2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Coordinated Control of Multiple Mobile Robots in Pursuit-Evasion Games

Feifei Huang, Long Wang, Qining Wang, Ming Wu and Yingmin Jia

Abstract— **This paper considers the scenario that a team of autonomous robots pursues a smart evader while concurrently building a global map in an unknown environment. We develop a modular hybrid system architecture to implement the decentralized control and coordination of multiple pursuers. A probabilistic framework is established to integrate the map building process with evader detection. Taking account of the nonholonomic constraints of vehicles, we improve the global-max pursuit policy with a novel underlying navigation method. Experimental results with physical robots demonstrate the effectiveness of our approach.**

I. INTRODUCTION

Over the last decades, the investigation on multi-robot systems ([1]-[3]) has attracted a tremendous attention in robotic literature, with extensive applications ranging from exploration [4], search and rescue [5], to military detection. The utility of multiple robots have shown a variety of potential advantages over a single robot, including higher efficiency, greater flexibility and robustness. Therefore, the control and coordination strategies for multi-robot systems is definitely significant to address complex tasks. In this paper, we study the coordinated multi-agent pursuit-evasion games, and emphasize on the autonomy of each robot while allowing for the team-level cooperation.

In pursuit-evasion games, a team of pursuers searches in parallel for a moving evader. Once a pursuer detects the evader, it signals the others via communication network. At the same time, the cooperative pursuit policy is activated to try to capture the evader who, in turn, tries to avoid being captured. Many researches have been done in developing pursuit policies with considering either known environments ([6][7]) or unknown environments ([8]-[11]). The former provide sophisticated approaches to deal with the worst-case motion of evaders. Whereas, from the point of view of implementation in practice, [9] proposed a probabilistic game theoretic framework and discussed two greedy pursuit policies: the local-max and the global-max. Then [10] extended this probabilistic approach and the global-max pursuit policies to cope with the problem for multiple evaders, and developed a distributed control architecture for a team of heterogeneous robots.

This work was supported by National High Technology Research and Development Program of China (863 Program) (No. 2006AA04Z258).

F. Huang, L. Wang, Q. Wang and M. Wu are with the Intelligent Control Laboratory, Center for Systems and Control, College of Engineering, Peking University, Beijing 100871, China. feifeihuang@coe.pku.edu.cn

Y. Jia is with the Seventh Research Division, Beijing University of Aeronautics and Astronautics, Beijing 100083, China.

In unknown environments, the pursuers have to construct a global map and determine the locations of themselves and the evader relative to the map, in order to plan their pursuit paths as well as to coordinate their actions. The robotic mapping problem has been deeply studied in the topic of *Simultaneous Localization and Mapping* ([12]-[14]). The state of the art in SLAM has been dominated by probabilistic techniques, of which most are based on Bayesian estimation and are performed using specially designed filters, such as Extended Kalman Filter (EKF), Particle Filter (PF) and so on. In the pursuit-evasion scenario, a computational feasible mapping algorithm is required for planning pursuit strategies. [11] posed the map building problem as determining the maximum-likelihood estimate of cells occupied by the evader and the obstacles. This process was then integrated with the pursuit-evasion games into a probabilistic framework. An alternative solution to the problem of modeling dynamic environments is the algorithm of Simultaneous localization, mapping and moving object tracking (SLAMMOT) [15].

In this paper, a modular hybrid system architecture is established for decentralized control and coordination of multiple robots. Then a probabilistic framework is formulated for the concurrent map building and evader detection. Taking account of the physical constraints of robots, a modified version of global-max pursuit policy is developed with a trade-off suboptimal navigation approach. Meanwhile, a smart evader is allowed to avoid being captured by using evasion policy.

The rest of the paper is organized as follows. In Section II, the system architecture is introduced. Section III presents the proposed pursuit-evasion algorithms in detail. Experimental results are presented in section IV with conclusions and future works in section V.

II. HYBRID SYSTEM ARCHITECTURE

The objective of our system is to make a team of mobile robots effectively implement complex missions. To achieve this goal, a unified framework for coordination of multiple platforms is developed with the following features: 1) The control system is decentralized across the robotic team. 2) The program is modular to make it possible to improve or extend the system performance. 3) The system architecture is hierarchical in order to facilitate development and maintenance.

