pursuit-evasion. In Section III we discuss the problem formulation using potential-based and Voronoi-based methods. Section IV includes a discussion of the results of the potential-based and Voronoi-based methods and a comparison of the two methods. Finally, we conclude with some key takeaways and future work in Section V.

II. Related Work

We surveyed related work in literature pertaining to pursuit-evasion problems focusing mainly on potential field and force-based methods and Voronoi-cell decomposition policies. Other related work includes Apollonius circle methods for pursuit-evasion and Partially Observable Markov Decision Processes (POMDPs).

A. Potential Method

There are numerous papers which discuss the use of potential methods for the distributed control of multi robot systems. Howard, Matarić, and Sukhatme propose a simple, yet powerful distributed method for covering an environment [4]. They create a potential-based controller by considering a virtual system in which the nodes are particles that experience friction and respond to repulsive forces away from environment obstacles and other network nodes. Thus, only local information is required, suggesting that the approach is easily scalable and robust to the loss of individual nodes.

Lynne Parker presents work on observing multiple moving targets using a fleet of robots [5]. This has applications in distributed sensing, surveillance, and warehouse or factory settings. In their approach they assume that there are several robots and several targets, with the goal of having the robots coordinate with each other to keep as many targets as possible within their sensing range at all times. Additionally, the robots can communicate with each other within some communication range. This paper takes a force based approach, where repulsive forces will be contributed by nearby robots, while attractive forces will be contributed by nearby targets. The forces are then weighted based on several rules that account for whether another robot is already tracking or sensing a target. They briefly discuss strategies around exploration if no targets are within sight as well. This goal of distributed sensing resembles our goal in the pursuit-evasion problem, although we would like to find and capture instead of simply observing a subset of the agents.

These potential field methods have been directly applied to pursuit-evasion as well. Dong et al. do just this, presenting a potential field and differential game theory based approach to pursuit-evasion with a single pursuer and a single evader [6]. This paper also considers obstacle avoidance and deadlock avoidance in the design of its strategy.

Lastly, the final approach to pursuit-evasion that we will discuss formulates planning the trajectory for the pursuer as a boundary value problem [7]. It then makes use of a time-dependent potential field in order to control the pursuer. This paper is able to provide strong guarantees about the abilities of the pursuer: no matter what the target does it will ultimately be captured by the pursuer. This approach is able to handle known, stationary obstacles.

B. Voronoi Method

Many methods using Voronoi cell control policies exist. Huang et al. present a decentralized method that uses multiple pursuers to cooperatively capture a single evader in a bounded, convex polytope in the plane by minimizing the area of the evader's Voronoi cell [1]. They show that capture can be guaranteed regardless of the evader's actions. The authors are able to show that the area of the evader's Voronoi cell is non-increasing. The evader has the same dynamics and identical maximum velocity as the pursuers. No other assumptions are made about the evader strategy or inputs. Each pursuer selects a pursuit strategy based on the observation of the evader's and other pursuer's positions.

Pierson, Wang, and Schwager expanded on the work by Huang et al. by developing Voronoi methods when multiple evaders are present [2]. They first propose a naive method in which pursuers' control policy is a summation of contributions from each evader. Since this method does not have guarantees for capture, they develop a second method which dictates that each pursuer targets the nearest evader. Their control policy drives pursuers to the shared Voronoi cell boundary of the nearest neighboring evaders. If no evaders are neighbors of the pursuer, the pursuer moves towards the nearest evader. Like in Huang et al., capture is guaranteed for any evader policy, but the optimal option for the evader is to move towards the centroid of their Voronoi cell. Together, the cooperative pursuit strategy decreases the area of the evader's Voronoi cell over time, resulting in guaranteed capture. All pursuer and evader velocities are bounded by the same maximum value.

Finally, Pan et al. implement a Voronoi-based pursuit-evasion method in a game of multiple pursuers and a single evader in a convex environment with an exit on the boundary [8]. One pursuer has the objective of defense while the other pursuers have an objective of capture. A key difference between this paper and other Voronoi based pursuit-evasion papers is that the defending pursuer implements a switching strategy that alternates between pursuit and a specialized defense strategy. The aim of the defense strategy is to prevent the evader's Voronoi cell from intersecting with the exit.

In our work we implement the policies presented in [2] as it enables pursuit-evasion with multiple evaders and pursuers. We are interested in exploring unbounded environments as discussed in [8], but leave this for future work.

C. Other Methods

While Voronoi methods decompose the environment into convex cells, Jin and Qu propose a pursuit-evasion game using Apollonius circles formed by the evader and each pursuer [9]. The evader is faster than the pursuers and the pursuers cooperate to contain the evader in a convex polygon. The evader looks for gaps between Apollonius circles if encircled to try to escape and the pursuers try to keep their Apollonius circles intersecting with their neighbors. If the evader cannot escape, it tries to prolong capture time. The pursuers try to intercept the evader, force it to change directions, or try to contain it. The authors also consider strategies for when the evader has the same speed as the pursuers and if the evader is faster. They also propose the minimum number of pursuers necessary for capture given the speed of the pursuers and the evader.

Ramana and Kothari consider a multiple pursuer, single evader scenario with holonomic constraints and an open domain [3]. Apollonius circles are used to develop an escape strategy for a high speed evader when it is enclosed by a perfectly encircled formation (PEF). A PEF is a regular polygon that entraps the evader instantaneously and needs at least 3 pursuers. However a PEF only ensures instantaneous nullification of escape paths and does not guarantee capture. The evader's strategy is to escape PEF by maneuvering to vertex or tangent points in order to create an indirect or direct gap to escape.

