Journal of Theoretical and Applied Information Technology

www.jatit.org

INSTANTANEOUS PATH FIXING OF AUTONOMOUS MOBILE ROBOTS

T. Sudhakara Rao[1], E.G. Rajan[2], M.V. Ramanamurthy[3], P.Ram Kumar[4] ,S.Salahuddin[5].

[1]Departments of Mathematics, Computer Science and Engineering Vasavi College of Engineering, Ibrahimbagh, Hyderabad–500 031, A.P., India

[2]President, Pentagram Research Centre, #201 Venkat Homes, MIGH 59, Mehdipatnam, Hyderabad – 500 028, A.P., India

[3]Department of Mathematics Georgia Mason University, Rsk,UAE.

[4]Department of Computer Science and Engineering , Osmania University3, Hyderabad–500 007, A.P., India

[5]Faculty Of Mathematics,AL-JOUF University,ALJOUF – K.S.A

ABSTRACT

The main problem in the guidance of Autonomous Mobile Robots is obstacle avoidance in unknown environment. This paper presents a novel technique for path fixing for obstacle avoidance using a framework of cellular logic. The images of the unknown environment obtained from local perception devices are scanned by an array of neighbourhood windows and processed by the techniques of contouring, skeletonizing and centroid determination. High risk regions and low risk regions are considered for nearby and distant obstacles. A framework of cellular logic is used to determine the probability of collision. The various possibilities are classified into equally likely classes. The optimal path is the one that has the common probability of collisions. The mission direction is accordingly fixed instantaneously to meet the challenge of obstacle avoidance.

Keywords: *Autonomous Mobile Robots, Obstacle Avoidance, Path Fixing, Probability of Collision, Optimal Path, Obstacle Avoidance*

1. INTRODUCTION

Autonomous Mobile Robots (AMR) are used in different applications like working in space, factory, airport transport, electronic library and hospital communication environment etc. of current era. The navigation problem in a constraint environment has been treated in many ways, the difference usually lying in the knowledge base[1]. The robot motion is generally decided by taking into account obstacle configuration, robot kinematics and the result of the modification of the local perception. Subject to these basic constraints, several procedures and methodologies have been developed for fixing optimal paths of AMRs in known as well as unknown environment. Traditionally two basic approaches are being used for path generation of an AMR: (i) fixing of the total path in a known environment and (ii) instantaneous fixing of path direction depending on situations in an unknown environment. The former approach is error free because the nature of the environment is a prior known, where as the latter approach is not. Most of the current research is being focused on to the second approach with an intention of developing a universal technique in order to meet the challenges of obstacle avoidance in any unknown environment by an AMR, a spaceship for instance, in deep space with asteroids. We too are concerned here with the problem of identifying a technique for path generation of an AMR in an unknown environment. Over past few years, a number of path fixing techniques have been developed each having its own merits and demerits. Almost all the techniques involve the geometries, velocities, time segments and directions of motion of obstacles and of the AMR while fixing the path of the AMR instantaneously. The work carried out in this paper centres around a novel path fixing technique in the framework of Cellular Logic.

2. PROBLEM FORMULATION

An autonomous mobile robot is moving in E^3 (Euclidean) space in a particular direction called Mission Direction (MD). It has the following objectives.

1. To avoid collision by reorienting itself by instantaneously fixing its mission paths in a collisioin prone environment (Target Avoidance Problem, TAP).

2. To set its aspect straight to a target by instantaneously fixing the attack points in the moving target (Target Seeking Problem TSP).

2.1 Traditional approach:

Traditionally two basic approaches[4] are being used for path generation of an AMR, fixing of the total path in a known environment, and instantaneous fixing of path directions depending on situations in an unknown environment. The Target Avoidance Problem TAP is of concern to us, especially now when PSLVs are made and launched successfully in India. In the case of Autonomous Homing Weapons too, TAP plays and important role. The combat space in which such weapons and vehicles are placed could be viewed as the union of three classes of targets, viz.

