Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

# Trajectory Planning for An Unmanned Ground Vehicle Group Using Augmented Particle Swarm Optimization in a Dynamic Environment

Yunji Wang, Philip Chen, Yufang Jin
Department of Electrical and Computer Engineering
University of Texas at San Antonio
San Antonio, TX 78249, United States
Yufang.jin@utsa.edu

***Abstract*— Optimal path planning is a key problem for the control of autonomous unmanned ground vehicles. Particle swarm optimization has been used to solve the optimal problem in the static environment; however, optimal path planning for UGV groups in a dynamical environment has not been fully discussed. Accordingly, a dynamic obstacle-avoidance path planning for an unmanned ground vehicle group was considered as optimal problem for shortest path with formation constraints. The problem was formulated in Cartesian space with detectable velocity of both the vehicles and obstacles. The fitness function was defined by minimizing the trajectory of the group while keeping the V-shape formation of the group. Stable region of the parameters are determined by analyzing the convergence of the PSO algorithm. The simulation results demonstrated that the augmented particle swarm optimization could get the shortest path while keeping the V-formation and converged very fast.**

***Keywords*—Obstacle avoidance, Particle swarm optimization, Unmanned ground vehicle**

## I. INTRODUCTION

Obstacle avoidance and path planning are two important problems for realization of an intelligent and autonomous unmanned ground vehicle (UGV). Extensive researches have been performed to address these problems and different methods have been applied, for example, potential field algorithms for a UGV in a static environment without any moving obstacles [1], [2], [3], [4]. In addition, trajectory planning for one UGV in dynamical environment has been presented by Fraichard [5]. Recently, cooperation among a UGV group imposed the demand of trajectory planning for a group of UGV while keeping a desired formation in a dynamic environment. Existing studies have demonstrated the effectiveness of fuzzy logic and potential fields method to find an optimal collision free trajectory for a group of UGV with a fixed formation in a static environment [6], [7]. However, little research has been performed on the optimal path planning of a UGV group in a dynamical environment. Therefore, the objective of this study is to formulate the path planning for a group of UGV as an optimal problem with some formation flexibility.

Particle swarm optimization (PSO) has been used to solve optimization problem [8]. As a population based stochastic optimization technique, PSO was inspired by social behavior of bird flocking. PSO cannot guarantee the global optimization, but it generates high-quality solution with less calculation time and stable convergence characteristic [10]. It has been introduced to trajectory planning for the shortest path in a static environment without any obstacles [9]. In addition, augmented Lagrangian particle swarm optimization (ALPSO) has been successfully applied to find the shortest path in an environment with static obstacles [5]. In this paper, we will extend the application of ALPSO to find shortest path to avoid multiple dynamical obstacles for a group of UGV while keeping the V-shape of the group.

## II. PARTICLE SWARM OPTIMIZATION

In the PSO algorithm, a particle means one single individual in the swarm and swarm is the entire collection of particles. Each particle has different potential velocities and, therefore, has different future positions. Each particle also represents a candidate solution of the problem. Goodness of a solution is evaluated by a performance index, namely, fitness function. The location of a specific particle with the best fitness is personal best (pbest). The location of the entire swarm with the best fitness is global best (gbest). Evaluation of the fitness function will update pbest and gbest position. The updating laws for every particle's velocity $v_{i,j}$ and position $x_{i,j}$ in the swarm are written as equations (1) and (2)

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1 rand(1)[pbest - x_{i,j}(t)] + c_2 rand(1)[gbest - x_{i,j}(t)] \quad (1)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1). \quad (2)$$

where $x_{i,j}(t)$, $v_{i,j}(t)$ be the current position and velocity of the j*th* dimension of the i*th* particle, $w$ is the inertia coefficient which slows the velocity over time to prevent explosions of the swarm and ensures ultimate convergence. Parameters $c_1$ and $c_2$ are the weights given to the attraction to the previous best location of the current particle, and of the particle neighborhood. Function $rand(1)$ means random number within the interval of $[0, 1]$. The dimension of the motion space is defined as the minimum number of coordinates needed to specify a point. In a Cartesian space, the dimension is 2.
A PSO algorithm is conducted as follows.

