Decentralized Control for Intelligent Robot System to Avoid Moving Obstacles

Yongsub Lim, Kyomin Jung*
Computer Science
KAIST
Daejeon, Korea
yongsub@kaist.ac.kr, kyomin@kaist.edu

***Abstract*—We propose a novel control law to deploy a decentralized multiagent mobile intelligent robot system in a dynamic environment containing moving obstacles. Since a collision to an obstacle causes failure of the robot and may break the whole system, obstacle avoidance is one of the fundamental task for the mobile robot system. Potential based approaches have been widely applied to deploying a mobile robot system for various goals, including obstacle avoidance, coverage maximization and target tracking. However, previous methods consider only a snapshot at each time to decide the dynamics of a robot. Therefore they have limitations in dynamic environments, possibly leading to frequent collisions to obstacles. In contrast, we show that by considering the velocities of the obstacles, the number of collision can be significantly reduced. In our method each robot records a few past histories of sensed obstacles, and based on the record the robot predicts a future trace of the obstacle. We introduce a new dynamics for robots by combining the current state information with the prediction. By extensive experiments, we compare our method with others and show that by our method robots rarely collide with moving obstacles while keeping the potential low.**

***Keywords*-Multiagent System, Intelligent System, Obstacle Avoidance, Decentralized Control**

I. Introduction

In recent years, designing control law for decentralized multiagent mobile intelligent robot systems has been actively studied [1, 2, 3, 4, 5]. A number of applications, including mobile sensor network deployment and swarm formation require highly autonomous and decentralized control law. For example, consider that a mobile robot system is deployed in a complex environment such as city, or in a wild environment such as desert and forest. In such environments, a robot belonging to the system cannot sense the whole environment, and its sensing and communication ranges are limited. Also a system consisting of many individuals should be robust to failure of a few individuals. Such limitations encourage design of simple and highly decentralized control law for mobile robot systems.

Fundamental goals of decentralized robot systems include coverage maximization [1, 6], target tracking [7], formation [5], and obstacle avoidance [8]. Notice that a mobile robot system may have some of those goals simultaneously. For example, a system may chase a target while maximizing the covered area. Among them, obstacle avoidance is one of the essential goals since if it is not accomplished, robots colliding with the obstacle may be broken and may even lead to a failure of the whole system. Thus, obstacle avoidance should be considered as one of essential tasks in any type of deployment a mobile robot system.

In many real world problems, the environment is dynamic: the boundary may change, or there can be moving obstacles in the environment. In this paper, we propose a novel decentralized control law for obstacle avoidance based on trace prediction in such a dynamic environment.

A. Previous Work

Obstacle avoidance has been widely studied for multiagent mobile robot system [9, 10, 11, 4, 12]. Potential based approaches are popular for various goals, including avoiding static or dynamic obstacles. This type of approaches considers that there is an artificial force filed on the area where the system operates. Robots in the system move according to the force arisen from this field, and eventually the system of robots go to equilibrium points which are essentially a Nash equilibrium of the corresponding non-cooperative game among the robots.

Using a potential based approach, it is shown that robots can reach the destination when there are static obstacles in a course [13]. Even in an environment where static obstacles densely present in the course, it is shown that the robot's motion is smooth and continuous while advancing to the destination [14, 15]. Zhang et al. considered obstacle avoidance while keeping formation and following a moving leader [16]. Using genetic algorithms, potential functions are computed for path planning when moving obstacles exist [4]. Yan et al. studied to avoid moving obstacles while tracking target [9]. Recently, [17, 12] applied the potential based approach to real-time strategy games in which obstacle avoidance and navigation are essential parts.