(i) W, a class of wanted targets. (enemy)

(ii) $\mathfrak{R}$, a class of rejected targets (friend) and

(iii) U, a class of unknown targets.

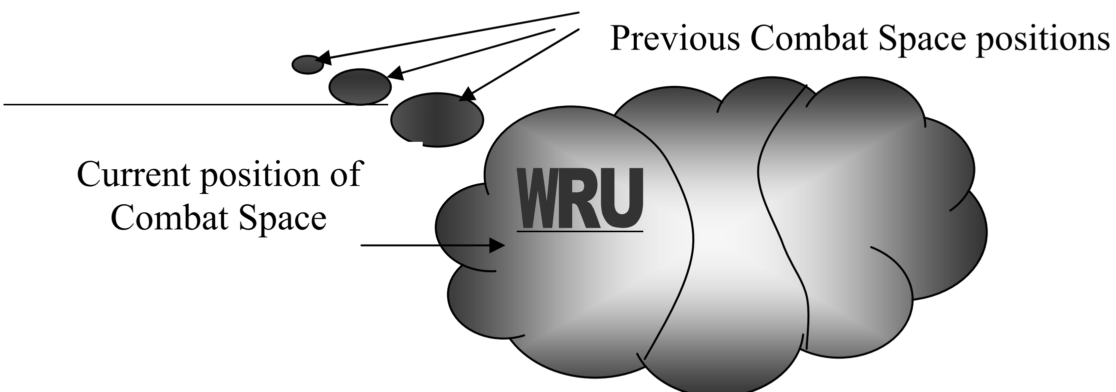

Figure 1: Combat space model and its classification

First, the system (weapon) should recognize a target and classify it as a member of any one of these three classes. Next it should use TAP related strategy to avoid the target if it belongs to $\mathfrak{R}$ or use Target Seeking Problem (TSP) related strategy to home on to the target if it belongs to W.

The former approach is error free because the nature of the environment is known a priori, whereas the latter approach is not. Most of the current research is being focused on to the second approach with an intention of developing a universal technique in order to meet the challenges of obstacle avoidance by an AMR in any unknown environment, e.g. a spaceship in the deep space with asteroids for instance.

2.2 Our approach:

Here we use a novel approach to instantaneous path fixing for an Autonomous Mobile Robot (AMR) sign certain image processing and pattern recognition tools and concepts from a paradigm called Cellular Logic Array Processing[11,12]. These tools are used (i) to find the boundaries of the digital image of an interfering object so that it can be avoided (TAP) and (ii) to find the skeleton of the image of the interfering object so that it could be sought after (TSP). Image acquisition is an important activity in such cases. In this regard, we use a technique called Collision Avoidance Using "Conic Projection Image Senior (COPIS)" for obtaining a composite digital image consisting of the scene in front of the robot and of the omni directional scene surrounding the robot[9]. From this composite image, control of the Mission Direction (MD) is carried out using constructive logical methods. This paper proposes certain TAP based logical formulas and a TSP based cellular logic algorithm using which the MD of a robot could be controlled.

3. MODELLING OF THREE DIMENSIONAL SPACE

The three dimensional Euclidean space E3 is discretized here as a 3-D grid of uniform cells, each cell taking the shape of a cube. The presence of an obstacle in a cell attributes a value 1 to that cell whereas the absence of an obstacle attributes a 0 to it. It is important to note that a cell in the 3-D grid is an abstract model and so it is not limited by morphology and size. Usually the perception space of an AMR is classified as a disjoint union of two subspaces (i) one consisting of recognized entries (ii) the other consisting of unrecognized entries. In our case, the perception space is treated as a disjoint union of three subspace (i) one consisting of objectively recognized entities, (ii) the other consisting of objectively unrecognized entities and (iii) the third consisting of subjectively recognized entities. It is clear from this classification that there is an in-built fuzziness in characterizing entities belonging to a perception space. It is to be noted here that a perception space of one AMR is not identical to that of another AMR. After identifying the presence of targets in the discrete perception space of the robot, it is essential to decide the strategy (TAP or TSP) to be adopted by the robot. If the TSP strategy is to be adopted by the robot, then its mission direction (MD) is fixed on to the selected attack point of the target. Thus, the MD of a target seeking system would always change with respect to a selected moving target. On the other hand, if the TAP strategy is to be adopted by the robot, then its mission direction is continued or changed depending on the number of targets and their positions in the perception space of the robot.