Von Moll et al. explores cooperation among pursuers to minimize capture time of an evader using a new geometric approach, which changes the problem into a discrete combinatoric optimization [10]. Four different types of pursuers are defined: interceptors, escorts, and two types of redundant pursuers. The control policies are based on the intersections of Apollonius circles formed by the pursuers and evader. The advantages of the method produced is that they are scalable, thus avoiding the curse of dimensionality. However, this method has not yet been extended to cases with multiple evaders.

Finally, Yi, Nam, and Sycara look at a pursuit-evasion game in an indoor environment using POMDPs [11]. POMDPs can incorporate uncertainty from when state variables cannot be observed, for instance when a target goes out of sight. The world is modeled with convex hulls and borders to guarantee that the evader is always in sight of a pursuer. Additionally, pursuers know the configuration of the environment while evaders do not. Pursuers are able to see the evader as long as there are no obstacles blocking the way. The objective of the pursuers is to find an optimal policy that minimizes the capture time of the evader, in other words, maximizing the reward for the pursuers.

III. Problem Formulation and Methods

Of the related methods reviewed, we chose to implement the following: potential-based and Voronoi-based pursuit-evasion policies. We chose to implement these two methods since the concept of attractive and repulsive potential fields is intuitive, decentralized, and simple, while Voronoi cell decomposition provides a geometric but not overly complex way to guarantee capture of evaders. In comparison, Apollonius circles and POMDPs are generally more complicated to implement. In our implementations, we favor the success of the pursuers. Next, we discuss the problem formulation, theory, and methods behind each method.

A. Potential-based

We designed our potential-based pursuit-evasion method based on the approaches in [4] and [5]. [5] uses force vectors to attract robots to goals and repel them away from other robots. The force experienced by each node due to the potential field was calculated and then node trajectories were generated using the equations of motion with a viscous friction term. Collision avoidance is implicitly encoded in this method. Some important features of this method were that it was able to achieve blanket coverage without models of the environment, location or communication between robots. This would be useful in our scenario where we could use potential-based techniques to push evader(s) away from the pursuer(s) and push pursuer(s) towards the evader(s) in changing environments without having to model the environments. We drew inspiration from these existing methods in designing the interaction forces between the pursuers and evaders.

We consider a double integrator system, with $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^2$ being the locations of robots 1 through n which live in the two dimensional plane. Each robot then behaves according to the system dynamics

$$\ddot{\mathbf{x}}_i = \frac{\mathbf{u}_i}{m}$$

where u_i is the control effort of each robot. We assume homogeneous and unit mass robots, giving us

$$\ddot{\mathbf{x}}_i = \mathbf{u}_i$$

We write this as a linear dynamical system by expanding our state to be $\mathbf{z}_i \in \mathbb{R}^4$ with

$$\mathbf{z}_i = \begin{bmatrix} \mathbf{x}_i \\ \dot{\mathbf{x}}_i \end{bmatrix}$$

We then have the system

$$\dot{\mathbf{z}}_i = A\mathbf{z}_i + B\mathbf{u}_i \qquad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We choose our control effort $\mathbf{u}_1, \ldots, \mathbf{u}_n$ in a distributed fashion inspired by the previous works on potential methods. We will define $\mathbf{F}_{ij}$ to be the force that agent i experiences as a result of agent j. Additionally, let $\mathbf{r}_{ij} = \mathbf{x}_j - \mathbf{x}_i$, the vector pointing from agent i to agent j. Let k be the number of pursuers and $p_1, \ldots, p_k$ be the indices of the pursuers. Similarly, we define $e_1, \ldots, e_{n-k}$ as the indices of the evaders. The force terms will depend on the relationship between the agents. Without loss of generality consider the interaction forces between p_1, p_2 and e_1, e_2. For constants k_{pp}, k_{pe}, k_{ep}, and k_{ee} all greater than or equal to 0 we will define interaction forces

$$\mathbf{F}_{p_1p_2} = -\frac{k_{pp}}{\|\mathbf{r}_{p_1p_2}\|_2^3}\mathbf{r}_{p_1p_2}$$

$$\mathbf{F}_{p_1e_1} = \begin{cases} \frac{k_{pe}}{\|\mathbf{r}_{p_1e_1}\|^3}\mathbf{r}_{p_1e_1} & e_1 = \arg\min_{e_j} \|\mathbf{r}_{p_1e_j}\| \wedge \|\mathbf{r}_{p_1e_1}\| \leq r \\ \mathbf{r}_{p_1e_1} & e_1 = \arg\min_{e_j} \|\mathbf{r}_{p_1e_j}\| \wedge \|\mathbf{r}_{p_1e_1}\| > r \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{F}_{e_1p_1} = -\frac{k_{ep}}{\|\mathbf{r}_{e_1p_1}\|^3}\mathbf{r}_{e_1p_1}$$

$$\mathbf{F}_{e_1e_2} = -\frac{k_{ee}}{\|\mathbf{r}_{e_1e_2}\|^3}\mathbf{r}_{e_1e_2}$$

$F_{p_1e_1}$ is designed to have the pursuer only be affected by the interaction force from the closest evader. If the pursuer is greater than a distance r away from the evader, then the policy reduces to Linear Consensus Protocol (LCP). This is because the effect of the evader's attractive potential field drops off as $1/r^3$ so pursuers are unable to track the evader effectively when they are too far away. Thus, LCP is used to bring the pursuer close to the evader and then once the pursuer is within a radius r, the potential field kicks in to lock the pursuer into the evader's location. This trade off works since the force from LCP is stronger at larger distances and the potential field is stronger at closer distances. This hypothesis is verified in the Results section (Section IV.A).