*1)* Determine the number of particles in a swarm and initialize the position and velocity of each particle at $x_{i,j}(0)$



978-1-4244-2794-9/09/$25.00 ©2009 IEEE

and velocity vector $v_{i,j}(0)$ randomly. At time 0, set the initial position as the pbest of each particle and the particle with minimum fitness function has gbest position.

*2)* Update the velocity vector $v_{i,j}(t+1)$ and position vector $x_{i,j}(t+1)$ of particle $i$ by (1) and (2) at time $t+1$.

*3)* Calculate the fitness at $x_{i,j}(t+1)$. If it is better than the fitness of pbest, the position vector $x_{i,j}(t+1)$ is set as pbest. If a fitness of pbest is better than fitness of gbest, the position vector pbest is set as gbest.

*4)* Go to step 2 and start a new iteration until the fitness reaches a desired value or the iteration reaches the predetermined values.

An improved PSO algorithms has been presented by Clerc [12] to guarantee the convergence of the PSO algorithm using a constriction factor $K$ as shown in equation (3)

$$v_{i,j}(t+1) = K[wv_{i,j}(t) + c_1 rand(1)(pbest - x_{i,j}(t)) + c_2 rand(1)(gbest - x_{i,j}(t))]. \quad (3)$$

The convergence speed of PSO algorithm can be adjusted by $K$, which will be discussed in section V.

## III. PROBLEM FORMULATION

In this section, collision free trajectory planning for a UGV group will be formulated into a mathematical description and then combined with shape formation constraints. The outcome of this study is to design a tool to guide a group of UGV to travel from its current position to a desired position in an environment with dynamic and static obstacles while maintaining its shape formation.

### *A. Representation of vehicles and obstacles*

Due to the heterogeneous shape of the unmanned ground vehicles and obstacles, a circle is used to enclose a vehicle or an obstacle. The center of a circle is considered as the position of an object and radius of a circle is denoted as the size of an object. Thus, vehicles and obstacles can be described by their positions in a Cartesian space and sizes as $<x_r, y_r, r_r>$ for vehicle, and $<x_{obi}, y_{obi}, r_{obi}>$ for the i*th* obstacle.

### *B. Obstacle avoidance constraint*

The necessary requirement for a collision free motion of two objects is to keep the distance between the centers of the two objects larger than the summation of their sizes. To further simplify the representation of the collision free condition, one can add the size of a vehicle to the obstacle where motion of a vehicle is diminished as one point. Thereby distance requirement of collision-free is fully imposed to the virtual size of obstacles as $r_r + r_{obi}$. For safety consideration, a bumper range $\varepsilon$ can be added to the virtual size of the obstacles also.

### *C. V-shape formation constraint*

A V-shape formation has a vertice and two edges. In this study a leader vehicle will be placed at the vertice and all followers in the UGV group will be placed on the edges. Span of the V-formation can be changed when the UGV group meets obstacles. As shown in Fig. 1, once the group avoids the obstacles, it returns to the original V-shape formation. Moving direction of the group is defined by $\theta$ and V-formation of the group is characterized by $\varphi$.

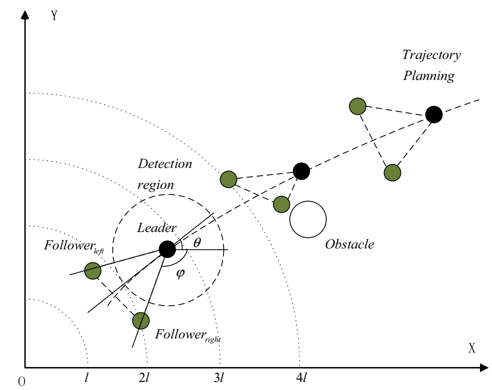


Figure 1. Illustration of span change of a V-shape UGV group with three vehicles in obstacle avoidance.

Given the position of leader $O_{leader}(O_{leaderx}, O_{leadery})$, the distance $O_{leader}O_{follower_r} = O_{leader}O_{follower_l} = d$ and the angle $\varphi$ are fixed, the positions of followers on the right and left edges to maintain V-formation can be calculated as

$$(O_{follower_{rx}}, O_{follower_{ry}}) = (O_{leaderx}, O_{leadery}) - (\sin(\varphi - 90), \cos(\varphi - 90))d, \quad (4)$$

$$(O_{follower_{lx}}, O_{follower_{ly}}) = (O_{leaderx}, O_{leadery}) - (\cos(2\theta + \varphi - 180), \sin(2\theta + \varphi - 180))d, \quad (5)$$

where $O_{follower_l}$ and $O_{follower_r}$ are the positions of the left and right followers.