For each pursuer, the *Coordination Module*, depicted in Fig. 1, is implemented by exchanging and sharing information with the others over the *Communication Network*. The *Map Merging and State Transformation* serves as a process for combining the local maps from multiple robots into a global map which describes the whole region explored. Then the states of evader and the other pursuers might be transformed into that new reference frame for the purpose of planning strategies. The *Coordination and Strategy Planner* component is responsible for the design of coordination method and pursuit policies for pursuers.

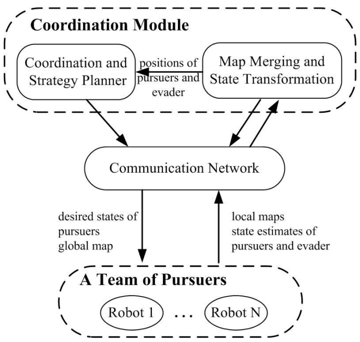


Fig. 1. Coordination level architecture.

In the vehicle level, the *Control Module* is responsible for the overall logic control of each robot, consisting of three components shown in Fig. 2. The *Sensor Fusion* component provides useful measurements to the other two components by integrating the raw sensor data from different sensors. The *Map Builder and Evader Detector* component is responsible for generating probabilistic maps with the state estimates of the evader and pursues. The *Regulation and Trajectory Planner* determines the destination way-points of pursuers in a time sequence.

The *Hardware Module* receives control signals from the *Regulation and Trajectory Planner* component, and then produces the actual states information as the output which is sent to the *Control Module* subsequently.

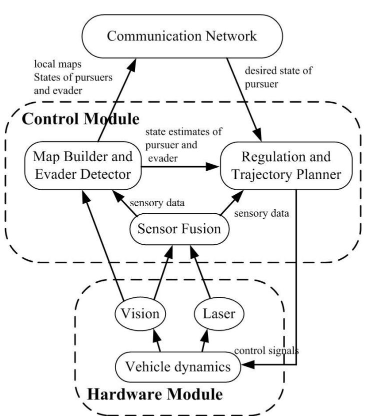


Fig. 2. Vehicle level architecture.

III. PURSUIT-EVASION ALGORITHMS

This section presents our approach to pursuit-evasion games in an unknown environment, involving concurrent map building and evader detection, pursuit policies, and evasion policy.

A. Concurrent Mapping and Evader Detection

We extend the general SLAM algorithm to address a more complex problem of *Concurrent Mapping and Evader Detection* (inspired by SLAMMOT in [15]). In pursuit-evasion games, each pursuer starts at a different position with a blank map. Then each one concurrently performs three tasks during the searching process: It determines a maximum likelihood estimate for the map and its own location relative to this map with maximum possibility. Besides, the pursuer who has detected the evasion should signals the other pursuers immediately, and should determine a maximum likelihood estimate for the evader. Before formulating our algorithms, three assumptions are made to simplify the computation of our algorithms: 1) The motion models of vehicles are Markov. 2) The measurements of the moving evader can be distinctly separated from those of the stationary objects. 3) The measurements of evader carries no information about either the pursuer's states or stationary objects.

1) Evader Detection: From the second assumption, we obtain that

$$Z_k = Z_k^m \cup Z_k^e \tag{1}$$

where Z_k^m and Z_k^e represent measurements of stationary landmarks and the moving evader, respectively. This assumption implies that a reliable method for the detection of moving evader must be applied. [16] proposed an optical flow based technique to detect moving objects, whose poses and velocities were estimated using subspace constraints. However the computational complexity of that method is too great to operate in real-time applications. In this paper, we simply use a salient color marker to identify the evader, whose state estimates are accomplished through the proposed algorithm of *Concurrent Mapping and Evader Detection*, as described below.

2) Probabilistic Framework: Considering a nonholonomic robot, whose kinematics is given by

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} \tag{2}$$

where the control input $u = (\upsilon,\ \omega)^T$ denotes the translational velocity and rotational velocity of the vehicle, respectively. $(x,\ y,\ \theta)^T$ is the position vector of the center of the robots.