4. Instantaneous path fixing

The work carried out here centers around a novel path fixing technique in the framework of Cellular Logic Array Processing. Development of fast algorithms for the instantaneous fixing of collision free paths by AMRs in unknown environment had been the major output of this intended research.

AMR control is modeled as a constructive system that operates on two regions (i) low risk region and (ii) high risk region as given below. The term ***High Risk Region*** (HRR) refers to, for example, the scenic coverage around the AMR whereas the ***Low Risk Region*** (LRR) refers to the front scenic coverage acquired by the vision system of the AMR. In a real life situation also, one could see that the risk due to obstacles approaching from the sides is more than the obstacles approaching from the front, hence head-on collisions are rare on roads when compared to collision from sides. Figure 2 shows the HRR and LRR with respect to the AMR.

Techniques are developed for instantaneous fixing of mission paths for AMRs in collision prone environment (Target Avoidance Problem, TAP).

This position of an autonomous mobile[5] robot (AMR) in its discrete perception space would be more or less like the one shown in figure-2.

The AMR is assumed to be fitted with two digital cameras attached to each other back to back so that the forward looking camera, (i.e. camera #1) acquires image of the scene that is in front of the robot, and the rear camera (i.e camera #2) acquires the omni directional image reflected from the conical mirror which is kept in front of it. In other words, camera #1 covers low level threat region and camera #2 covers high level threat region show in figure-1. The composite image acquired by the robot is processed for the purpose of pattern recognition and decision making. The gray image is of size, say, 512 x 512. The digital image acquired by camera #1 is