## IV. TRAJECTORY PLANNING OF A UGV GROUP WITH PSO

Without losing generality, four assumptions have been made while we carry out the trajectory planning for a UGV group. 1) Velocity and position of each vehicle in a UGV group is detectable and shared in the UGV group. 2) The maximum available velocity of each vehicle in the UGV group is known. 3) The leading vehicle can detect the position and velocity of the obstacles. 4) Size of the vehicles and obstacles are known as priory. Due to the latest sensor technology, these assumptions can be easily satisfied in real applications. In addition, in a real application, the leader has limited detection range R. Therefore if an obstacle is out of the detection range, the collision free constraint on this obstacle will not be considered.

To apply the PSO algorithm, the dynamical environment is separated by concentric circles with radii $nl$ to $n$ subspace. $n$ is the positive integer. The increase of the radius $l$, is determined by the maximum velocity of a vehicle and obstacles.

$$l \le (\max\{\vec{v}_{obi}\} + \vec{v}_r)\triangle t. \quad (6)$$

$\vec{v_{obi}}$ is the velocity of i*th* obstacle and $\vec{v_r}$ is the velocity of a vehicle, $\triangle t$ is the updating time for obstacle avoidance. In the searching region including the origin, we choose the number of particles is chosen as 20. In order to have the same particle density in every region, the number of particles is $20n^2$ in the n*th* region.

*A. Collision free path of the leader*

*1)* Collision avoidance strategy for a single vehicle

Collision free constraint means at a specific time point, the vehicle's position ($x_r$,$y_r$) cannot stay in the range of imposed region of the obstacle, i.e. the circle centered at $(x_{obi}(t), y_{obi}(t))$ with radius $(r_r + r_{obi} + \varepsilon)$. Since PSO algorithm determines the motion in a small period of time (iteration), the collision free constraint is illustrated in Fig. 2. Within each trajectory updating period $(n\triangle t, (n+1)\triangle t)$, travel of an obstacle covers a rectangular area. Height of the rectangle is $v_{obi}\Delta t + 2(r_r + r_{obi} + \varepsilon)$, and width of the rectangle is $2(r_r + r_{obi} + \varepsilon)$. $O_r(t)(O_{rx}(t), O_{ry}(t))$, $O_r(t)(O_{rx}(t+1), O_{ry}(t+1))$ are the positions of the UGV group at time $t$ and $t+1$; $v_r(t)(v_{rx}(t), v_{ry}(t))$ is the velocity of a vehicle; $v_{obi}(t)(v_{obix}(t), v_{obiy}(t))$ is the velocity of the i*th* obstacle in Cartesian space.

Within the period of $\triangle t$, the vehicle located at $O_r$ and obstacle located at $O_{obi}$ might arrive the same position on the X direction at

$$\triangle t_{left} = \frac{O_{rx}(t) - x^i_{left}(t)}{v_{rx}(t) + v_{obix}(t)}. \quad (7)$$

At $\triangle t_{left}$, the left edge of the obstacle path on y-coordinate is

$$y_{leftA} = y^i_{left}(t) + v_{obiy}(t)\triangle t_{left}. \quad (8)$$

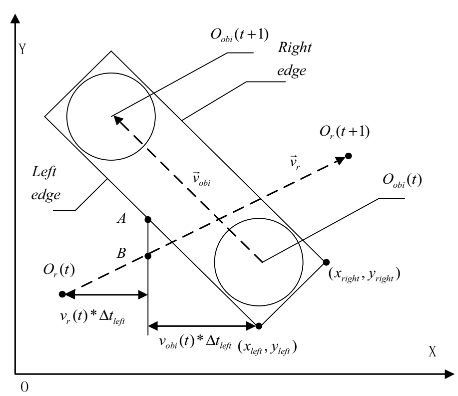


Figure 2. Using PSO to avoid obstacles.