Most of the above methods use the following potential or its variation [1]. This method focuses on deploying mobile sensors so that covered area is maximized in an unknown environment (while possibly performing other tasks). The idea is motivated from the electrostatic potential, and a

*This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(2012032786)

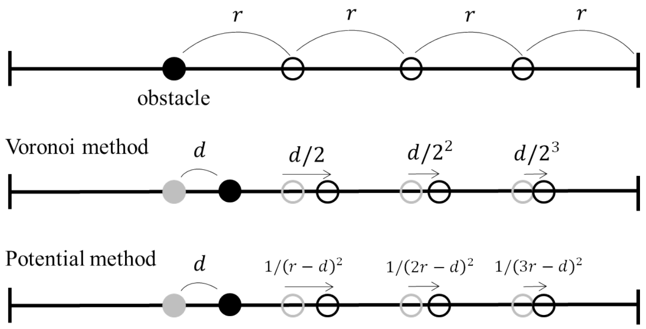


Figure 1. Comparison of influence of a moving obstacle to dynamics of robots. The effect of a moving obstacle decays exponentially on distance for the Voronoi method, but polynomially for the potential method.

potential for each robot is defined as follows.

$$U_i = \sum_{j \in \mathcal{N}_i} \frac{\beta_i}{||p_i - p_j||} + \sum_{j \in \mathcal{M}_i} \frac{\beta_j}{||p_i - q_j||}, \qquad (1)$$

where p_i is the location of a robot i, q_j is the location of an obstacle j, $\mathcal{N}_i$ and $\mathcal{M}_i$ are sets of robots and obstacles, respectively sensed by robot i, β_i is the weight between robots, β_j is density of obstacles, and $|| \cdot ||$ denotes the Euclidean norm. In [1], they show through experiments that by using a gradient of U_i over p_i as the dynamics of the robot i, high quality coverage can be obtained even in an unknown environment having a complex boundary. We refer this type of control laws as the potential (based) method in this paper.

Another very popular control law is a method based on the Voronoi cell. In this method, each robot moves to the centroid of its Voronoi cell. This is called a centroidal Voronoi tessellation [18, 19], and we refer this type of control laws as the Voronoi method. This method is widely used especially for coverage maximization. It is known that in this method the robots move along the gradient direction of the following potential.

$$H = \sum_{i=1}^{n} \int_{W_i} f(||s - p_i||)\phi(s) \; ds, \qquad (2)$$

where W_i is the Voronoi cell for a robot i, and ϕ represents the density at s. Cortés et al. show that the locations of robots in a local minimum of H are weighted centroids of the Voronoi cells [18]. This type of control laws is also used for maintaining connectivity of relay wireless robots [20]. In the Voronoi method, movement of a robot depends only on its neighbors. It means that in this approach, a robot essentially uses only local information. Therefore even if a robot has limited sensing range, a final state of the system is not much different from the case when the robots have full sensing range. Due to this advantage, the Voronoi method has been also very popular for decentralized robot systems with limited sensing range.

B. Drawback of Previous Work

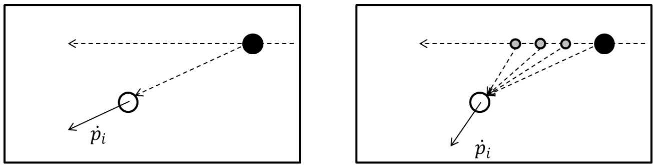
The problem we consider is to design a decentralized control law for a mobile robot system in an unknown environment, by which the robots efficiently avoid moving obstacles. In this section, we explain drawbacks of the Voronoi and naive potential method applied to this problem.

Notice that in the Voronoi method, dynamics of each robot depends only on the direct neighbors. This property makes the system stable to small change of environments. However, this property may not be an advantage in a dynamic environment with moving obstacles. We observed that in the Voronoi method, the effect of movement of an obstacle decreases exponentially on the number of robots between the robot and the obstacle, which is approximately proportional to their distance. Figure 1 shows how a robot is affected by an approaching obstacle. When there are k robots between a robot and an obstacle, if the obstacle moves 1 meter closer per unit time, the robot runs away only $2^{-(k+1)}$ meters during that time. This is because even a robot is close to an obstacle, if its Voronoi cell is not adjacent to that of the obstacle, the dynamics of the robot is not directly affected. Instead, the effect is propagated to the robot indirectly through the intermediate robots.

On the other hand, the potential method is more suitable for obstacle avoidance since repelled force from an obstacle is more directly applied to the robot. In this case, the effect decreases polynomially on their distance, not exponentially. Figure 1 shows how the movement of an obstacle affects the robots in this case.