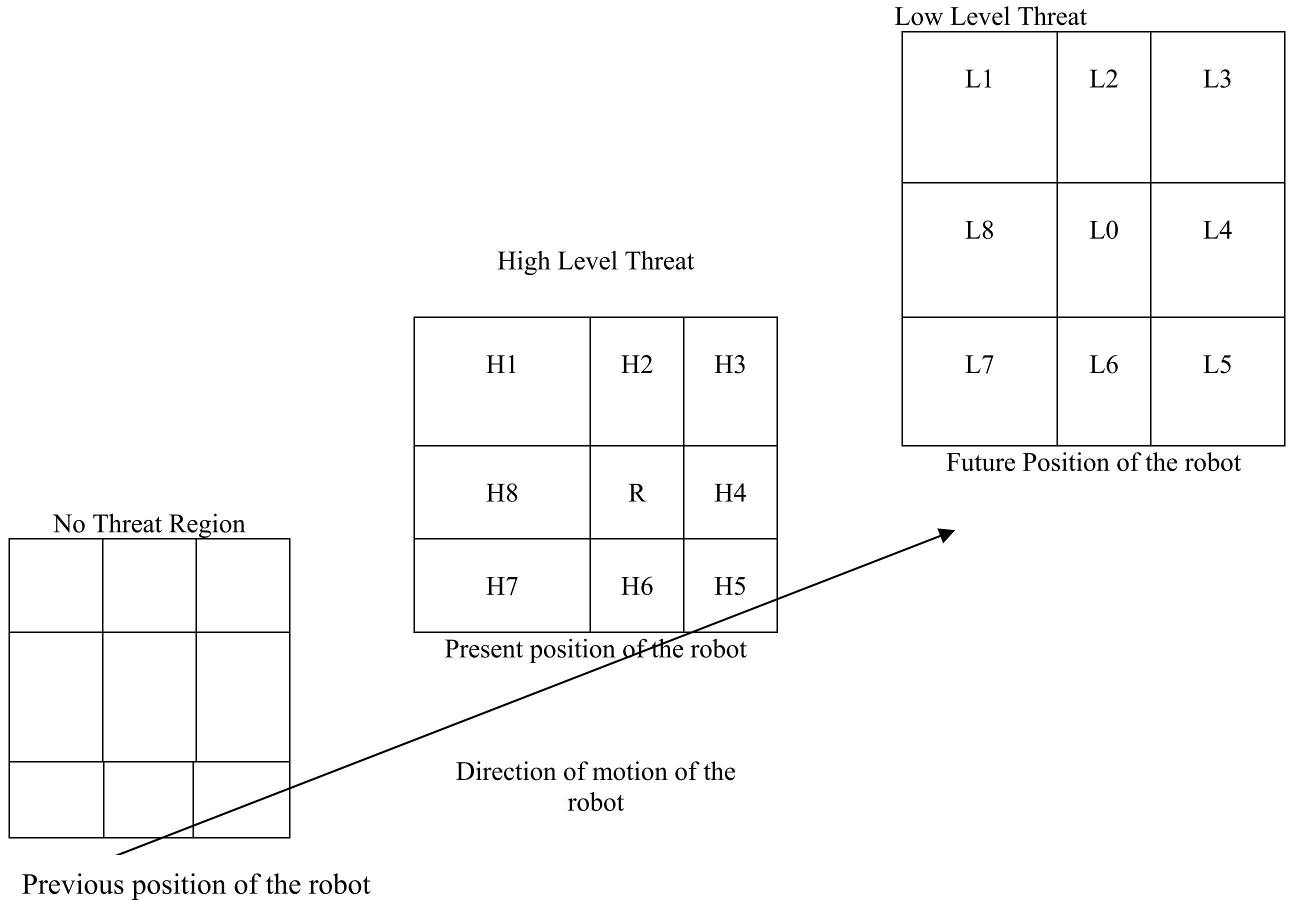

Figure-2: Position of AMR in its discrete perception Space

also of size 512 x 512 but it is reduced to the size of 128 x 128 in order to be superimposed exactly on the central blind region of the conical mirror. It is to be noted here that this composite image has two blind regions (i) central blind region and (ii) rear blind region with certain solid angle which depends on the angle of the conical mirror. This types of image acquisition system is recommended for a AMR.

4.1 Target seeking problem:

We are concerned herewith the problem of instantaneous path fixing of a vision based autonomous mobile system, say a missile sent on a mission especially in an unknown space. The formulation of a technique has been tried theoretically in the framework of cellular logic array processing. The central idea behind this technique is that a homing system decides its direction of movement based on the neighborhood in which it is situated at a particular instant of time. The direction parameter of an autonomous system for next instant of time is decided by a formula that involves the current position and intended direction of the system and the current position and direction of motion of the target. Moreover, the bull's eye point, otherwise called attack point, of the target image is obtained from the skeleton of the image. Now the Autonomous Mobile Robot has three processed images corresponding to a sampled image. From the contoured image S, the AMR estimates the size of the obstacle. From the skeletonized O, it understands the orientation of the obstacle. From the Centroid fixed image D it compares the position of the Centroid to its positions from the previous image frame and estimates the direction of the obstacle. The data set consisting of three images is represented <S, O, D>, the Autonomous Mobile Robot decides whether the obstacle size grows from frame to frame. It decides that the object is moving away from it if the size goes on decreasing and vice-versa.

A powerful novel technique for collision avoidance based on omni-directional viewing and capturing of images by a Conical Projection Image Sensor(COPIS) is used. A fast algorithm for obtaining the skeleton of a digital image is given below.