At $\triangle t_{left}$, the coordinate of the vehicle on y-coordinate is:

$$y_{leftB} = O_{ry}(t) + v_{ry}(t)\triangle t_{left}. \quad (9)$$

Similarly, trajectory of the vehicle and the right edge of ith obstacle might also meet on x-axis at $\triangle t_{right}$, and coordinates of the obstacle and vehicle at $\triangle t_{right}$ are $y_{rightA}$, $y_{rightB}$, which can be obtained using similar equations as (7-8). The condition of collision free path is:

$$(y_{leftA} - y_{leftB})(y_{rightA} - y_{rightB}) > 0. \quad (10)$$

*2)* Fitness function for a single vehicle

Fitness function quantifies the optimality of a solution in the PSO and is the criteria to select particles as the candidates to the global best. An appropriate fitness function, which is based on the algorithm's goal, is the key of the PSO algorithm. For a single vehicle, the fitness function minimizes the distance between the candidate solution and desired point and can be written as:

$$F_{leader} = min \sum_{i=1}^{k} ((O_{rx}(t+1) - x_{desired})^2 + (O_{ry}(t+1) - y_{desired})^2), \quad (11)$$

where $(x_{desired}, y_{desired})$ is the coordinate of desired point and k is the number of current region.

*3)* PSO algorithm with constraint for a single vehicle

The updating law of a PSO algorithm was described as (1)-(3) in section II. The purpose is to find the global best point that can minimize the fitness function (11) with the constraint (10). The PSO algorithm with constraint for a single vehicle is presented as follows:

*a)* Get the environment information from the sensors included the size, position and velocity of the obstacles. Initialize the work space, position and velocity of the vehicle, pbest and gbest.

*b)* Initialize $20n^2$ particles with randomly chosen positions in the nth region. Evaluate each position by fitness function according to equation (11) in each sub-space.

*c)* Update the position and velocity by updating laws (1) and (3) for each particle, if these particles do not satisfy the collision free requirement described in (10), these particles can be ignored. Otherwise, compare fitness functions of these particles with their previous pbest. If the new fitness functions of these particles are better, set their personal best point. If the fitness of personal best point is better than fitness of current global best point, set this personal best point as the global best point.

*d)* After 50 iterations, stop the searching.

*e)* With steps a-d, the trajectory for the leader is determined in its current subspace. Repeat step a-d while the leader enters a new subspace, until the vehicle arrives the desired point.

*B. Collision free path for a group of UGV*

*1)* Physical constraint and V-shape formation constraint

To keep the V-shape formation on a collision free path, the followers stay in a limited range centered at the position of the leader, and there are two physical constraints between the leader and followers and between the two followers. The physical constraints mean

$$d = O_{follower}O_{leader} \geq r_{follower} + r_{leader} \quad (12)$$

$$O_{follower_l}O_{follower_r} \geq 2r_{follower}, \quad (13)$$

where $O_{leader}$, $O_{follower}$ are the central points of the leader and follower; $r_{leader}$ and $r_{follower}$ denote the size of the leader and follower.

Given the optimal position of the leader, the desired position of a follower $O_{desiredf}$ to keep the V-shape formation is known by (4) and (5). To keep the V-shape, the angle $\varphi$ is constant, span of the edges $d$ is adjustable but has to stay in the V-shape formation region.

$$d = O_{follower}O_{leader} \leq r_{Group}, \quad (14)$$

where $r_{Group}$ is the predefined size of the UGV group. The formation constraint has been rewritten as a distance constraint, which leads to some flexibility of the span of the V-shape. With such flexibility, the group will keep the V-shape in general but not the exact formation all the time. The reduced formation constraint is very important while the UGV group passes through a doorway.

*2)* Fitness function for a UGV group

The goal of PSO for a UGV group is to find the shortest path for the whole group and keep the V-shape formation as long as possible. The fitness function includes evaluations for both the leader and followers. For the leader, the fitness function evaluates the distance between the candidate solution and desired point

$$F_{leader} = min \sum_{i=1}^{k} ((O_{rx}(t+1) - x_{desired})^2 + (O_{ry}(t+1) - y_{desired})^2), \quad (15)$$

where $(x_{desired}, y_{desired})$ is the coordinate of desired point.