For arbitrary pursuer $P_i, (i \in [1, 2, \cdots N])$, the formula for sequential map building and evader detection can be expressed as the posterior

$$p(x_k,\ x_k^e,\ M_k |\ U_k,\ Z_k) \tag{3}$$

where x_k denotes the state vector of pursuers describing $2D$ coordinates and orientation at time instant k, $x_k^e = [q_k^e,\ \upsilon_k^e,\ \omega_k^e]^T$ the state vector of the evader, where q_k^e describes the $2D$ coordinates and orientation of the evader. $U_k = [u_1, u_2, \cdots, u_k]$ is the history of control inputs of the pursuers, $Z_k = [z_1,\ z_2, \cdots,\ z_k]$ the set of all the observations, and M_k the map involving the stationary landmarks in the environment.

Based on Bayes' rules and the assumptions mentioned above, the general recursive Bayesian formula for concurrent mapping and evader state estimation can be derived as

Prediction:

$$\begin{aligned} & p(x_k,\ x_k^e,\ M_k|\ U_k,\ Z_{k-1}) \\ = \ & p(x_k^e|\ Z_{k-1}^e,\ U_k) p(x_k,\ M_k|\ Z_{k-1}^m,\ U_k) \\ = \ & \underbrace{\int p(x_k|\ U_k, x_{k-1}) p(x_{k-1},\ M_{k-1}|\ Z_{k-1}^m,\ U_{k-1}) dx_{k-1}}_{\text{SLAM Prediction}} \\ & \cdot \underbrace{\int p(x_k^e|\ x_{k-1}^e) p(x_{k-1}^e|\ Z_{k-1}^e,\ U_{k-1}) dx_{k-1}^e}_{\text{Evader State Prediction}} \end{aligned} \tag{4}$$

Update:

$$\begin{aligned} & p(x_k,\ x_k^e,\ M_k|\ U_k,\ Z_k) \\ = \ & \underbrace{p(Z_k^m|\ X_k,\ M_k)}_{\text{SLAM Update}}\ \underbrace{p(Z_k^e|\ x_k^e,\ x_k)}_{\text{Evader State Update}} \\ & \cdot \frac{p(x_k,\ x_k^e,\ M_k|\ U_k,\ Z_{k-1})}{p(Z_k|\ Z_{k-1},\ U_k)} \end{aligned} \tag{5}$$

It is worth noting that initializing a new moving object is a relatively hard problem, in that the velocity cannot be directly obtained using camera and laser rangefinder. To address this problem, data association techniques must be emphasized. The Joint Probabilistic Data Association Filter (JPDAF), one of the most popular approaches among all the solutions, is employed in this paper. A more detailed discussion refers to [17].

B. Pursuit Policy

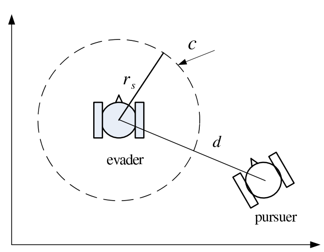


Fig. 3. Capture criterion.

Given a probabilistic map of environment, the capture criterion must be defined in order to plan pursuit policies. Due to the physical constraints, two vehicles cannot arrive at the same location or occupy the same cell. As shown in Fig.3, C refers to the circle boundary of the safe region which is defined with respect to the center of the evader. The radius of C is denoted by r_s which is assumed as constant. Then C can be represented in polar coordinates as

$$C = q^e + r_s e^{i\theta} \tag{6}$$

Capture Criterion: Once the distance d between the evader and an arbitrary pursuer $P_i, i \in [1,\ 2, \cdots,\ n_p]$ satisfies $d \leq r_s$, the evader is considered to be captured, and the pursuit-evasion game is over.

1) Improved Global-max Policy: Global-max policy has proved an efficiently suboptimal pursuit policy in [9]. However, the global search is based on the assumption that both the evader and pursuer occupy the same cell when the evader is captured. The assumption overlooks the mechanical constraints of physical robots. In this paper, we improve this global-max policy by taking account of the nonholonomic characteristic. Our *improved Global-max* policy also searches over the entire map, attempting to maximize the probability of capturing the evader at next time instant.