The given image is raster scanned by a five neighbourhood windows. On each move, the 3X3 subimage covered by this system of windows is examined to determine whether the difference D between the maximum and the minimum gray value, say D, is less than or equal to a threshold, say, T. if D is less than or equal to T, then the boundary is removed. Otherwise the window is moved to the next pixel. The overall effect would be the boundary removed version of the original image. This procedure is continued till there is no boundary left for removal. The resulting image would be the skeletonised version of the original image. From the skeletonised image, the attack point is chosen which in turn decides the direction of the homing system.

4.2 Target avoidance problem:

We are concerned here with the problem of instantaneous path fixing of a vision based autonomous mobile system, say a space probe, sent on a mission especially in an unknown space. The problem of concern here is to decide the direction of movement of the system which would avoid collision with identified obstacles. The direction parameter of an autonomous system for next instant of time is decided by a formula[6] that involves the current position. The intended direction of the system in a neighborhood consisting of obstacles, be they mobile or not, and their positions and estimated directions of notion.

The central idea on which the decision making is done is that the moving objects in space are recognized either as harmful or as harmless. Harmful objects are those which approach towards the AMR and harmless ones are those which move away from the robot. Less harmful objects are those which are captured by camera #1, whereas, those captured by camera #2 are more harmful. The loci of harmful objects are evaluated using cellular logic principles and Fast contact determination in dynamic environment[12,15]. The logic goes as follows:

The intended path of the robot is given R→L0. Normal speed of robot is denoted as S. A. path is fixed by the robot depending on the positions and speeds of obstacles. At a particular instant the robot dynamics is determined by the ordered pair <S, R→Li>, $1 \le I \le 8$. There are 8 loci corresponding to less harmful objects approaching the robot from 8 directions: Lj →R, $1 \le j \le 8$. Now the cellular logic scheme that fixes the instantaneous path is < S, R → Li > = θ (R→LJ, $1 \le j \le 8$, Hk → R, $1 \le k \le 8$).

5 A theoretical case study:

Let us assume that R→L0 is the intended direction of an AMR's mission. The first collision possibility is due to a single obstacle approaching the AMR from the direction L0 →R. This low threat position is pictorially represented in figure-3. The presence of a 1 in a cell indicates the presence of an obstacle.

Now the AMR has to change its direction of motion to any one of the following ones: R → L1, R → L2, R → L3, R → L4, R → L5, R → L6, R → L7, R → L8 with the probability of 1/8. The AMR can adjust its speed

0	0
1	0
0	0

Figure-3: Low threat position indicating head-in collision.

depending on its relative velocity with respect to that of the obstacle's. Let us assume that the AMR takes on the direction R → L1 with a deviation of D^0 and continues in that direction for t1 time units with the same speed is or with a different speed, says'. Then it necessarily changes its direction to R → L0 with the deviation of $(2D)^0$ exactly for t1 time units and move with the same speed. After 1 time units, the AMR changes its direction once again with the deviation of D^0. This strategy ensures regain of original direction of its mission.

For brevity, we call this strategy as Restorative Deviation.

Next, let us assume that two low threat obstacles approach the AMR from the directions L0 → R and L1 → R. Now the AMR can change its direction of motion to any one of the following ones: R → L2, R → L3 R → L4, R → L5, R → L6, R → L7, R → L8. One can construct a total of eight two-obstacles threat positions that have an obstacle at L0, as shown below in figure-4.

1 0 0	0 1 0	0 0 1	0 0 0
0 1 0	0 1 0	0 1 0	0 1 1
0 0 0	0 0 0	0 0 0	0 0 0
0 0 0	0 0 0	0 0 0	0 0 0
0 1 0	0 1 0	0 1 0	1 1 0
0 0 1	0 1 0	1 0 0	0 0 0

Figure-4: Eight 2-obstacles threat positions.