In our experiments, we had $k_{pp} = 0$ and $k_{ee} = 0$ to simplify the problem by ignoring pursuer-pursuer and evader-evader interactions.

We now also presume our environment consists of m walls described by hyperplanes $W_i = \{\mathbf{c}_i^T\mathbf{x} = \mathbf{b}_i\}$ for $i = 1, \ldots, m$. Let $dist(\mathbf{x}, W_i)$ be defined as the Euclidean distance from the point to the wall

$$dist(\mathbf{x}, W_i) = \inf_{\mathbf{y} \in W_i} \|\mathbf{x} - \mathbf{y}\|_2$$

Note that this can be efficiently calculated by projecting x onto the hyperplane. If the walls are aligned with the principal coordinate axes, this distance becomes just a subtraction of the appropriate coordinate of x from the position of the wall. We would like a force from the wall to push the agents away once they are within some threshold distance. We can then write the forces from the walls for some threshold distance d as

$$\mathbf{F}_{W_i} = \begin{cases} 0 & dist(\mathbf{x}, W_i) \leq d \\ -\frac{k_{wall}}{\|dist(\mathbf{x}, W_i)\|^3}\mathbf{c}_i & \text{otherwise} \end{cases}$$

The parameter $k_{wall} \geq 0$ was tuned so as to keep the agents from passing through the wall. With an upper limit on the interaction forces between agents a parameter that will always prevent the agents from leaving the area could be analytically determined, although we did not do so in this work.

Combining the interaction and wall forces, we can define the control law for an individual agent based on the sum of its interactions with its neighbors to be

$$\mathbf{u}_i = \sum_{j \in \mathcal{N}_i} \mathbf{F}_{ij} + \sum_{j=1}^{m} \mathbf{F}_{W_i}$$

Where the expression for the force $\mathbf{F}_{ij}$ will depend on whether i and j are evaders or pursuers as described above. In our experiments, all agents are neighbors with one another in an undirected, fully connected graph. Plugging this control law into our dynamics, we arrive at the closed loop system dynamics

$$\dot{\mathbf{z}}_i = A\mathbf{z}_i + B(\sum_{j \in \mathcal{N}_i} \mathbf{F}_{ij} + \sum_{j=1}^{m} \mathbf{F}_{W_j})$$

Note that the interaction forces are decentralized, as an agent only needs to know the location of its neighbors. Similarly, the wall forces only rely on knowing the distance to and the direction of walls within the threshold d.

B. Voronoi-based

For our Voronoi-based implementation, we utilize the control policies derived by Pierson, Wang, and Schwager [2]. In this formulation, we decompose the environment into Voronoi cells as exemplified in Fig. 1, and use its geometry to determine the control inputs for evaders and pursuers. The control input is the agent velocity normalized to a magnitude of one. As in [2], we set the evader policy to go towards the centroid of its cell:

$$\mathbf{u}_e = \frac{\mathbf{C}_{V_e} - \mathbf{x}_e}{\|\mathbf{C}_{V_e} - \mathbf{x}_e\|} \quad (1)$$

where $\mathbf{C}_{V_e}$ is the centroid of the evader's cell and $\mathbf{x}_e$ is the position of the evader.

Meanwhile, the pursuer policy is to go towards the centroid (midpoint in 2D), of a pursuer-evader boundary:

$$\mathbf{u}_p = \frac{\mathbf{C}_{b_{pe}} - \mathbf{x}_p}{\|\mathbf{C}_{b_{pe}} - \mathbf{x}_p\|} \quad (2)$$

where $\mathbf{C}_{b_{pe}}$ is the midpoint of the shared boundary of the pursuer and evader, and $\mathbf{x}_p$ is the position of the pursuer. In the case where a pursuer shares boundaries with multiple evaders, it will choose the boundary midpoint with the nearest evader. In the case where a pursuer does not share any boundaries with an evader, it will drive directly towards the nearest evader.

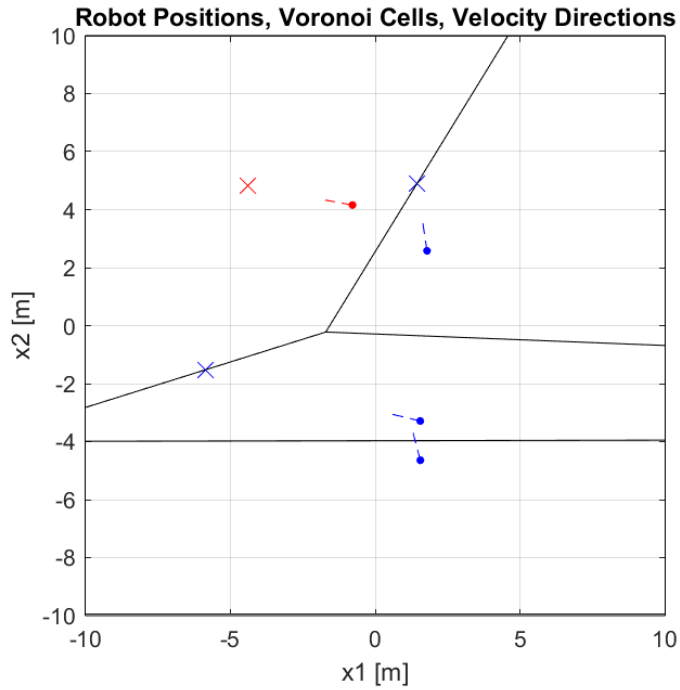


Fig. 1: Illustration of Voronoi method. The Voronoi decomposition is shown with black lines. Evaders (red) drive towards their cell centroid (red x). Pursuers (blue) with a neighboring evader drive towards the midpoint of the shared cell boundary (blue x). Pursuers without a evader neighbor drive directly towards the nearest evader.