From (4), the location of evader at time $k+1$ with maximum probability can be predicted relying on a sequence of measurements Z_k, such that

$$q_{k+1}^{e*} = \arg \max_{q_{k+1}^e \in \mathbb{R}^2} p(q_{k+1}|\ Z_k) \tag{7}$$

The desired positions of pursuers can be calculated by (5)

$$\begin{aligned} q_k^* &= [x_{k1},\ x_{k2}, \cdots,\ x_{kn_p}] \\ &= \arg \max_{x_{k1}, \cdots,\ x_{kn_p} \in \mathbb{R}^2} \Sigma_{i=1}^{n_p} p(x_{ki}|\ Z_k,\ U_k) \end{aligned} \tag{8}$$

Let us describe our pursuit policy as

$$U_{k+1} = nav(q_k^*, i^*, Z_k) \tag{9}$$

where $i* \in 1, \cdots,\ n_p$ is the integer for which

$$q_k^* = \ q_{k+1}^{e*} + \ r_s e^{i\theta^*} \tag{10}$$

where $\theta^* \in [0,\ 2\pi)$ is an angle determined by the proposed navigation method.

In comparison with ([9][10]), another difference is that a novel trade-off underlying navigation *nav* method is employed, as discussed below. We distribute path for each pursuer by taking a sequence of destination way-points q^*, in order to produce a state reachable in a single time step to the desired position. According to the *Capture Criterion*, one pursuer has to move and ultimately arrive at a certain location that lies in the safe region of evader. Since the circle boundary C moves along with the evader and the capture time is unpredicted, searching the global shortest optimal path is indeed a complex and time-consuming task, which is unsuitable for real-time implementation. Therefore, an intuitive heuristic navigation policy is developed for each pursuer.

At each time instant, the shortest path for one pursuer to capture the evader is the straight line linking the way-point q^* and the current position of the pursuer. Both the distance and bearing from the pursuer to the evader can be measured using laser rangefinder and camera, respectively. Thus the value of θ_k^* can be uniquely determined. Although each segment of trajectory is the shortest in small time interval, the entire trajectory may not be the optimal but tractable. For simplicity, we call this direct path planning method *Move To Destination* action. It is understandable that the *Move To Destination* action is executed in the situation without risks of any collisions, therefore it cannot generate safe paths for pursuers.

As a team of robots pursuits an evader, there may be a danger of collisions among those robots and other stationary obstacles, such that the pursuers should switch to the *Collision Avoidance* action. Inspired by [18], we introduce a *Dynamic Role based Limit Cycle* approach to cooperatively avoid collisions among multiple pursuers (the situation for stationary obstacles is similar). Before describing our algorithm, a brief discussion about the limit cycle approach is given as below.

2) Limit Cycle: Let us consider the nonlinear system

$$\begin{aligned} \dot{\bar{x}} &= \lambda(\bar{y} + \varepsilon\bar{x}(r^2 - \bar{x}^2 - \bar{y}^2)) \\ \dot{\bar{y}} &= \lambda(-\bar{x} + \varepsilon\bar{y}(r^2 - \bar{x}^2 - \bar{y}^2)) \end{aligned} \tag{11}$$

where λ,ε and r are positive parameters. Next we introduce the following Lyapunov function

$$V(\bar{x},\bar{y}) = \bar{x}^2 + \bar{y}^2 \tag{12}$$

such that the derivative of $V(\bar{x},\bar{y})$ along the trajectories of the given system is

$$\begin{aligned} \dot{V}(\bar{x},\bar{y}) &= 2\lambda\varepsilon(r^2 - \bar{x}^2 - \bar{y}^2)(\bar{x}^2 + \bar{y}^2) \\ &= 2\lambda\varepsilon(r^2 - V(\bar{x},\bar{y}))V(\bar{x},\bar{y}) \end{aligned} \tag{13}$$