With one obstacle at the L0 position, one can verify that there are 256 level but high risk threat positions. Similarly, one can construct 256 low level low risk threat positions with out an obstacle in position L0. By low risk threat, we mean that the AMR can continue in its intended direction R→ L0 with low risk in spite of the fact there are obstacles present elsewhere in the neighbourhood. Figure-5 shows a two-obstacles low risk threat position.

From the above argument we see that an AMR faces a total of 512 low threats with 256 high risk zones and 256 low risk ones. In addition to the above an AMR faces 256 high threat zones from its sides. The AMR changes its speed and initial position depending on the high threat level. Hence, with the

1	0	0
0	0	1
0	0	0

Figure-5: *2-obstacles low risk threat positions*

three dimensional space being modeled as a discrete grid, an AMR faces a total of 768 threat zones while undertaking a mission in the discrete space, out of which one is totally threat free. The restorative deviation method is adopted by the AMR to avoid different types of collusions. Further precise modifications could be made if we resort to probabilistic methods in evaluating the relative velocities and assigning suitable priority values to different directions while fixing the path.

With the basic model for a collision avoidance system[5-10], our efforts would now be focused on to the possibility of using logical control of the AMR in the framework of Constructive Mathematical framework. The constructive logical formulas that govern the motion of an AMR are given below:

CASE #1

<u>The state vector control is carried out by control formulas. For convenience, we denote the MD (Mission Direction) of an AMR by $R \to L_0$, where R denotes the current position of the robot and L_0 the next position of the robot in LRR. Now, we shall assume that there is no obstacle in the L_0 position. We shall also assume that there is no obstacle in the high risk region. Then the following formulas are valid. MD (Mission direction) of an AMR is assumed to be $R \to L_0$.</u>

<u>There are obstacles in the high threat region.</u>

<u>Then the following eight control formulas are valid</u>

1. $$(L_k = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k} RL_i$$

(one obstacle in Eight possible regions)

2. $$(L_{k1\,\&}\, L_{k2} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2} RLi$$

3. $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3} RLi$$

(Three obstacles in Eight possible regions)

4. $$(L_{k1} \,\&\, L_{k1} \,\&\, L_{k1} \,\&\, L_{k1} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3,\, K_4} RLi$$

5. $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} \,\&\, L_{k4} \,\&\, L_{k5} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3,\, K_4,\, k_5} RLi$$

6. $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} \,\&\, L_{k4} \,\&\, L_{k5} \,\&\, L_{k6} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3,\, K_4,\, k_5,\, k_6} RLi$$

7. $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} \,\&\, L_{k4} \,\&\, L_{k5} \,\&\, L_{k6} \,\&\, L_{k6} \,\&\, L_{k7} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3,\, K_4,\, k_5,\, k_6,} Rli$$

(Seven obstacles in Eight possible regions)

8. $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} \,\&\, L_{k4} \,\&\, L_{k5} \,\&\, L_{k6} \,\&\, L_{k6} \,\&\, L_{k7} \,\&\, L_{k8} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i \neq k_1,\, k_2,\, K_3,\, K_4,\, k_5,\, k_6,\, k_7} Rli$$

9. . $$(L_{k1} \,\&\, L_{k2} \,\&\, L_{k3} \,\&\, L_{k4} \,\&\, L_{k5} \,\&\, L_{k6} \,\&\, L_{k6} \,\&\, L_{k7} \,\&\, L_{k8} \,\&\, L_{k9} = 1) \Rightarrow \bigvee_{i=0}^{8, i=k1,\, k2,\, k3,\, k4,\, k5,\, k6,\, k7,\, k8,\, k9} Rli$$

(Nine obstacles in Eight possible regions)

Now, refer to the figure 6 Direct neighbors to the future position (LRR) of the robot L_0 are L_2, L_4, L_6, L_8. The indirect neighbors are L_1, L_3, L_5 and L_7.