A key takeaway from [2] is that this pursuer policy guarantees capture of all evaders, regardless of evader policies, through an area-minimization strategy. It can be shown that the area of the evader cell is non-increasing for the single-evader scenario. However, the evader policy selected is best for the evader (in terms of delaying capture) [2].

IV. Results

To characterize and compare the two methods, we implemented both control methods in MATLAB and ran multiple pursuit-evasion simulations.

For all simulations agents were enclosed in a 20 meter by 20 meter square (number of walls, $m = 4$). The capture radius is defined to be 0.2 meters in all scenarios. The capture time is defined to be the time it takes for all evaders to be caught. For Monte Carlo simulations, success rate is defined as the number of trials of successful capture (all evaders caught within 60 seconds) over the number of total trials. In this section, we talk about the general behavior of each method individually and then compare the two methods in terms of average capture time and success rate with respect to pursuer-evader ratio.

A. Potential-based Method Results

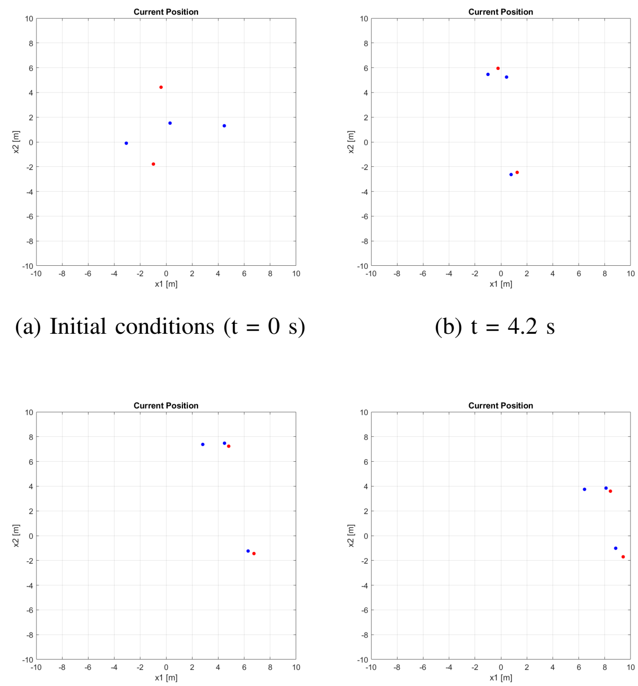
First, we will verify that the potential method combined with LCP performs better than only using potential fields or only using LCP. Evaders always model the pursuers as repulsive potential fields in all cases. Table I shows a comparison of average capture time and success rate using 3 pursuers and 2 evaders in a Monte Carlo simulation running 100 trials with random initial conditions. It is clear that LCP is not able to catch the evader at all except when it gets lucky. In 2 trials the initial conditions happened to result in close pursuer and evader locations and short capture times. This is as expected since using LCP, the pursuers keep chasing after the evaders naively and the force decreases as the pursuers get closer to the evader. However, LCP is good at following the evader. On the other hand, if all pursuers model the closest evader as an attractive potential field, then the success rate is 0.37. However, since the potential field drops off as $1/r^3$, the pursuers that are far away from the evader have difficulty catching up. Using only potential field models results in better following of the evader if the pursuer is close to the evader to start with. In Table I we see that using LCP to get the pursuer to a region close to an evader and then switching to a model of the evader as a potential field helps lock the pursuer onto the evader's location. This combined control policy results in an increase in success rate to 0.87. However, the increase in success rate is at the expense of average capture time.

	Potential	LCP	Potential + LCP
Success Rate	0.37	0.02	0.87
Avg. Capture Time [s]	13.815	0.61	27.27

TABLE I: The joint LCP and potential field control policy outperforms using only LCP or only a potential field control policy.

Fig. 2 shows a simulation with 3 pursuers and 2 evaders using the potential-based method. The maximum magnitudes of velocity and acceleration are 1 m/s and 10 m/s^2 respectively. The evaders are depicted by the red dots and the pursuers by the blue dots. The pursuer and evader initial conditions are randomized in the grid space, shown in Fig. 2a. In Fig. 2b, the pursuers each select their nearest evader and heads toward it. The pursuers continue chasing the evaders in Fig. 2c while the evaders are repelled by all the pursuer forces. Finally the lower-most evader is caught and the tailing pursuer now starts heading toward the remaining evader that has not been caught in Fig. 2d. Shortly after, the second evader is caught and the capture time is around 15 seconds.

The trajectory plot for all evaders and pursuers for the simulation in Fig. 2 is shown in Fig. 3. The solid blue and red dots are the initial conditions for the pursuers and evaders, respectively. The blue crosses and red crosses indicate the final positions of the pursuers and evaders respectively at the end of the simulation (or when the evader stops moving once it is captured). The blue dotted line is the pursuer trajectory over time and the red solid line is the evader trajectory over time.