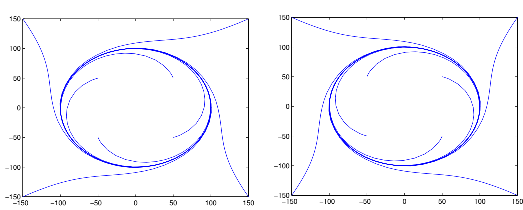
We can see that $\dot{V}(\bar{x},\bar{y}) > 0$ for $V(\bar{x},\bar{y}) < r^2$, while $\dot{V}(\bar{x},\bar{y}) < 0$ for $V(\bar{x},\bar{y}) > r^2$. This shows the region

$$\Omega = \{\mu_1 \le V(\bar{x},\bar{y}) \le \mu_2, |\ 0 < \mu_1 < r^2, \mu_2 > r^2\} \tag{14}$$

is absorbing. When μ_1, μ_2 get close to r^2, the region Ω shrinks toward the periodic orbit $V(\bar{x},\bar{y}) = r^2$, which is called a *limit cycle*.The trajectory from any point $(\bar{x},\bar{y})$ ultimately converges to this limit cycle clockwise. The counter-clockwise situation can be derived by

$$\begin{aligned} \dot{\bar{x}} &= \lambda(-\bar{y} + \varepsilon\bar{x}(r^2 - \bar{x}^2 - \bar{y}^2)) \\ \dot{\bar{y}} &= \lambda(\bar{x} + \varepsilon\bar{y}(r^2 - \bar{x}^2 - \bar{y}^2)) \end{aligned} \tag{15}$$

The convergence speed of the system toward the limit cycle can be adjusted by the value of ε. Fig. 4 shows the phase portraits of (13) and (15) with a relatively slow convergence speed ($\varepsilon = 0.0001$), respectively. Fig. 4(a) illustrates the clockwise limit cycle, and Fig. 4(b) the counter-clockwise.



(a) Clockwise limit cycle. (b) Counter-clockwise limit cycle.

Fig. 4. Flexible limit cycle.

Since the trajectory from any point inside or outside the limit cycle moves toward the cycle and therefore keeps a certain distance to the center point, the limit cycle provides a approach to collision avoidance among multiple robots.

3) Dynamic Role based Limit Cycle Navigation Policy: In order to plan collision-free paths for the pursuers to capture the evader, we will show how to employ the limit cycle method in the *Collision Avoidance* action by multiple pursuers.

Assume that when all the pursuers are implementing the *Move To Destination* action, some pursuers are at the risk of collisions with each other. For simplicity, we consider the situation for two pursuers P_i and P_j, $(i,\ j \in [1,\cdots,n_p],\ i \neq j)$ which are assumed to be circular. Since the goal of both pursuers is the same evader, it is impossible that both pursuers are obstructed by each other. The unique possibility is that one pursuer is obstructed by the other one. Assume P_i obstructs P_j on the path of P_j at time instant k. Then P_i will act the role of an dynamic obstacle and keep implementing *Move To Destination* action, while P_j will switch to the *Collision Avoidance* action to avoid colliding with P_i.

The first step for P_j is to decide the rotational direction taken to avoid P_i (clockwise or counter-clockwise). By using the process of *Concurrent Mapping and Evader Detection*, the global coordinates of the evader R_e and the two pursuers can be estimated. Thus the line l through the centers of P_j and R_e can be uniquely determined, which is described as:

$$\alpha x + \beta y + \gamma = 0$$

Next we calculate the distance ρ from the center of P_i to the line l, such that

$$\rho = \frac{\alpha x_i + \beta y_i + \gamma}{\sqrt{\alpha^2 + \beta^2}} \tag{16}$$

To adjust (11) to adapt the navigation plan, it can be rewritten by

$$\begin{aligned} \dot{\bar{x}} &= \lambda(\frac{\rho}{|\rho|}\bar{y} + \varepsilon\bar{x}(r^2 - \bar{x}^2 - \bar{y}^2)) \\ \dot{\bar{y}} &= \lambda(-\frac{\rho}{|\rho|}\bar{x} + \varepsilon\bar{y}(r^2 - \bar{x}^2 - \bar{y}^2)) \end{aligned} \tag{17}$$

we can see that if $\rho > 0$, the pursuer P_j avoids P_i in clockwise direction. Whereas the avoidance takes place counter-clockwise.

