Robot Path Planning for Multiple Robots Considering safest and shortest path

Zhanna Gabbassova[1] and Davoud Sedighizadeh[2]

[1] Department of software engineering, Atyrau State University named after Khalel Dosmukhamedov, Kazakhstan
[2] Department of Industrial Engineering, College of Engineering, Saveh Branch Islamic Azad University, Saveh, Iran
lncs@springer.com

Abstract. In this paper, a new algorithm is developed for multi-robot path planning of mobile Robot in unknown environment. The robots use radiation from their sensors to detect their surroundings, as well as positions of the other robots. In this paper a new approach for simultaneous consideration of two objectives is extended. The first objective is finding the safest path and another one is finding a path with minimum length. Voronoi Diagram is applied to achieve the safest path. In order to provide the safest path, we try to minimize the distance to the Voronoi Diagram (VD). Because the VD is a geometric location that is a distance from all obstacles to the workspace, it is therefore a safe place in terms of distance to obstacles. For gaining the second objective, shortest path, Euclidean distance from the current position of the robot to target is used. For solving this problem, it is used from particle swarm optimization (PSO).

Keywords: Multi Robot Path Planning, Particle Swarm Optimization, unknown environment

1 Introduction

The general Robot Motion Planning (RMP) problem deals with finding a collision-free path for a robot from a start path to a goal path, in a workspace containing multiple obstacles considering an objective function. In modeling this issue, the more we bring things closer to the real world, the accuracy of the model increases when it is used in the real world. It is also assumed that the problem has different objective, with two objectives being the safest and shortest path. Finally, the problem is considered online and it has been tried to make real world affairs possible under the assumptions of the problem.

For the first time, online RMP for multiple robots is expanded in [1]. After that, it was more extended in [2]. In this paper, it was used form a hieratical coordinator for a systematic design procedure in a multiple robot system. This work aided to reduce running time of the planner. In 2003, a decentralized motion planning for multiple robots subject to sensing and communication constraints are expanded [3]. The goal of this paper is reaching each robot to its goal keeping connectivity with the neighbors. In 2016, a model is developed for online Multi-robot MP using a modified grav-

ICID-2019 Conference

itational search method [4] Furthermore, an adaptive multi-objective PSO is developed for multi-RMP [5]. In this algorithm, five robots are considered for path planning and two objectives mentioned are including minimum length path and maximum distance from the danger zones. Additionally, in 2017, a multi-objective multiple robot is implemented in an online situation [6]. In this work, the problem is named deployment problem. Two objectives are considered such as estimation of final positions that robots are reached and shortest path. In 2017, a multi-objective model is developed for multiple systems in two cases [7]. In the first case, two objectives are considered including finding a minimum distance to ward goal pints for two robots and in the second case, shortest and smoothness path for three and four robots. In 2019, a multi-objective model is presented for Multi-Robot M [8]. Environment in this model is continues and offline that a proposed Artificial Potential Filed (APF) is used to build all feasible path for guarantee at least one feasible path. An enhanced Genetic algorithm is applied for obtaining optimal path. In this algorithm, are used form five new crossover and mutation operators. The objectives in this work are path length, smoothness, and safety. In most articles, such as the article above-mentioned, the greatest attention is paid to the minimum path length, and sometimes the smooth or safe path is considered. In addition to the above mentioned, in our article, the minimum length is also considered as the second objective. Furthermore, the environment is assumed offline, and the criteria or objective function considered for the safest is different. In addition, we examined more different complex workspace for testing our model and algorithm.
The rest of this paper is organized as follow. Section 2 states the proposed objective function. Then, the problem formulation is discussed in Section 3. Afterwards, Section 4 explains the proposed algorithm and numerical results and finally, conclusions are given in Section 5.

2 Objective Function

Total Objective Function including safest and minimum length on the path

$$fitness_j^i = \lambda_1 \times {fitness_1}_j^i + \lambda_2 \times {fitness_2}_j^i \tag{1}$$

By minimizing the general objective function and taking into account the appropriate weights allocated to each criterion, a suitable path is obtained. The weights for the safest and shortest length criteria are λ_1 and λ_2 respectively. After performing the simulation and trying and error, the best values for them were $\lambda_1 = 1$, $\lambda_2 = 0.25$.