(a) Initial conditions (t = 0 s)

(b) t = 4.2 s

(c) t = 9.5 s

(d) t = 13.3 s

Fig. 2: 3 pursuers (blue), 2 evaders (red) simulation. Pursuers head to the nearest evader that is not caught until all evaders are captured. The capture radius is defined to be 0.2 meters.

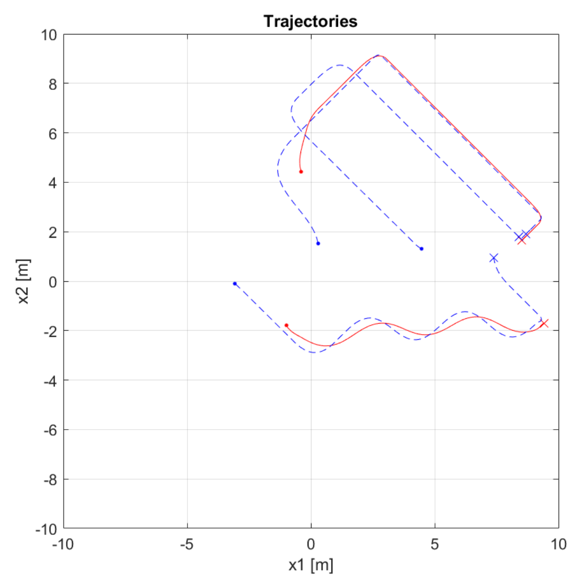


Fig. 3: Agent trajectories for pursuers (blue) and evaders (red) using the potential-based method. Starting positions are dots and final positions are x's.

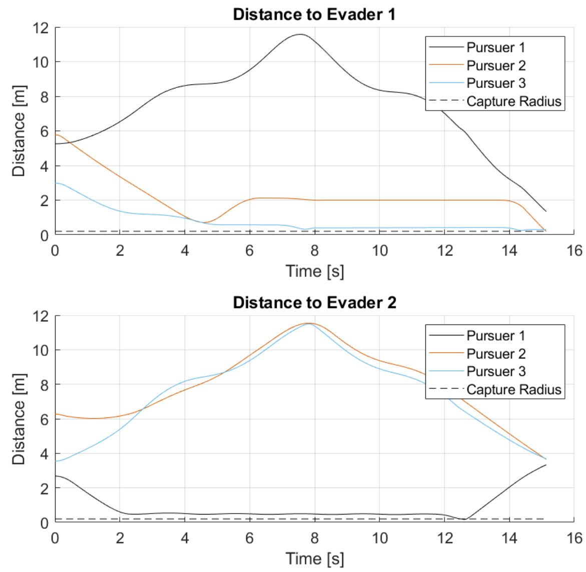


Fig. 4: Distance from pursuers to evaders over time. An evader is captured when a pursuer is within 0.2 meters of it. Evader 1 is captured around 15 seconds and Evader 2 is captured around 13 seconds.

For the same simulation, the Euclidean distance from each pursuer to each evader is shown in Fig. 4. The first evader is chased by two pursuers (Pursuers 2 and 3) while the second evader is chased by one pursuer. The second evader is caught just before 13 seconds by Pursuer 1. Pursuer 1 then changes direction once it has captured the second evader and heads toward the first evader. Pursuer 2 captures the evader around 15 seconds.

In summary, the potential-based method is still quite naive since the pursuer policy is to simply go towards the evader. The control policy does not allow the pursuer to strategically intercept the evader or use formations to form a trap around the evader. In general, the pursuers tend to catch the evaders near the wall where the evader has to slow down to change directions. The pursuers and evaders are unable to take into account the shape of the environment in their control policies. These are all areas for improvement in the future.

B. Voronoi-Based Method Results

We demonstrate the Voronoi method by running a simulation with four pursuers and two evaders in the same square environment, and a capture radius of 0.2 meters. The overall trajectories of all agents are shown in Fig. 5, and snapshots at specific times are shown in Fig. 6. From these plots, we see that different pursuers initially target different evaders, and when an evader was captured, its pursuers switch to focus on the remaining un-captured evader. We also tracked the distances between pursuers and evaders (Fig. 7) as well as the area of evader Voronoi cells (Fig. 8). In this case, since there is more than one evader, the area is not non-increasing through time. However, over the full simulation from 0 seconds to time of capture, each evader's Voronoi cell decreases to an area to 0 from an initial non-zero value. From these plots, we see pursuers 1 and 2 capturing evader 2 around 16 seconds, and then making their way towards evader 1. Finally, we see capture of evader 1 around 21 seconds.