Now a motion controller is designed for P_j. The input signals $(\upsilon_j,\ \omega_j)$ can be calculated as follows.

Considering the situation for clockwise direction (the situation for counter-clockwise is similar), the system (11) can be expressed in the original frame as

$$\begin{aligned} x_j &= \cos\theta_i(\bar{x} + x_i) - \sin\theta_i(\bar{y} + y_i) \\ y_j &= \sin\theta_i(\bar{x} + x_i) + \cos\theta_i(\bar{y} + y_i) \end{aligned} \tag{18}$$

From the kinematics (2) and (18), we can obtain the values of the input signals

$$\begin{aligned} \upsilon_j &= \sqrt{\dot{\bar{x}}^2 + \dot{\bar{y}}^2} \\ \omega_j &= \frac{\dot{\bar{x}}\ddot{\bar{y}} - \dot{\bar{y}}}{\dot{\bar{x}}^2 + \dot{\bar{y}}^2} + \omega_i \end{aligned} \tag{19}$$

Fig. 8(d) depicts the situation that P_j avoids P_i clockwise and then tracks a target moving in sine wave.

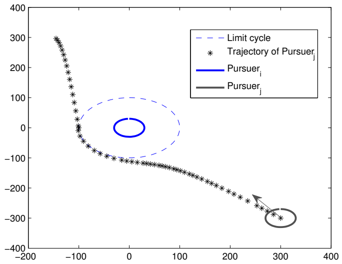


Fig. 5. Clockwise limit cycle based collision avoidance for two pursuers.

Fig. 6. Pioneer 2DX robots used in the experiments.

In the pursuit-evasion scenario, the pursuers have no knowledge of the motion trajectory of the evader *a priori*. Also the control inputs of pursuers are different from each other, yielding diverse actual states. Consequently, new obstructions may occur. For example, at the time instant $k+m$ (m is a sample time), P_j possibly obstructs P_i on the path of P_i. At that time the roles of both the pursuers have to exchange, that is to say, P_j switches to the role of dynamic obstacle and executes the *Move To Destination* action, whereas P_i switches to the *Collision Avoidance* action until the danger of collision disappears. In other words, there are a variety of uncertainties in such a dynamic environment of pursuit-evasion games. To plan completed safe paths in the pursuit of an evader, the pursuers must adjust their roles to the *Collision Avoidance* action. Thus, we call it the *dynamic role based limit cycle* navigation method, which can be extended to the situations for N pursuers.

C. Evasion Policy

In this paper, an intelligent evader is allowed to make evasion policy corresponding to the pursuers. As described above, the limit cycle navigation method can also act as the evasion policy for the evader to avoid collisions with the pursuers. Since r_s is the critical distance value for the evader at the risk of being captured, the radius of the cycle limit for each pursuer must be control by $r > r_s$.

IV. EXPERIMENTAL RESULTS

The proposed approach has been tested both on real robots and on a multi-robot simulation platform–Mobilesim .

The real robot experiment is carried out in an office-like environment. we consider a team of three Pioneer 2DX robots equipped with laser rangefinders (LRFs) covering a 180^0 field of view, a pan-tit-zoom (PTZ) CCD camera, and a wireless device, as shown in Fig. 6. The robot with salient yellow markers acts as the evader, so that it can be identified easily and uniquely by the camera sensors; whereas the other two robots act as the pursuers, who can communicate with each other through the wireless network. In our experiments, the messages transmitted through the wise network include the map data and the state estimates of each robot.

The detailed procedure of mapping is implemented as follows: 1) We select lines as features for map building in the corridor environment. 2) Line segments are extracted from the raw laser data by using *Hough Transformation*. The matching process is performed in Hough domain, where measuring distances of points corresponds to measuring displacement of lines in the Cartesian coordinates. The next step is to find correspondences between a determined threshold and the reference points. 3) JPDAF is employed for the data association of concurrent mapping and evader detection. In the mission of SLAM, we prefer the well-known FastSLAM algorithm, which has a relative lower computational burden and thus is more suitable for the on-line map building. However, the integration of mapping and moving evader tracking leads to a hard work in initializing the moving evader since its velocity cannot be directly calculated. To guarantee the accuracy of data association, the tradeoff approach JPDAF is employed by sacrificing the computational time. The experiment of our pursuit policy