However, the naive potential method is still not enough for our goal. It only takes the snapshot into account at each time, and the dynamics of the system is determined based on only the current state, not on the velocity of the obstacles. If the speed of an obstacle is fast, the collision risk becomes higher. Hence, the robot should move more orthogonal to the direction of the obstacle's movement. Notice that this is impossible by just using snapshots, and hence we need an additional strategy to avoid the obstacle in relatively short time.

Thus, we focus on developing a more suitable method which utilizes movement information of the obstacles. In our method, each robot first predicts a future trace of a sensed moving obstacle using a few past histories. Combining the prediction with the current state, the robot computes the repelling force from the obstacle. Figure 2 shows the effect of the prediction compared with the naive potential method. Note that by considering the prediction, the robot tries to be far from the future trace of the obstacle, not just from the obstacle's current location. In other words, the robot receives more force orthogonal to the predicted direction of the obstacle. This effect enables the robot to avoid the obstacle before it reaches too close. In Section IV, through extensive experiments, we show that in our method, collision rarely



Naive Our method

Figure 2. The effect of the prediction. In our mehtod, the robot receives larger force in the orthogonal direction to the obstacle's movement compared with the naive potential method.

occurs unlike the Voronoi and naive potential methods. We note that our method can be also applied to the target tracking by using the prediction of the target movement and similar attraction force from the target as discussed in Section 3.

II. Preliminaries

In this section, we introduce a general framework of potential based methods to deploy a mobile robot system. A mobile robot system is a set of mobile robots $\mathcal{R} = \{1, \ldots, n\}$ and we denote its location by $p_i \in \mathbb{R}^d$. In general, $d = 2$ or 3. An environment consists of a set of obstacles $\mathcal{O}$ and the boundary $\mathcal{B}$ where $\mathcal{V}$ can be considered as a collection of static obstacles. We denote the center of mass of an obstacle $j \in \mathcal{O}$ by q_j, and consider an obstacle as a point at its center of mass. Then, the potential of the system is defined as follows.

$$V = \frac{1}{2} \sum_{i \in \mathcal{R}} \left(V_{i;1} + V_{i;2} + V_{i;3} \right), \tag{3}$$

where

$$V_{i;1} = \sum_{j \in \mathcal{R} \setminus i} \frac{w_{ij}}{||p_i - p_j||^h}, \quad V_{i;2} = \sum_{j \in \mathcal{O}} \frac{w_{ij}}{||p_i - q_j||^h},$$
$$V_{i;3} = \int_B \frac{w_i(s)}{||p_i - s||^h} \, ds.$$

Here, w_{ij} is the weight between $i \in \mathcal{R}$ and $j \in \mathcal{R} \cup \mathcal{O}$, and $w_i(s)$ is the density function of the boundary. Here, $V_{i;1}$ is a potential among the robots, $V_{i;2}$ is a potential between the robots and the obstacles, and $V_{i;3}$ is that between the robots and the boundary. In general $h > 0$ and can be any positive real number. In this paper, we consider the most common case of $h = 1$ for ease of explanation. However, our method can be applied to any $h > 0$. Then, the dynamics of a robot i is defined as the gradient of V for the coordinate p_i.

$$\dot{p}_i = \frac{dp_i}{dt} = -\frac{\partial V}{\partial p_i} = -\frac{\partial V_{1;i}}{\partial p_i} - \frac{\partial V_{2;i}}{\partial p_i} - \frac{\partial V_{3;i}}{\partial p_i}, \tag{4}$$

where

$$-\frac{\partial V_{1;i}}{\partial p_i} = \sum_{j \in \mathcal{R} \setminus i} \frac{w_{ij}(p_i - p_j)}{||p_i - p_j||^3}, \quad -\frac{\partial V_{2;i}}{\partial p_i} = \sum_{j \in \mathcal{O}} \frac{w_{ij}(p_i - q_j)}{||p_i - q_j||^3},$$
$$-\frac{\partial V_{3;i}}{\partial p_i} = \int_B \frac{w_{is}(p_i - s)}{||p_i - s||^3} \, dj.$$

Intuitively, moving in the gradient direction to minimize (3) can be interpreted that each robot regards all the robots, obstacles, and the boundary as electrons which cast repelling force to each other.