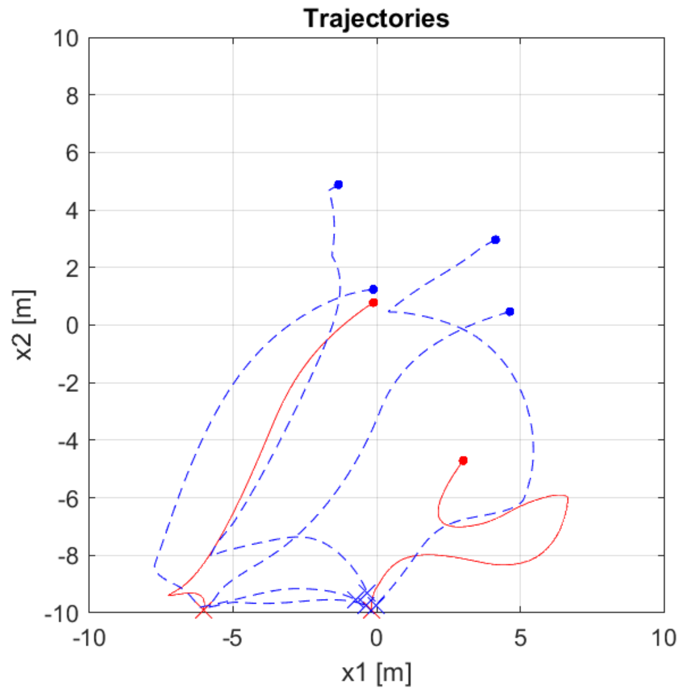


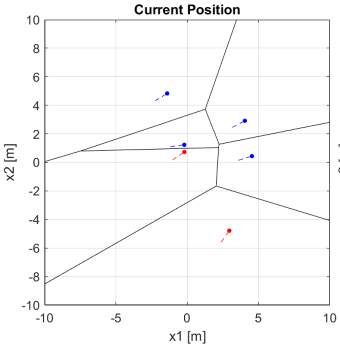
Fig. 5: Agent trajectories for pursuers (blue) and evaders (red) using Voronoi-based control policies. Starting positions are dots and final positions are x's.

C. Comparison of Potential-based and Voronoi-based Methods

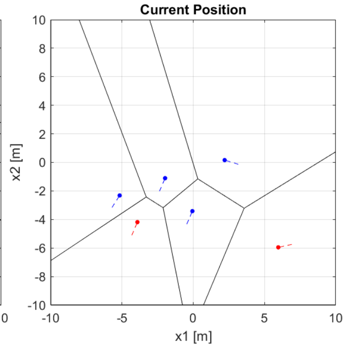
Two metrics were selected to compare the performances of potential and Voronoi-based methods.

The first metric was success rate. As shown in Fig. 9, the capture success rate of Voronoi and potential methods is plotted against the number of pursuers to evaders ratio. These are the results of a Monte Carlo simulation with 100 trials at each pursuer-evader ratio for the potential method and 25 trials at each ratio for the Voronoi method. The difference in number of trials is due to the longer computation time required to run each simulation of the Voronoi method. Each simulation was allowed to run up to 60 seconds and a constant bounded environment was used, i.e., a 20 meter by 20 meter closed square, for all simulations.

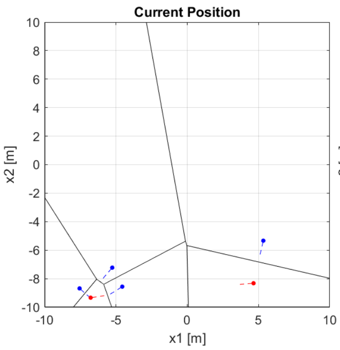
Since the Voronoi method guarantees capture, the success rate of all trials, except one, is 100%. The only case where the Voronoi method is unable to achieve 100% capture is when the pursuer to evader ratio is less than 1. When there are fewer pursuers than evaders, it takes longer for the pursuers to catch the evaders. In this case, they were unable to capture all evaders within the 60 second time limit. Failure to capture the evaders in the allotted time could also have occurred if agents



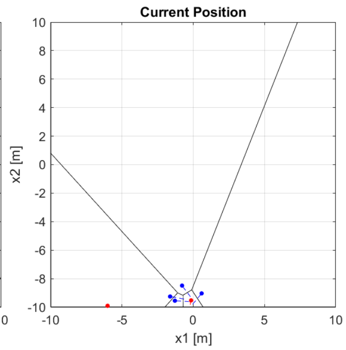
(a) Initial conditions (t = 0 s)



(b) t = 6.4 s

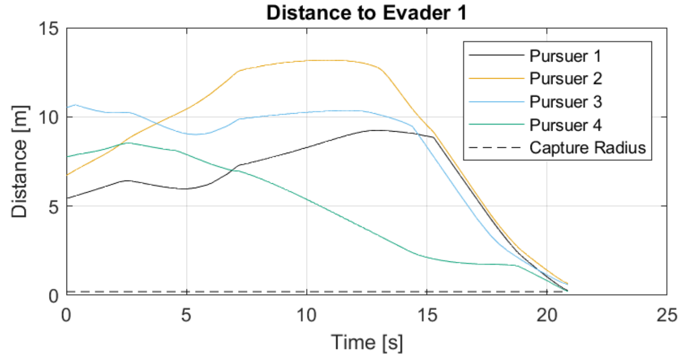


(c) t = 9.5 s



(d) t = 20.9 s

Fig. 6: Snapshots from a simulation with 4 pursuers (blue) and 2 evaders (red) using Voronoi-based control policies. The capture radius is defined to be 0.2 meters.



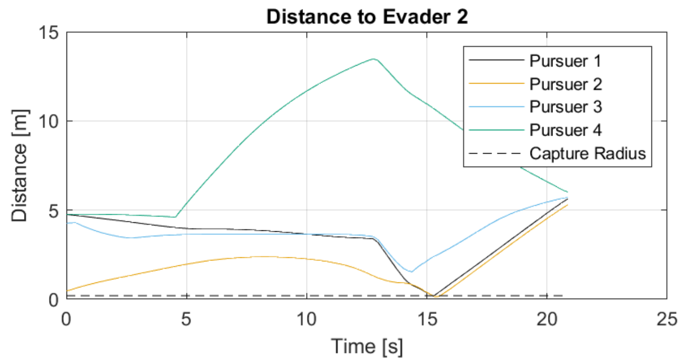


Fig. 7: Distances of pursuers to evaders during a simulation with Voronoi-based control policies. The capture radius is set to 0.2 m, so pursuers 1 and 2 capture evader 2 around 16 seconds and pursuers 1 and 4 capture evader 1 around 21 seconds.