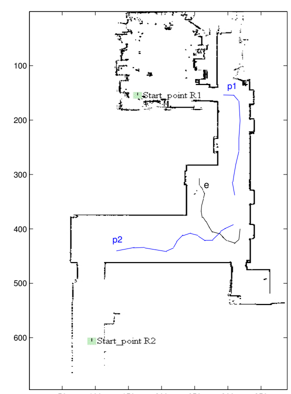
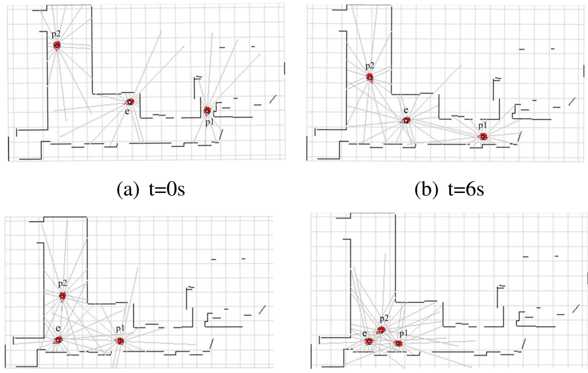


Fig. 7. The paths of two pursuers and an evader in real environment.

is performed based on the assumption that all the pursuers and the evader have known the global map *a priori*. As shown in Fig. 7, three trajectories are drawn in the global map that is produced by merging the local maps built by the pursuers. The blue lines express the pursuit trajectories of the two pursuers P_1 and P_2, respectively. Whereas the black line denotes the evasion trajectory of the evader. The initial translational velocities of the pursuers and the evader are set as $0.4m/s$ and $0.3m/s$, respectively. When a pursuer discovers the evader, it communicates with its partner, and sends the state estimates of the evader. And the pursuit policies are activated simultaneously. Once the evader detects itself in the danger state, the evasion policy is activated. That



(a) t=0s (b) t=6s

(c) t=10s (d) t=16s

Fig. 8. Screen shots of simulations for two pursuers and an evader in the corridor map using Mobilesim.

is, if the distance measured from the evader to the nearest pursuer d is smaller than the defined critical value, the evader implements the evasion policy. When one pursuer moves into the safe region of the evader, the evader stops at once and sends commands to make the pursuers stop concurrently at their current locations.

To get a more quantitative assessment of our algorithms, we perform a simulation experiment by using the multi-robot simulation platform Mobilesim. In this situation, we also employ two pursuers and one evader in the given map. Screen shots of the simulation results during a run of the pursuit-evasion game are shown in Fig. 8. To carry out these experiments, we used sensors with a 180^0 field of view. Fig. 8(a) depicts the initial states of the three robots in the map, Fig. 8(b) and Fig. 8(c) the states of the three robots at $t = 6s$ and $t = 10s$, respectively. Fig. 8(d) the states of the robots at $t = 16s$. At $t = 16s$, the distance between the evader and the pursuer P_2 is smaller than the given threshold. Thus the evader is considered to be captured and all the robots stop moving. These simulation results illustrate the validity and effectively of the proposed approach.

V. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this paper, we investigate the pursuit-evasion games with multiple autonomous robots in an unknown environment. First, a modular hybrid system architecture is proposed for the decentralized control of multiple pursuers. Then we incorporate processes of SLAM and the moving evader detection into a probabilistic framework. Additionally, we improve the global-max pursuit policy with a novel underlying navigation method, which takes into account the physical constraints of robots and introduces a dynamic role based limit cycle technique to address the problem of cooperative collision avoidance. Both the real robot experiments and the simulation using Mobilesim demonstrate the validity and effectiveness of our approach.

B. Future Works

In the future, we will extend the proposed multi-robot control strategies to the situation of multiple evaders. For actual applications, we also shall consider clutter environments. For example, in the surroundings with several people randomly walking, the limited field of view of pursuers may be blocked and therefore lose the tracking of evaders. How to deal with these uncertainties and how to recover the detection of evaders should be deeply studied in future works.