III. Our Method

Although (4) works well for a static environment [1], it may fail for a dynamic environment as we have explained in Section I-B. In this section, we describe our method to resolve this problem. First we assume full sensing range for every robot, and we later explain the limited sensing range case.

For convenience, we introduce the following notations. We assume that there is a set of virtual fixed obstacles on the boundary $\mathcal{B}$ (possibly continuous). Let $\mathcal{I} = \mathcal{R} \cup \mathcal{O} \cup \mathcal{B}$, then we call each $i \in \mathcal{I}$ an object. The location of an object $i \in \mathcal{I}$ is denoted by p_i. Then, we can rewrite (3) and (4) as follows.

$$V = \frac{1}{2} \sum_{i \in \mathcal{R}} \sum_{j \in \mathcal{I} \setminus i} \frac{w_{ij}}{||p_i - p_j||}, \tag{5}$$

$$\dot{p}_i = \sum_{j \in \mathcal{I} \setminus i} \frac{w_{ij}(p_i - p_j)}{||p_i - p_j||^3}, \quad \forall i \in \mathcal{R}. \tag{6}$$

In the system we consider, each robot can record short term histories of sensed obstacles. Then based on the records, each robot can predict a future trace of the obstacle. If we have prior knowledge on the obstacle or can successively record history of the obstacle for a long time, we can adopt a typical prediction method such as Kalman filter. However, because a robot can record only a short history for an obstacle, a simple method can show better performance in quality and cost. For example, a future trace can be predicted by a polynomial. Especially, a parabola is suitable for our case: it is the simplest polynomial which can deal with a curve motion.

Thus, we simply predict the trace of an obstacle by a parabola using the past two records and the current location of the obstacle. Note that even we have three records including the current location, a corresponding parabola is not uniquely determined due to rotations. Let $p^{(0)}, p^{(-1)}, p^{(-2)}$ be the current location and the past two records. A parabola we suggest for the prediction is determined as follows: $p^{(-1)}$ becomes the bisector of the focus and directrix of the parabola, and the directrix is orthogonal to the line passing through $p^{(0)}$ and $p^{(-2)}$. Figure 3 illustrates this prediction. We note that this decision makes the curvature of a predicted parabola largest. By the same spirit, we can predict a future trace of an obstacle even with non-successive record, for example, $p^{(-k_2)}, p^{(-k_1)}, p^{(0)}$ where $0 < k_1 < k_2$.

On the other hand, the boundary is a non-moving object, thus trace prediction is meaningless. Since robots spontaneously avoid collision among themselves, trace prediction for other robots would be also meaningless. Hence, it is enough to consider trace prediction only for the obstacles.

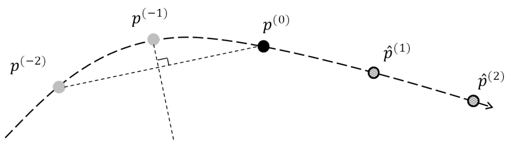
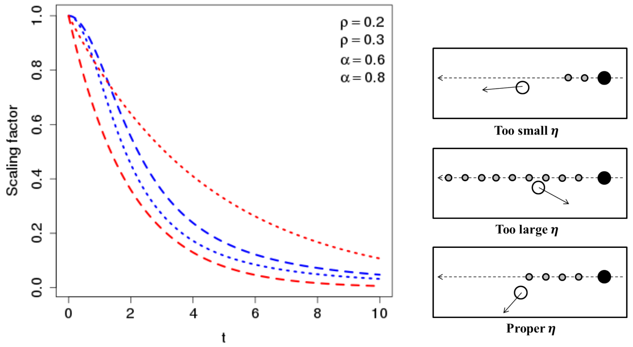


Figure 3. Predicting a trace of an obstacle as a parabola. The dash curve is a predicted parabola. Here, $p^{(-1)}$ becomes the extreme point.