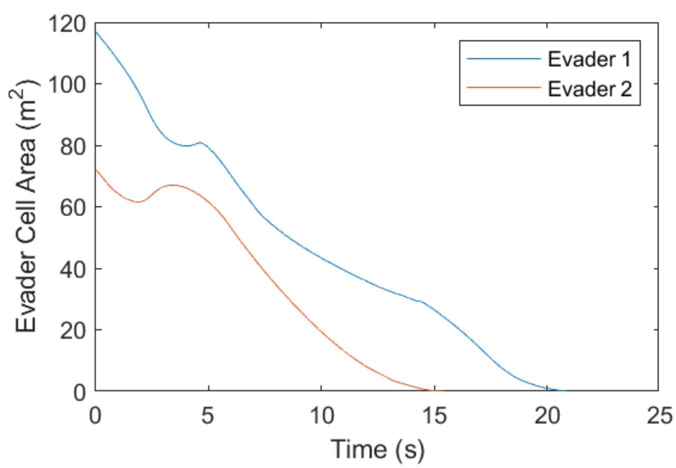


Fig. 8: Evader cell area during a simulation with Voronoi-based control policies.

were placed in a larger environment or if initial conditions favored the evaders. Thus, only a 72% success rate is achieved.

When the number of evaders is kept constant, the potential method's success rate increases as the number of pursuers is increased. This can be seen in the increase in success rate from 1:1 to 6:1 in the left half of the graph. Although 1:1 and 6:6 represent the same ratios of pursuers to evaders, the performance of the potential method degrades as more pursuers and evaders are added as many more force interactions need to be accounted for which would increase the time required for capture to greater than 60 seconds. The same trend is seen between the ratios of 5:3 and 10:6. Similar to the Voronoi method, the performance of this method is the worst at a ratio of 3:4 where only 27% of the trials were successful in capturing all evaders due to the increased interactions between agents and the limited capture time of 60 seconds.

Comparing the two methods in terms of success rate, it is clear that the Voronoi method due to its guarantee on capture is far superior to the potential method in capturing any number of evaders with any number of pursuers. The potential method is sensitive to pursuer-evader ratios and performs significantly worse than the Voronoi method in the 3:4 ratio case.

The second metric used to compare the two methods was average capture time, shown in Fig. 10. This figure was also generated with the results of a Monte Carlo study with 100 trials of the potential method per pursuer-evader ratio and 25 trials of the Voronoi method for each ratio. Each trial was run up to 60 seconds in the same bounded 20 meter by 20 meter square environment.

The first plot shows variation in capture time with pursuer-evader ratio for the potential method. Following the trend for success rate, as the number of pursuers is increased while the

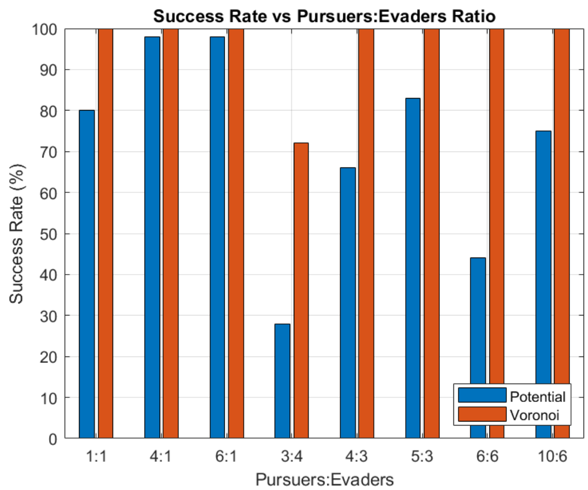


Fig. 9: Success rate vs. pursuer to evader ratio for Voronoi and potential methods (results of a Monte Carlo study).

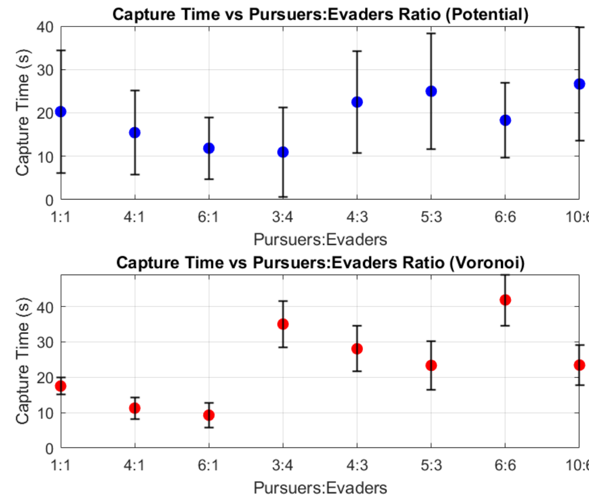


Fig. 10: Average capture time vs pursuer:evader ratio for Voronoi and potential methods (results of a Monte Carlo study). The error bars show two standard deviations around the mean for each simulation.

number of evaders is kept constant, capture time decreases from 20 s to 11 s as shown in the left half of the plot. Standard deviations, shown by the error bars, also decrease as average capture time decreases. The highest capture time and greatest variability in capture time is seen in the 10:6 case. An interesting trend is that the average capture times for the same pursuer-evader ratio are within 10% of each other which is seen with the average capture times of 20 s and 18 s for 1:1 and 6:6, as well as 25 s and 26 s for 5:3 and 10:6.