2.1 Model Description

Since paths generated by a program must be run by robots, and each robot has its own static and dynamic constraints, the program should generate a path that is collision-free and efficient in relation to some performance criteria. The main performance indicators and constraints inherently part of a motion planning is listed in Table 1.

2.2 Multi-objective function of the problem

The majority of motion planners are designed to produce an optimal path by considering a criterion similar to the time path or path length. However, in practice, it is a feasible path if it meets two criteria including safety and shortest length, and so on.
A path that is defined as an optimal path with a particular criterion may not necessaril y be optimal, for example, in terms of safety or other criteria [9].

Table 1 Performance Indicators and Constraints on Path Planning

Constraints	Performance indicators
Position, speed, acceleration and shocks of joints	Time tracking
The forces and dynamics of the stimuli	Speed of links and joints
Kinematics	Length of path
Collision with obstacles	

There are some papers in the Multi Objective Robot Motion Planning [10]. In this work, the second-order motor model is used that helps the robot to Obstacle avoidance. In this model, information is included such as the goal position of the robot and the direction and speed of the obstacles. In addition, the problem has been discussed in several ways, but of course, there are two goals that are counted in a weighted aggregated approach. However, in these two goals, this is in fact a kind of obstacle avoidance. Motion planning in dynamic environments focuses on the need to consider several goals in motion planning.
The most important performance indicator is the time it takes to navigate. To find it and some other factors, several equations will be extracted. First, assume that the path consists of a number of discrete sections connected together to make the path of motion. Moving along the path will have several characteristics, and these will form the basis of a series of equations. The important phrases involved in the proposed model are described in detail below.

3 Formulation

In this section, we describe the formulation as below.

3.1 Safest Path

In order to provide the safest path, we try to minimize the distance to the Voronoi Diagram (VD). Because the VD is a geometric location that is a distance from all obstacles to the workspace, it is therefore a safe place in terms of distance to obstacles. Initially, the description of this algorithm is presented. The Voronoi diagram: The fascinating truth about the Voronoi diagram is that its history dates back to the seventeenth century, When Descartes used a subjectivity to describe the structure of solar systems. Later mathematicians such as Dirichlet and Voronoi were principally the first to introduce this approach. They used it for quadratic shapes in which the branches, as well as the points, are measured in a regular diagram and measured by Euclidean distance. The structure of the result is under titles: Dirichlet Tessellation and Medial axis and Voronoi diagram, which is the standard name for this approach today [11].

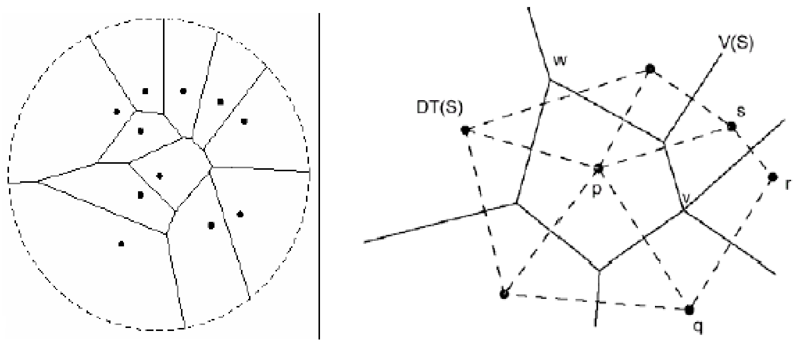

Fig. 1 (a) Voronoi diagram and triangular pieces, (b) A Voronoi diagram consisting of 11 points on the Euclidean page

In order to give a mathematical representation of VD; assume

$$d(p,x) = \sqrt{(p_1 - x_1)^2 + (p_2 - x_2)^2} \qquad (2)$$

that indicates Euclidean distance of $p=(p_1,p_2)$ and $x=(x_1,x_2)$. Suppose pq is the line segment p to q. Enclosed set A is shown as A . For p, $q \in S$, assume

$$B(p,q) = \{x \mid d(p,x) = d(q,x)\} \qquad (3)$$

is bisector p and q. $B(p, q)$ is a vertical line from the center of the line segment pq. B is the half-panel separator:

$$D(p,q) = \{x \mid d(p,x) < d(q,x)\} \qquad (4)$$

contains p of the half-panel $D(p,q)$ containing q. We will say that V is the V region relative to S. Thus,

$$VR(p,s) = \bigcap_{q \in s, q \neq p} D(p,q) \qquad (5)$$

that is region p Voronoi related to S. Finally, the Voronoi diagram for S is:

$$V(S) = \bigcap_{p,q\in s,\, p\neq q} \overline{VR(p,s)} \bigcap \overline{VR(q,s)} \tag{6}$$