(a) (b)

Figure 4. (a) Comparison of two choices of the scaling function. The blue lines show $1/(1+\rho t^2)$ and the red lines show α^t. (b) The effect of η. The grey circles are the predicted locations which have significant weights.

Now, we propose the following dynamics for a robot $i \in \mathcal{R}$.

$$\dot{p}_i = \sum_{j \in \mathcal{I} \setminus (\mathcal{O} \cup i)} \frac{w_{ij}(p_i - p_j)}{||(p_i - p_j)||^3} + \sum_{j \in \mathcal{O}} \int_0^\infty \frac{w_{ij}(p_i - p_{ij}^{(t)})}{||p_i - p_{ij}^{(t)}||^3} \frac{\zeta_{ij}(t)}{Z_{ij}} dt, \tag{7}$$

where $p_{ij}^{(t)}$, $j \in \mathcal{O}$, is the predicted location of an object j after $t \in (0, \infty]$ time from the current time, and ζ_{ij} is a time scaling function such that $\zeta_{ij}(0) = 1$ and ζ_{ij} decreases over t. Here, $Z_{ij} = \int_0^\infty \zeta_{ij}(t)dt$ is a normalizing factor. One may understand that (7) is a time scaled version of (6) with the prediction $p_{ij}^{(t)}$. The time scaling is natural because the accuracy of the prediction decreases over time.

There are some possible choices for the scaling function ζ_{ij}, for example, the polynomial of the form $1/(1 + \rho t^2)$, the exponential, or the logistic functions. Figure 4(a) shows comparison of scaling functions. Note that in the polynomial scaling, the slope at $t = 0$ is 0 while it cannot be 0 in the exponential scaling. Thus, the scaling function value rapidly decreases in the exponential scaling compared with the polynomial scaling. In this sense, the polynomial scaling is more reasonable because it gives somewhat higher weight to near future. We note that other scaling functions having similar properties, such as the logistic function, could be also used. The exact form of our scaling function is as follows.

$$\zeta_{ij}(t) = \frac{1}{1 + \rho_{ij} t^2}. \tag{8}$$

Note that as ρ_{ij} gets larger, the future becomes more insignificant. We determine the value ρ_{ij} as follows. As described in Section I-B, if a robot i is close to an obstacle j, the collision risk is much higher than when i is far from j. In that case, it is reasonable to give a higher weight to the prediction for the near future. So, first we determine the time $\tau_0 = \tau_0(i, j)$ until which we will give relatively high weights. The time τ_0 depends on $||p_i - p_j||$ and the estimated speed $\hat{v}_{ij}$ of the obstacle j by a robot i. We let

$$\tau_0 = \eta \frac{||p_i - p_j||}{\hat{v}_{ij}}. \tag{9}$$

Here, η is a time scaling factor. If η is too small, prediction is less meaningful and if η is too large, a robot may even move toward the obstacle. This is illustrated in Figure 4(b). In Section IV, we suggest by experiments that $\eta \approx 0.6$ is an adequate value. Then, we obtain ρ_{ij} by calculating the following equation.

$$\frac{\partial^2}{\partial t^2} \left[\frac{1}{1 + \rho_{ij} t^2} \right] (\tau_0) = 0. \tag{10}$$

This is the same as

$$\frac{2\rho(1 + \rho_{ij}\tau_0^2)(1 - 3\rho_{ij}\tau_0^2)}{(1 + \rho_{ij}\tau_0^2)^4} = 0. \tag{11}$$

Then, we obtain

$$\rho_{ij} = \frac{\hat{v}_{ij}^2}{3\eta^2 ||p_i - p_j||^2}. \tag{12}$$

Putting (8) and (12) into (7), the second term changes into

$$\sum_{j \in \mathcal{O}} \int_0^\infty \frac{p_i - p_j^{(t)}}{||p_i - p_j^{(t)}||} \frac{w_{ij}}{||p_i - p_j^{(t)}||^2 + (\varphi_{ij;t}\hat{v}_{ij}/\sqrt{3}\eta)^2 t^2} \frac{1}{Z_{ij}} dt, \tag{13}$$

where $\varphi_{ij;t} = ||p_i - p_j^{(t)}||/||p_i - p_j||$.