For the followers, the fitness function evaluates the distance between the candidate solution of follower and the desired follower point.

$$F_{follow} = min[(O_{followerx}(t+1) - O_{desiredfx}(t+1))^2 + (O_{followery}(t+1) - O_{desiredfy}(t+1))^2] \quad (16)$$

$(O_{followerx}(t+1), O_{followery(t+1)})$ is the potential point of follower at $(n+1)\triangle t$. The fitness function for the UGV group is the summation of the leader and followers.

*3)* PSO algorithm with constraint for a UGV group

The aim of the PSO algorithm for the UGV group is to find the global best point that can minimize both the fitness function (15) with the constraint (10) for the leader and the fitness function (16) with the constraints (12-14) for the followers. The algorithm is described as follows.

*a)* Carry out the PSO algorithm for single vehicle to design the trajectory for the leader.

*b)* Carry out the PSO algorithm for two followers in the UGV group to find the optimal point for followers.

*c)* Repeat step a-b while the leader enters a new subspace, until the vehicle arrives the desired point.

## V. STABILITY OF PARTICLE SWARM OPTIMIZATION

Stability analysis of the PSO algorithm was conducted based on the difference equations defined in the equations (2-3).

By defining a nominal best $p$, and an error $\varepsilon(t)$ between the nominal best and the current position

$$p \leftarrow \frac{\xi_1 pbest + \xi_2 gbest}{\xi}, \quad (17)$$

$$\varepsilon = p - x(t), \quad (18)$$

where $\xi_1 = c_1 rand(1)$, $\xi_2 = c_2 rand(2)$, and $\xi = \xi_1 + \xi_2$.

The dynamic system can be rewritten as

$$v(t+1) = K[v(t) + \xi\varepsilon(t)], \quad (19)$$

$$\varepsilon(t+1) = -v(t) + (1-\xi)\varepsilon(t). \quad (20)$$

Based on the equation (19-20), the updating law of error equation and velocity equation are derived as

$$\varepsilon(t+2) - (K - \xi + 1)\varepsilon(t+1) + K\varepsilon(t) = 0, \quad (21)$$

Characteristic equation of difference equation is

$$\lambda^2 - (K - \xi + 1)\lambda + K = 0. \quad (22)$$

Given the roots of Characteristic equation as $\lambda_1$ and $\lambda_2$, the general solution is:

$$\varepsilon(t) = \alpha_1 \lambda_1^t + \alpha_2 \lambda_2^t, \quad (23)$$

where $\alpha_1$ and $\alpha_2$ are the coefficient of the general solution. The convergence condition is

$$|\lambda_1| < 1, |\lambda_2| < 1. \quad (24)$$

Thus, when the convergence condition is satisfied, the particle will converge to the nominal best and stay there. From the equation (22),

$$\lambda_1 + \lambda_2 = K - \xi + 1, \quad (25)$$

$$\lambda_1 \lambda_2 = K, \quad (26)$$

therefore, the stable region is $K < 1$ and $-3 < K - \xi < 1$.

## VI. SIMULATION RESULT

Trajectory planning for a single vehicle and a UGV group has been performed using PSO algorithm with MATLAB. We will show 4 cases in this section.

Case 1: Trajectory planning for a single vehicle with two moving obstacles without detection range limit.

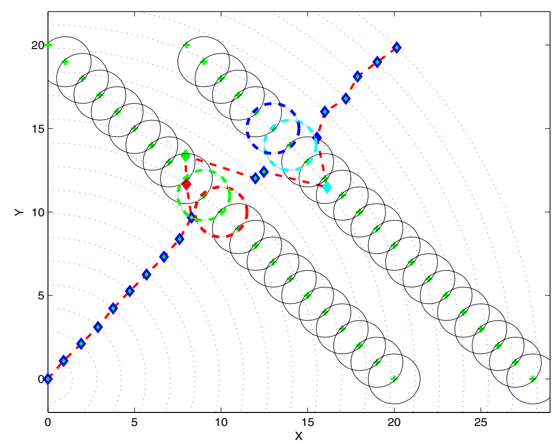


Figure 3. Collision free path for a single vehicle without range limited.