According to the above definition, each Voronoi region, such as *VR* (*p*, *s*), is a 1-*n* intersection of the half-open plate containing the *p* location. Therefore, *VR* (*p*, *s*) is open and convex. Voronoi areas are as distinct and discrete as shown in Fig. 1.
The remarkable point is that due to inaccuracy of the scanner, the barriers of the obstacles are not exactly recognized. The generalized local VD obtained from the VD is based on the algorithm [12]:
After VD is drawn up in the free space, it tries to achieve the minimum value in the objective function of the safest path of the distance between each particle and VD:

$$\begin{gathered} d(p,x) = \sqrt{(p_1 - x_1) + (p_2 - x_2)} \\ d(p,q) = \{x \mid d(p,x) < d(q,x)\} \\ VR(p,s) = \cap_{q\in s, q\neq p} D(p,q) \\ VS = \bigcap_{q\in s, q\neq p} \overline{VR(p,s)} \bigcap \overline{VR(q,s)} \end{gathered} \tag{7}$$

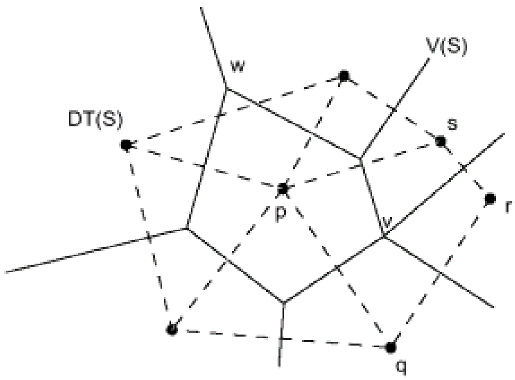

Fig. 2 Voronoi Diagram and Dirichlet

In the above relation, x is the sum of the geometric locations of the points on VD, which aims at minimizing the distance between the current position and VD, thus:

$$fitness_{1_j}^{\,i} = \sqrt{(prtpos_j^i(x) - x_1)^2 + (prtpos_j^i(y) - x_2)^2} \tag{8}$$

3.2 Shortest Path

In this paper, two objectives are considered that are including safest path and the shortest path as below:

Most path planners aim to generate an optimal path considering a single criterion like path travel time or path length. However, in practice, a path is feasible if it meets several conditions, such as safety, estimated needed time for navigation, shortest length, etc.

A path which is considered as optimal in terms of a single criterion may not essentially satisfy other criteria all together [9]. For instance, a shortest path is not required at the expense of safety along the path.

Some works exist in the field of multi-objective robot motion planning, including an approach for obstacle avoidance with multi-objective optimization by PSO in dynamic environment in [10], and multi-objective optimal trajectory planning of a space robot using PSO in [13].

Path planning in dynamic environments epitomizes the necessity of considering multiple objects in path planning: when the environment is time varying, the minimum length path and minimum delay path are usually different issues. Delay is defined as the time needed for traveling from start to goal, whereas length is the distance actually traveled by the robot along the path.

For robots needing to reach their destination as early as possible, a minimum-time path might seem desirable, but it may require a lot of time to be traversed due to uneasy terrain. Surely there exist various feasible paths between start and goal points being neither short nor fast but providing reasonable tradeoffs between shortness and fastness. These are generally desirable paths, while a path optimal for a single criterion without considering other equally important criteria is not desirable [9]. This is just one type of problem for which our multi-objective search is designed.

A common method for enforcing multiple objectives is the Simple Additive Weighting (SAW) method, in which a weighted sum of multiple objectives is expressed as a conventional single-objective function in the form of Total Cost = $w_1z_1 + w_2z_2 + \ldots$, where zi is the i-th cost and wi is its weight. By selecting proper weights, a path with desirable property can be obtained by planning with a single objective [14].