As mentioned at the beginning of the section, we have assumed that a robot has the full sensing range. If the sensing range is limited, a robot can compute (7) only for sensed objects. As a consequence, the computation is performed only on objects within i's sensing range. Let $\mathcal{J}_i \in \mathcal{I}$ is a et of objects sensed by i, then

$$\dot{p}_i = \sum_{j \in (\mathcal{R} \cup \mathcal{B}) \cap \mathcal{J}_i} \frac{w_{ij}(p_i - p_j)}{||(p_i - p_j)||^3} + \sum_{j \in \mathcal{O} \cap \mathcal{J}_i} \int_0^\infty \frac{w_{ij}(p_i - p_{ij}^{(t)})}{||p_i - p_{ij}^{(t)}||^3} \frac{\zeta_{ij}(t)}{Z_{ij}} dt. \tag{14}$$

Although in our description only obstacles move over time, the similar method can be applied to moving boundary. Indeed we just defined the boundary as obstacles with velocity 0. Finally, we note that our method can be also applied to the target tracking by using a prediction of the target movement and similar attraction force from the target. Using the prediction, even if a robot is currently somewhat far from the target, it can chase the robot earlier because the tracker receives a more attraction force from the future location of the target rather than just following the target.

IV. Experimental Results

A. Setup and Overview

In this section, we experimentally show the performance of our method. We compare our method with the Voronoi method and the naive potential method using two measures. The first measure is the number of collision to obstacles during the simulation. The other is the potential value (5) by which we can indirectly check the quality of coverage [1]. In this section, we refer the Voronoi method to Voro and naive potential method to Naiv.

The detailed setup is as follows. We consider n robots deployed in an octagon containing m moving obstacles. The radius of the boundary is 18, and each robot's sensing range is limited by 8. The weight w_{ij} depends only on the object j, i.e. $w_{ij} = w_j$. We fix $w_j = 1$ for all $i \in \mathcal{R}$, and $w_j = 6$ for $j \in \mathcal{O}$. We assume that all robots have the same constant speed, and the speed of each obstacle is set by 1.5 times speed of a robot. Each obstacle moves along a straight line and randomly starts a circular movement for a randomly chosen time, then it resumes a straight line movement. If a robot collides to an obstacle, we assume that the robot is broken. Hence, we remove the robot, and put a new robot at a random corner far from all of the obstacles. Through experiments, we consider a discrete time, and the integral in (14) is approximated as follows.

$$\int_0^\infty \frac{w_{ij}(p_i - p_{ij}^{(t)})}{||p_i - p_{ij}^{(t)}||^3} \frac{\zeta_{ij}(t)}{Z_{ij}} dt = \sum_{t=0}^{\tau} \frac{w_{ij}(p_i - p_{ij}^{(t)})}{||p_i - p_{ij}^{(t)}||^3} \frac{\zeta_{ij}(t)}{Z_{ij}}, \tag{15}$$

where τ is a large number.

B. Effect of Scaling Function

In this section, we examine the effect of the scaling function. Especially, we compare the polynomial scaling with varying η and the exponential scaling α^t with varying α. For each parameter value, we did $R = 20$ simulations each of which initial robot locations are randomly chosen, and each simulation continues for $T = 5000$ time steps. We let $n = 80$ and $m = 3$. The table in Figure 5 shows the number of collision happened during the simulation for each method and varying η and α. The bottom graphs in Figure 5 show the ratio of our method's potential over the compared methods averaged over each time step in the 20 simulations. Observe that for all the values of η and α, the number of collision is significantly small in our method compared with the other methods. Especially Voro, which is widely used for coverage maximization, suffers from frequent collisions. In Voro, during the 5000 time steps, 58 robots among 80 in the system collided with obstacles on average. Naiv shows less collisions than Voro, but there are still 11 collisions on average. On the other hand, in our method collision hardly occurs. For 5000 time steps, the averages are below 0.5 for all the polynomial scaling factors regardless of η. Exponential scaling function also shows small number of collisions. Especially the number of collision with $\alpha \approx 0.6$ is as low as that of the polynomial scaling. Note that our method is less affected by the choice of the scaling function as long as the scaling function has similar shape.