REFERENCES

[1] J. Feddema and D. Schoenwald, "Decentralized control of cooperative robotic vehicles," *IEEE Trans. Robotics and Automation*, vol.18, no.5, 2002, pp. 852-864.

[2] R. Fierro, A. Das, J. Spletzer, J. Esposito, V. Kumar, J.P. Ostrowski, G. Pappas, C.J. Taylor, Y. Hur, R. Alur, I. Lee, G. Grudic, and B. Southall, "A framework and architecture for multi-robot coordination," *Int. J. Robotics Research*, vol.21, no.10-11, 2002, pp. 977-995.

[3] H. Yamaguchi, "A cooperative hunting behavior by mobile robot troops," in *Proc. Conf. Robotics and Automation*, Belgium, 1998, pp. 3204-3209.

[4] W. Burgard, M. Moors, C. Stachniss, and F.E. Schneider, "Coordinated multi-Robot exploration," *IEEE Trans. Robotics*, vol.21, no.3, 2005, pp. 376-386.

[5] J.S. Jennings, G. Whelan, and W. F. Evans, "Cooperative search and rescue with a team of mobile robots," in *Proc. Int. Conf. Advanced Robotics*, Canada, 1997, pp. 193-200.

[6] L. Guibas, J. Latombe, S. LaValle, D. Lin, and R. Motwani, "Visibility-based pursuit-evasion in a polygonal environment," *Int. J. Computational Geometry and Applications*, vol.9, no.5, 1999, pp. 471-494.

[7] S. LaValle and J. Hinrichsen, "Visibility- based pursuit-evasion: an extension to curved environments," in *IEEE Int. Conf. Robot and Automation*, 1999, pp. 1677-1682.

[8] R. Vidal, S. Rashid, C. Sharp, O. Shakernia, J. Kim, and S. Sastry. "Pursuit-evasion games with unmanned gound and aerial vehicles," in *IEEE Int. Conf. Robot and Automation*, 2001, pp. 2948-2955.

[9] R. Vidal, O. Shakernia, J. Kim, D.H. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation," *IEEE Trans. Robotics and Automation*, vol.18, no.5, 2002, pp. 662-669.

[10] J. Kim, R. Vidal, H. Shim, O. Shakernia, and S. Sastry, "A hierarchical approach to probabilistic pursuit-evasion games with unmanned ground and aerial vehicles," in *Proc. 40th IEEE Conf. Decision and Control*, 2001, pp. 634-639.

[11] J. Hespanha, J. Kim, and S. Sastry, "Multiple-agent probabilistic pursuit-evasion games," in *Proc. 38th IEEE Conf. Decision and Control*, vol.3, 1999, pp. 2432-2437.

[12] M.W.M.G. Dissanayake, P. Neuman, S. Clark, H.F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building problem," *IEEE Trans. Robotics and Automation*, vol.17, no.3, 2001, pp. 229-241.

[13] S. Thrun, "A probabilistic online mapping algorithm for teams of mobile robots," *Int. J. Robotics Research*, vol.20, no.5, 2001, pp. 335-363.

[14] M. Di Marco, A. Garulli, A. Gianitrapani, and A. Vicino, "Simultaneous localization and map building for a team of cooperating robots: a set membership approach," *IEEE Trans. Robotics and Automation*, vol.19, no.2, 2003, pp. 238-249.

[15] C-C. Wang, C. Thorpe, S. Thrun, M. Hebert and H. Durrant-Whyte, "Simultaneous Localization, Mapping and Moving Object Tracking," *Int. Jour. Robotics Research*, vol.26, no.9, 2007, pp. 889-916.

[16] R. Vidal and S. Sastry, "Vision-based detection of autonomous vehicles for pursuit-evasion games," in *Proc. IFAC World Congr. Automation Control*, Spain, 2002.

[17] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, (1983). "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Oceanic Engineering*, vol.8, 1983, pp. 173-184.

[18] D. Zhang, L. Wang, G. Xie, and W. Zhang, "Coordinated Collision Avoidance of Multiple Biomimetic Robotic Fish," *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Berlin Heidelberg, 2005, pp. 215-224.