In the proposed method, the criterion for path shortness is defined as the Euclidean distance between each particle and the goal point in each iteration, and the criterion for path smoothness is defined as the angle between two hypothetical lines connecting the goal point to the robot's positions in two successive iterations, i.e. gbest i and gbest $i-1$, in which i is the iteration number. The definition of path smoothness in this way is a novel idea. The first objective function, the shortest path, is defined as:

$$F_{\mathrm{short}\,j}^{\;i} = \sqrt{(x_{prtpos_j^i} - x_{goal})^2 + (y_{prtpos_j^i} - y_{goal})^2} \qquad (9)$$

3.3 Total objective function

The aim of the total objective function is the simultaneous maximizing safety path and minimizing length of path. As mentioned above for the safest path, minimizing the distance to the geometric location of the points on VD is used and we use the aggregated weighted approach to minimize them simultaneously. In order to make the safest path, the total distance between the segments of the path and VD is considered. In order to minimize length of path, Euclidean distance from the current position of the robot to target of each segment is calculated. Each of these objective functions has a weight in the total objective function.

4 Computational results and analysis

In this section, it is assumed that the robot is not only unaware of its surroundings but also of its location. Therefore, the robot uses the two PSO and VD algorithms as the general and local search algorithms in the work environment to sense and identify the environment, respectively. The paper assumes that borderlines are static in the workplace and do not change over time.

How to use two of these algorithms in an online environment is that after sensing the environment by a robot, the sensor is in a visual environment that is circular to the radius of the robot's vision in its surroundings, a local and global search in the visible space is done. First, using the PSO algorithm, the best robot next position, which is the gbest, is determined in the visual environment.

Because in determining the gbest in the PSO algorithm, goal is secure path and shortest length, and these goals are presented in the PSO objective function. How to do it is that the gbest point is determined by the two goals of most safe path and minimum path length and the robot moves to the gbest point. In the other worlds, the gbest point obtained which satisfies the above-mentioned two goals.

Using the VD algorithm, which is applied as a local optimizer, the visible region is first obtained. How to use the Voronoi diagram, which is addressed as a first time in this paper, that's is that VD is using lines with an equal distance of the intersection points obtained from the collision point of the sensor with surrounding obstacles. An illustration of how to create a Voronoi diagram in each stage for the robot is shown in Fig. 3.

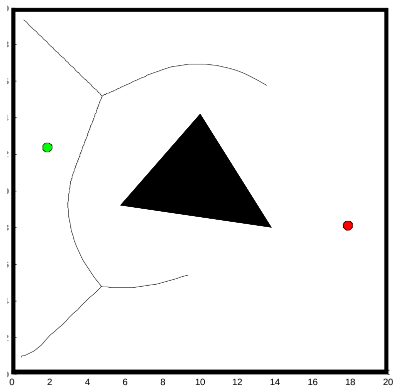

Fig 3. An overview of how to create a Voronoi diagram at each stage for the robot

For solving the problem and implementation it, we used from a hybrid of the algorithms titled Basic PSO (BAPSO) and VD. In each of the mentioned algorithms, the robot's motion is planned. In this way, the initial population is generated through related mechanism's BAPSO, then VD is established, and in the visual range, based on population upgrade mechanisms in the mentioned algorithms are searched.
Updating BAPSO is according equations (10) to (13).

$$prtvel_j^i = w \times prtvel_j^{i-1} + c_1 \times rand \times (pbest_j^{i-1} - prtpos_j^{i-1}) + c_2 \times rand \times (gbest^{i-1} - prtpos_j^{i-1}) \tag{10}$$

$$prtpos_j^i = prtpos_j^{i-1} + prtvel_j^i \tag{11}$$

$$prtpos_j^0 = x_{\min} + rand(x_{\min} - x_{\max}) \tag{12}$$

$$prtvel_j^i = \frac{x_{\min} + rand(x_{\min} - x_{\max})}{\Delta t} \tag{13}$$

After several tests for 14 problems, the mean and standard deviation of the results of the time and distance traversed for the the BAPSO + VD and Genetic Algorithms (GA) +VD method were obtained and the results are presented in Table 2 and the Figs (4) to (6) are given.