			exp. scale (α)		
	Voro	Naiv	0.4	0.6	0.8
Mean	58	11	2.3	0.25	0.70
Std.	6.7	3.0	1.7	0.55	0.73

	poly. scale (η)					
	0.2	0.4	0.6	0.8	1.0	1.2
Mean	0.50	0.40	0.25	0.30	0.45	0.25
Std.	0.51	0.68	0.72	0.57	0.69	0.44

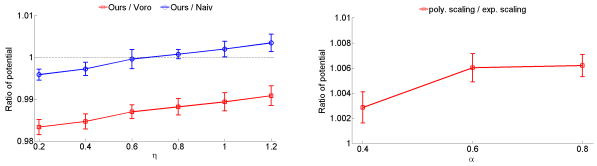
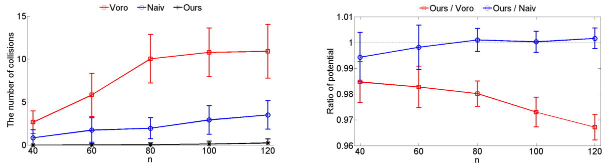


Figure 5. The number of collision and ratio of potential with different η and α. For the all cases, $n = 80, m = 3, T = 5000$ and $R = 20$. In the right graph, η for the polynomial scaling is 0.8 and the exponential scaling function α^t is used.

The bottom graphs of Figure 5 show that our method does not sacrifice the potential value due to the obstacle avoiding effect. The potential of our method is more than 1% lower than that of Voro. Our method keeps the potential even lower than Naiv when $\eta < 0.6$. Note that the larger η, the larger the potential. This is because for large η, far futures are assigned relatively high weights as explained in Figure 4(b). The low potential and small number of collision implies that our method properly considers movement of moving obstacles. Hence, we could verify that our method is much better than the existing methods in a dynamic environment, and observed that the choice of the scaling function is less significant.

C. Effect of External Factors

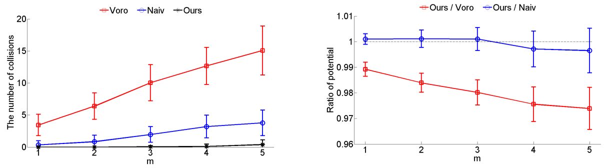
In this section, we examine the effect of the number of robots, the number obstacles, and the speed of the obstacles. We let $T = 1000$ and $R = 50$, and initial locations of robots are randomly generated. We let $n = 80, m = 3$ and $\eta = 0.8$ as default values while varying possibly one parameter. Figure 6 shows the number of collisions for each method and the ratio of our method's potential over the compared methods averaged over time while varying n. We check that in our method the number of collision is much smaller than the others. Note that not only the number of collision is small in our method, but also the growth over n is very small compared with the others. The growth is nearly flat in our method. Figure 7 shows the results when m varies. The tendency is similar to the former case: as m gets larger,



(a)

(b)

Figure 6. The number of collision and ratio of potential with different number of robots n, the number of obstacles. Here $m = 3, \eta = 0.8, T = 1000$ and $R = 50$.



(a)

(b)

Figure 7. The number of collision and ratio of potential with different number of obstacles m. Here $n = 80, \eta = 0.8, T = 1000$ and $R = 50$.

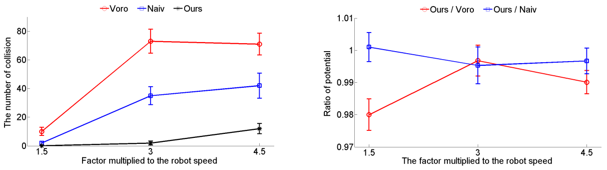
collision is more frequent, and the growth for our method is flat while it is not for the others.

Our method also guarantees low potential. When n gets larger, the potential by our method is slightly above that by Naiv while ours becomes below Naiv's when m gets larger.