In this case, no detection range is imposed on the vehicle, i.e., the vehicle knows all information about the obstacles in the whole space. The size of the vehicle, the obstacles, and the bumper range are defined as $r_r = 0.4$, $r_{obi} = 0.4$ and $\varepsilon = 0.2$. Velocities of the obstacles and vehicle are chosen as $\vec{v}_{ob1} = [-1, 1]$, $\vec{v}_{ob2} = [-1, 1]$, and $v_r \leq 2max\{v_{obi}\}$. Initial positions of obstacles are chosen as $x_{ob1} = [20, 0]$, $x_{ob1} = [28, 0]$. The vehicle will start from the position [0, 0], and the desired destination is located at $x_{desired} = [20, 20]$. Given $\Delta t = 1$, the working space unit is calculated as $l = \sqrt{2}$

from (6) ; In updating law, parameters are chosen as $w = 0.75$, $c_1 = c_2 = 2$ , $K = 1$ . They satisfy the condition in the equations (3, 25, 26).

The simulation result is shown in Fig. 3. The black circles denote the virtual sizes of the obstacles and the green plus signs are the center of the imposed obstacle in each period. The red, green, cyan and blue circles denote the potential collision positions. The red dash line is the trajectory of the vehicle and the blue diamonds represent positions of the vehicle at each updating time point. PSO generated a collision free path for the vehicle with path length $L_1 = 35.60$.

Case 2: Trajectory planning for a single vehicle with two obstacles and detection range limit.

In this case, we compare the effects of detection range on the trajectory planning. Initial conditions are same as those in case 1. The vehicle will find its optimal trajectory without considering the obstacle avoidance constraint until the obstacle goes into the detection region.

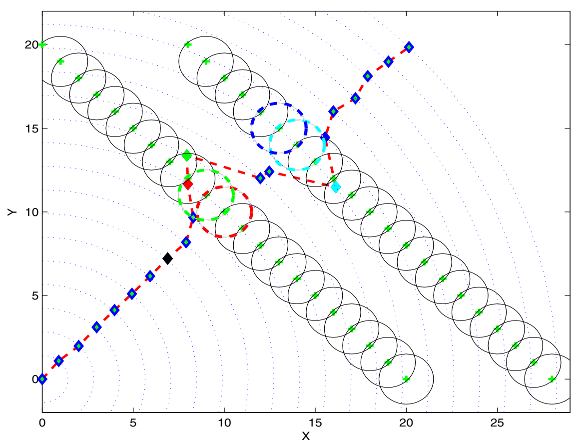


Figure 4. Collision free path for a single vehicle with range limited (range=3).

With the detection region set as 3, the vehicle changes its direction at the $8th$ step as shown in Fig. 4. In this step, the vehicle detects the obstacle and is guided by the PSO algorithm with collision free constraint. The length of the path is $L_2 = 35.74$.

With the detection range set as 7, the vehicle detects the obstacle and the PSO algorithm starts to work at the $5^{\text{th}}$ step. The length of the path is $L_3 = 35.63$. If the detection range is increased, the vehicle can get more information of the obstacle in the working space and detect the obstacle earlier. The abundant information of the obstacles leads to a shorter path in Fig. 5 than the path in Fig. 4, and the best path is demonstrated in Fig. 3.

Case 3: Trajectory planning for a UGV group with two moving obstacles.

Initial conditions in this case are similar as those in case 1 except that the initial positions of followers are $x_{fleft} = [0, -1]$, $x_{fright} = [-1, 0]$. The desired positions of followers are $x_{desiredfleft} = [20, 19]$, $x_{desiredfright} = [19, 20]$, and the size

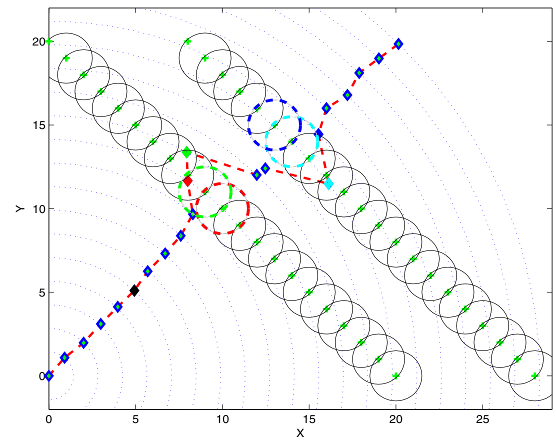


Figure 5. Collision free path for a single vehicle with range limited (range=7).