The second plot shows variation in capture time with pursuer-evader ratio for the Voronoi method. Between the Voronoi and potential methods, the average capture times are lower for the Voronoi method than the potential method in the single-evader case. The variability in capture times is also lower for all ratios in Voronoi than in potential. However, this may be because only 25 trials of the Voronoi-based policy were run in each Monte Carlo simulation, while 100 trials were run for the potential-based policy. Although the potential method did much poorly than the Voronoi method in the 3:4 pursuer-evader ratio case, the average capture time for Voronoi is much greater than for potential at this ratio. This is because the Voronoi method guarantees capture and each trial took a longer time to capture all evaders due to the fewer number of pursuers, while the potential method may have gotten lucky due to the initial conditions.

V. Conclusion

In this paper, we implemented two different types of approaches for solving the pursuit-evasion problem and analyzed their results and compared the two techniques with one another. A potential-based method combined with LCP is effective in capturing the evader in under 60 seconds. However, the spread of average capture time is large. This method is simple, decentralized, and intuitive, but there are no guarantees for capture. However, the agents behave rather naively and are unable to use information about their environment or other agents to plan strategically. The potential-based method also resulted in some oscillatory behavior that could be due to the interactions of various potential fields pushing and pulling on the agents. One downside of this method is that the potential field strengths must be tuned and a physical interpretation of these gains is an area for future work.

The Voronoi-based method ensures guaranteed capture in finite time using an area-minimization strategy. [2]. Using this method, we found that the pursuers were almost always able to capture all evaders within 60 seconds. Even if the pursuers did not capture the evaders within 60 seconds, they were still guaranteed to capture the evaders at some finite time past 60 seconds. While the Voronoi-based method is more strategic than the potential-based method in how it captures the evader by using the geometry of the Voronoi-decomposed environment, the guaranteed capture is specifically for a closed and bounded environment.

In general, the Voronoi-based method does better than the potential-based method in terms of success rate. This is as expected since for the Voronoi-based method there is guaranteed capture. In all Monte Carlo simulations examined, the Voronoi-based method was able to capture the evader(s) 100% of the time except when the evaders outnumber the pursuers. This is because it took longer to capture all of the evaders within 60 seconds, which was our time-constraint for solving the pursuit-evasion problem. The Voronoi-based method has a smaller spread of capture time compared to potential-based method and has faster capture times for all of the single evader scenarios.

There are several directions for future work from this paper. Some extensions include giving the evaders a head start over pursuers and exploring how this would affect capture time. Different types of environments could be tackled in other extensions of this work including a 3-dimensional box. The agents could be modelled with more realistic sensing and communication models which could possibly affect guarantees on capture in the Voronoi method and the success rate and capture time of the potential method. Other interesting extensions to this would be having heterogeneous agents with different sensing capabilities, and varying the maximum speed limits on these agents. Obstacles could also be added to the environment and a collision avoidance algorithm could be implemented with pursuit-evasion strategies. The potential and Voronoi methods could also be mixed in simulations where the evader could be implementing a potential policy while the pursuer implements a Voronoi policy. Additionally, other agent models can be added such as the unicycle model and non-holonomic constraints. Finally, other related work on Apollonius Circles could be explored as prospective policies to be mixed with the Voronoi and potential methods to determine an optimal policy for each type of agent.

Acknowledgments

This paper was completed as part of the requirements of AA 277: Multi-Robot Control and Distributed Optimization, a class offered in the Aerononautics and Astronautics department at Stanford University. We would like to thank Professor Mac Schwager and the teaching assistants, Keiko Nagami and Jun En Low for organizing this class and supporting us throughout this project.

References

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

[2] 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, 2016.

[3] M. V. Ramana and M. Kothari, "Pursuit-evasion games of high speed evader," *Journal of intelligent & robotic systems*, vol. 85, no. 2, pp. 293–306, 2017.

[4] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Distributed Autonomous Robotic Systems 5*. Springer, 2002, pp. 299–308.

[5] L. E. Parker, "Distributed algorithms for multi-robot observation of multiple moving targets," *Autonomous robots*, vol. 12, no. 3, pp. 231–255, 2002.

[6] J. Dong, X. Zhang, and X. Jia, "Strategies of pursuit-evasion game based on improved potential field and differential game theory for mobile robots," in *2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control*. IEEE, 2012, pp. 1452–1456.

[7] A. A. Masoud, "A boundary value problem formulation of pursuit evasion in a known stationary environment: a potential field approach," in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3. IEEE, 1995, pp. 2734–2739.

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

[9] S. Jin and Z. Qu, "Pursuit-evasion games with multi-pursuer vs. one fast evader," in *2010 8th World Congress on Intelligent Control and Automation*. IEEE, 2010, pp. 3184–3189.

[10] A. Von Moll, D. W. Casbeer, E. Garcia, and D. Milutinović, "Pursuit-evasion of an evader by multiple pursuers," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2018, pp. 133–142.

[11] S. Yi, C. Nam, and K. Sycara, "Indoor pursuit-evasion with hybrid hierarchical partially observable markov decision processes for multi-robot systems," in *Distributed Autonomous Robotic Systems*. Springer, 2019, pp. 251–264.