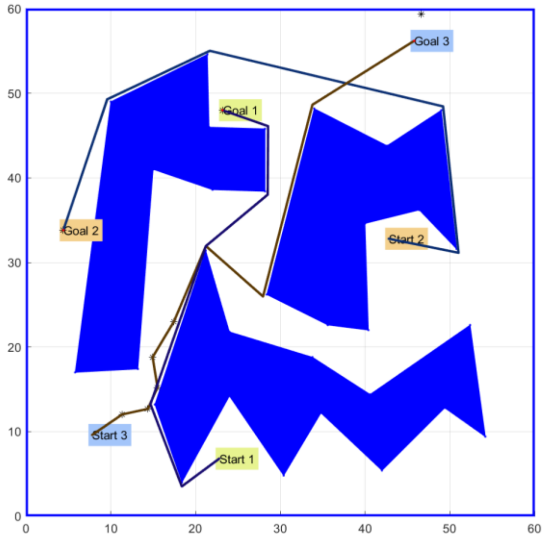

Fig. 4 shows a simulation from robot motion planning

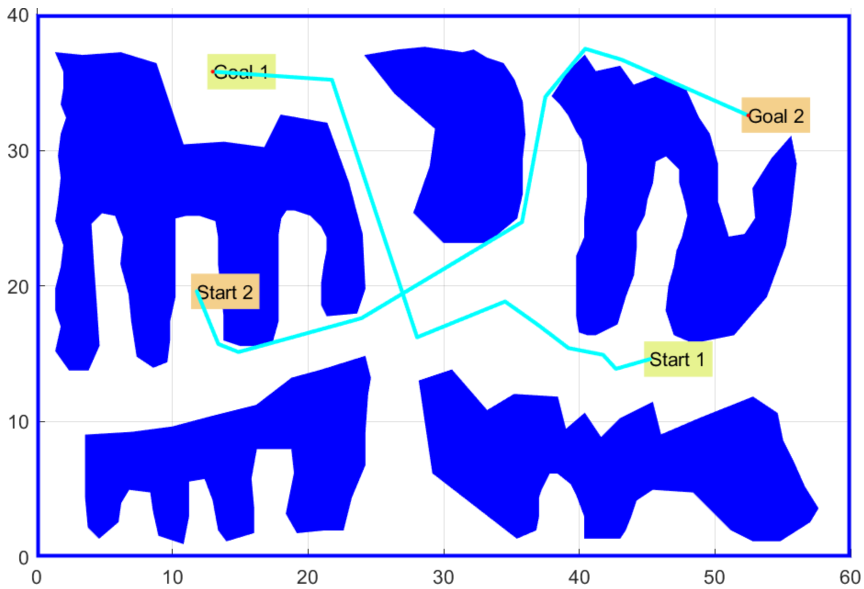

Fig. 5 shows a simulation from robot motion planning in the more complicated environment

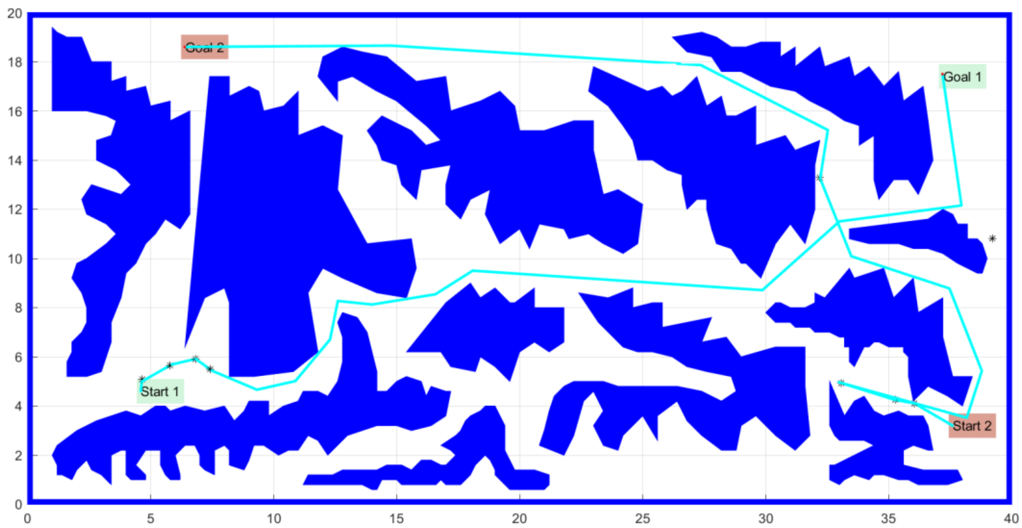

Fig. 6 shows a simulation from robot motion planning in the most complicated environment

Table 3 Details of time consumed

.	Two Robots	Three Robots	Four Robots	Five Robots	Six Robots	Seven Robots
BAPSO+VD	30.18	33.14	37.24	42.15	46.18	50.11
GA+ VD	43.12	47.11	51.16	58.19	60.23	66.18

In addition, updating mechanism in BAPSO is illustrated as Fig 7.