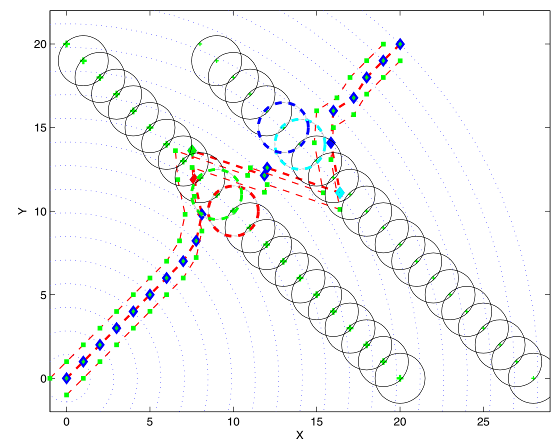


Figure 6. Collision free path of a UGV group without V-formation flexibility.

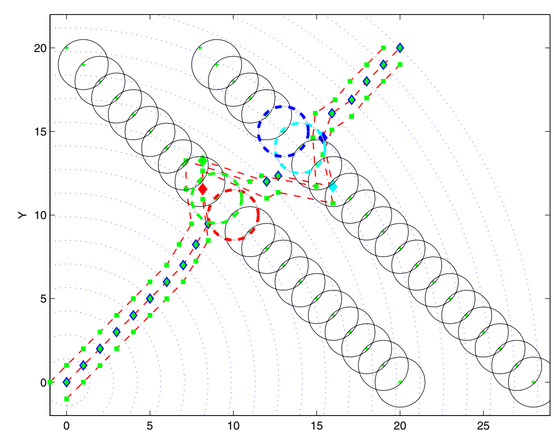


Figure 7. Collision free path of a UGV group with V-formation flexibility.

of the group is chosen as $r_{Group} = 1.2$ . If there is no formation flexibility, the leader and followers can be considered as a bigger vehicle, the radius of the imposed obstacle is $r_{Group} + r_{obi} + \varepsilon$. The trajectory is shown in Fig. 6, where the green squares are the followers of the UGV group. If there is some formation flexibility, the V-formation of the group was adjusted while avoiding the obstacles as shown in Fig. 7. The right follower moves closer to the leader, which generates a shorter path than the path demonstrated in Fig. 6.

Case 4: Trajectory planning for a UGV group with two moving obstacles constraints in the same period of time.

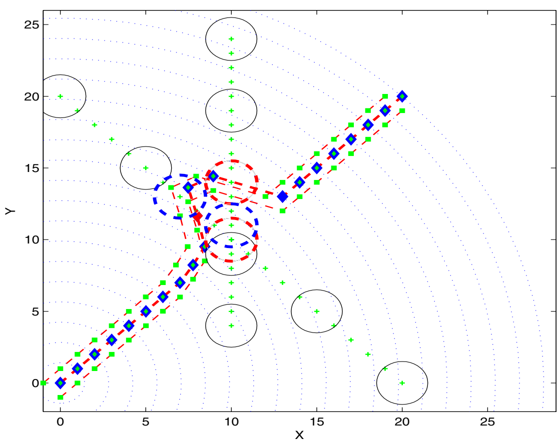


Figure 8. Collision free path for a UGV group without changing V-formation.

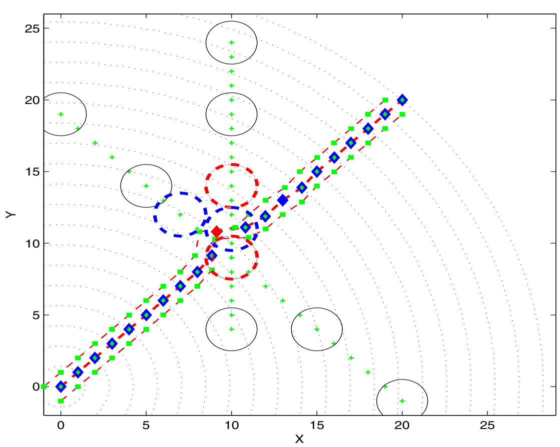


Figure 9. Collision free path for a UGV group without changing V-formation.