Figure 8 shows comparison of three methods when the speed of obstacles varies. The performance gap between ours and the others for the number of collision dramatically increases when the obstacles' speed is three times of the robots' speed. In our method, the number of collision is still very small. For higher speed, 4.5 times of the robots' speed, collision is more frequent in our method, but note that still it is quite small compared with the other methods.

V. Conclusions

In this paper, we proposed a novel decentralized control law for mobile robot systems to avoid moving obstacles. We show that our method significantly reduces the number of collision while keeping the potential low compared with the Voronoi and the previous potential method. We believe that our method also provides a general framework for number of tasks of multiagent mobile robot systems, including target tracking and leader following.



(a)

(b)

Figure 8. The number of collision and ratio of potential with different obstacle speed (robot speed ×1.5, ×3 and ×4.5). Here $n = 80, m = 3, \eta = 0.8, T = 1000$ and $R = 50$.

References

[1] A. Howard, M. J. Mataric, and H. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Int. Conf. Distributed Autonomous Robotic Systems*, 2002.

[2] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 34, 2004.

[3] A. Fagiolini, L. Tani, A. Bicchi, and G. Dini, "Decentralized deployment of mobile sensors for optimal connected sensing coverage," in *Proceedings of the 4th IEEE international conference on Distributed Computing in Sensor Systems*, 2008.

[4] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," in *In Proceedings of the 2000 Congress on Evolutionary Computation*, 2000.

[5] W. Kowalczyk and K. Kozlowski, "Artificial potential based control for a large scale formation of mobile robots," in *Proceedings of the Fourth International Workshop on Robot Motion and Control (RoMoCo)*, 2004.

[6] S. Poduri and G. S. Sukhatme, "Constrained coverage for mobile sensor networks," in *IEEE International Conference on Robotics and Automation*, 2004.

[7] L. Huang, "Velocity planning for a mobile robot to track a moving target - a potential field approach," *Robotics and Autonomous Systems*, vol. 57, 2009.

[8] J. H. Chuang, "Potential-based modeling of three dimensional workspace for obstacle avoidance," *IEEE Transactions on Robotics and Automation*, vol. 14, 1998.

[9] J. Yan, X. P. Guan, and F. X. Tan, "Target tracking and obstacle avoidance for multi-agent systems," *International Journal of Automation and Computing*, vol. 7, 2010.

[10] C. Fulgenzi, A. Spalanzani, and C. Laugier, "Probabilistic motion planning among moving obstacles following typical motion patterns," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009.

[11] M. Massari, G. Giardini, and F. Bernelli-Zazzera, "Autonomous navigation system for planetary exploration rover based on artificial potential fields," in *Dynamics and Control of Systems and Structures in Space (DCSSS)*, 2004.

[12] J. Hagelbäck and S. J. Johansson, "A multiagent potential field-based bot for real-time strategy games," *International Journal of Computer Games Technology*, vol. 2009, 2009.

[13] F. Fahimi, C. Nataraj, and H. Ashrafiuon, "Real-time obstacle avoidance for multiple mobile robots," *Robotica*, vol. 27, 2009.

[14] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, 1991.

[15] ——, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, 1989.

[16] T. Zhang, X. Li, Y. Zhu, S. Chen, and Y. Cheng, "Formation and obstacle avoidance in the unknown environment of multi-robot system," in *Proceedings of the 2009 international conference on Robotics and biomimetics (ROBIO 09)*, 2009.

[17] J. Hagelbäck and S. J. Johansson, "Using multi-agent potential fields in real-time strategy games," in *AAMAS*, 2008.

[18] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transaction on Robotics and Automation*, vol. 20, 2004.

[19] R. Xiong, Y. Q. Bai, W. Sun, and X. Liu, "Mobile sensor network node deployment via central voronoi tessellation," in *2nd International Conference on Intelligent Control and Information Processing (ICICIP)*, 2011.

[20] Y. Uchimura, T. Imaizumi, and H. Murakami, "Mobile robot deployment based on voronoi diagram," in *International Symposium on Access Spaces*, 2011.