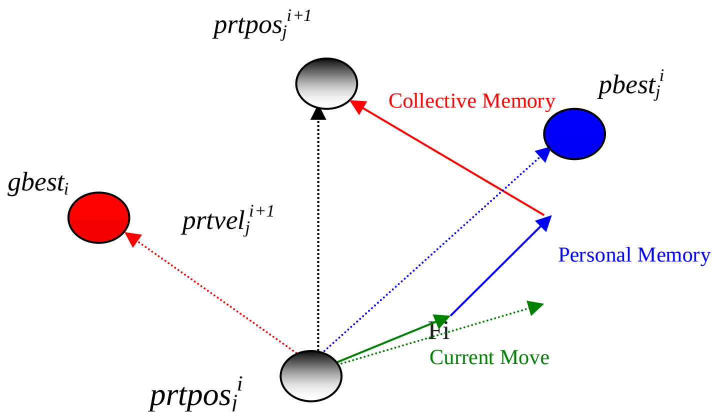

Fig. 7 updating mechanism in BAPSO

5 Conclusion

In this article, the problem of Motion Planning for mobile Multiple Robots Considering two objectives, the safest path with minimum of path length was analyzed. As the environmental conditions are timely. In fact, at this stage, for reaching to a safest path, we used from Voronoi Diagram for this objective. In addition, minimizing the length of path of the robot is also added to another problem objective and is considered in the objective function of the algorithm. In finally the problem is solved with two algorithm named BAPSO+VD and GA+VD that the results illustrated that BAPSO+VD have the better results.

References

1. Freund, E., Hoyer, H., Pathfinding in multi-robot systems: Solution and applications, Proceedings. IEEE International Conference on Robotics and Automation, (1986).
2. Freund, E., Hoyer, H., Real-Time Pathfinding in Multirobot Systems Including Obstacle Avoidance, the International Journal of Robotics Research, (1988).

3. Pereira, G. A. S., Das, A. K., Vijay Kumar, R. Campos, M. F. M. Decentralized motion planning for multiple robots subject to sensing and communication constraints, Departmental Papers (MEAM), Department of Mechanical Engineering & Applied Mechanics, (2003).
4. Das, P.K., Behera, H.S., Panigrahi, B.K. A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning, *Swarm and Evolutionary Computation* Volume 28, Pages 14-28, (2016).
5. Abbas, NH, Abdulsaheb, JA, An adaptive multi-objective particle swarm optimization algorithm for multi-robot path planning, Journal of Engineering, (2016).
6. Alitappeh, R. J., Jeddisaravi, K. , Guimarães, F. G. Multi-objective multi-robot deployment in a dynamic environment, 21(21):6481–6497, (2017).
7. Vered, M., Kaminka, G. A. Heuristic Online Goal Recognition in Continuous Domains, Computer Science, Artificial Intelligence, (2017).
8. Nazarahari, M., Khanmirza, E., Doostie, S. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm", Expert Systems with Applications, Volume 115, Pages 106-120, (2019).
9. Fujimura, K., Path Planning with Multiple Objectives. Journal of IEEE Robotics and Automation Society, 13(1):33-38. (1996).
10. Min, H.Q., Zhu, J.H., Zheng, X.J. Obstacle avoidance with multi-objective optimization by PSO in dynamic environment. Proceedings of the IEEE International Conference on Machine Learning and Cybernetics,5, p.2950-2956, (2005).
11. Aurenhammer, F. Voronoi diagrams—a survey of a fundamental geometric data structure, ACM Computing Surveys (CSUR), 23(3):345-405, (1991).
12. Mahkovic, R. Slivnik, T. Generalized local Voronoi diagram of visible region, Proceedings. 1998 IEEE International Conference on Robotics and Automation., (1998).
13. Gang, L., Jianping, Y., Yangsheng, X. Multi-objective optimal trajectory planning of space robot using particle swarm optimization. Proceedings of the International Symposium on Neural Networks, p.171-179, (2008).
14. Kang, D.-O., Kim, S.-H., Lee, H., Bien, Z. Multi objective navigation of a guide mobile robot for the visually impaired based on intention inference of obstacles. Journal of Autonomous Robots, 10(2):213-230, (2001).