The initial positions of obstacles are set as $x_{ob3} = [20, 0]$, $x_{ob4} = [10, 24]$. The velocities of obstacles are chosen as $\vec{v}_{ob3} = [-1, 1]$, $\vec{v}_{ob4} = [0, -1]$. In this case, we demonstrated the trajectories for the UGV group without formation flexibility in Fig. 8 and with formation flexibility in Fig. 9. In Fig. 8, the UGV group is considered as a bigger vehicle and the UGV group cannot go through the space between two obstacles at the red point, therefore, following a longer path to the desired point. The UGV group can adjust the V-formation to avoid the obstacle, pass the space between two obstacles and follow a shorter trajectory as shown in Fig. 9. Comparison of the simulation results illustrates the effectiveness of the proposed PSO algorithm.

## VII. Conclusion

In this paper, path panning for the UGV group is formulated as an optimal problem with constraints on obstacle avoidance and V-Shape formation. PSO algorithm was applied to solve the optimal problem due to its fast convergence. We have illustrated that single vehicle can find optimal path by the PSO algorithm with obstacle avoidance constraint. We also extended the application of PSO algorithm for a UGV group with obstacle avoidance and V-formation constraints to find the collision free trajectory in a dynamical environment. Convergence of the PSO algorithm was analyzed to determine the proper region of the parameters. The simulations demonstrated that the PSO algorithm shortens the path effectively while keeping the V-formation with constraints.

## Acknowledge


The authors acknowledge grant support from NSF 0649172 and AT&T foundation (to YJ).


## References


[1] Hattori, Y., E. Ono, and S. Hosoe, “Optimum Vehicle Trajectory Control for Obstacle Avoidance Problem. Mechatronics,” IEEE/ASME Transactions on, 2006, 11(5): pp. 507-12.

[2] Borenstein, J. and Y. Koren, “Obstacle avoidance with ultrasonic sensors”. Vehicleics and Automation, IEEE Journal of, 1988, 4(2): pp. 213-18.

[3] Goldman, J.A. “Path planning problems and solutions,” in Aerospace and Electronics Conference, 1994. NAECON 1994., Proceedings of the IEEE 1994 National. 1994.

[4] Lee, S. and G. Kardaras. “Collision-free path planning with neural networks,” in Vehicleics and Automation, 1997. Proceedings., 1997 IEEE International Conference on 1997.

[5] Fraichard, T. “Dynamic trajectory planning with dynamic constraints: A `state-time space' approach,” in Intelligent Vehicles and Systems 1993, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on. 1993.

[6] Jia, W., W. Xiao-Bei, and X. Zhi-Liang. “Decentralized Formation Control and Obstacles Avoidance Based on Potential Field Method,” in Machine Learning and Cybernetics, 2006 International Conference on 2006.

[7] Sisto, M. and G. Dongbing. “A Fuzzy Leader-Follower Approach to Formation Control of Multiple Mobile Vehicles,” in Intelligent Vehicles and Systems, 2006 IEEE/RSJ International Conference on 2006.

[8] Yanjun, Y. and L.A. Osadciw. “Varying dimensional particle swarm optimization,” in Swarm Intelligence Symposium, 2008 IEEE. 2008.

[9] Li, W., L. Yushu, D. Hongbin, and X. Yuanqing. Obstacle-avoidance Path Planning for Soccer Vehicles Using Particle Swarm Optimization. in Vehicleics and Biomimetics, ROBIO 2006. IEEE International Conference on 2006.

[10] Kadirkamanathan, V., K. Selvarajah, and P.J. Fleming, “Stability analysis of the particle dynamics in particle swarm optimizer,” Evolutionary Computation, IEEE Transactions on, 2006, 10(3): pp. 245-55.

[11] Tae-Hyoung, K., I. Maruta, and T. Sugie. “Particle swarm optimization based robust PID controller tuning scheme,” in Decision and Control, 2007 46th IEEE Conference on 2007.

[12] Clerc, M. and J. Kennedy, “The particle swarm - explosion, stability, and convergence in a multidimensional complex space,” Evolutionary Computation, IEEE Transactions on, 2002, 6(1): pp. 58-73.