L_1	L_2	L_3
L_8	L_0	L_4
L_7	L_6	L_5

Figure 6: Low risk region with direct and indirect neighbors

Now, the first control formula permits eight LRR possibilities, which are shown in figure 7.

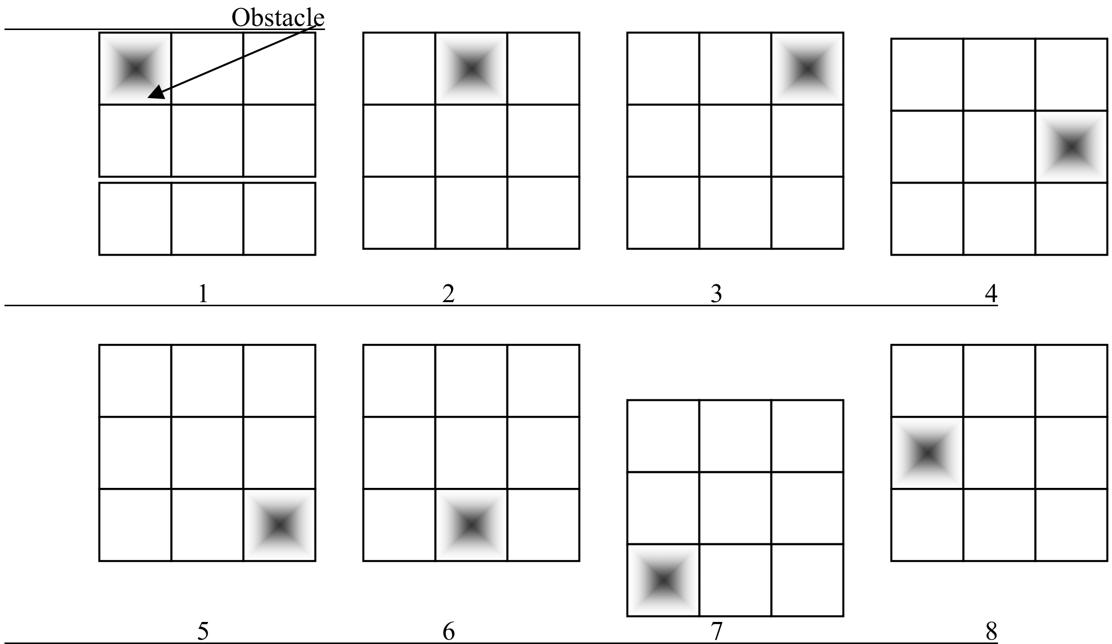

Figure 7: Eight possibilities of single obstacle present in LRR

We assume that the AMR is moving towards L_0. Now, if there is an object in a direct neighborhood, then its distance from L_0 is assumed to be 1 unit. On the other hand, if an obstacle is found in an indirect neighborhood, then its distance from L_0 is assumed to be $2^{1/2}$ units. If one obstacle is found in a direct neighborhood and one obstacle in an indirect neighborhood, then the total distance from these two obstacles to L_0 would be calculated as $1+2^{1/2}$ units. There are 9 cells in the discrete region as defined in figure 6. The scalar metric (total distance) from all obstacles to any empty cell is denoted as $D(L_i)$ where $D(L_i) = \sum d(L_i, L_j) \geq 1$ where L_i is the position of the empty cell and L_j is a position containing an obstacle. For example consider an empty cell at L_0. Assume that obstacles are in L_1, L_3, L_5. Then $D(L_0) = d(L_0, L_1) + d(L_0, L_3) + d(L_0, L_5) = 2^{1/2}+2^{1/2}+2^{1/2} = 3(2^{1/2})$. Now the probability of collision due to an obstacle at L_1 is denoted as $P\{L_1\}$ or in short, as $P\{1\}$. With these basic definitions and assumptions, the algorithm proposed fixes the change in the Machine Direction (MD) of the robot.

CASE #2

The MD(Mission Direction) of an AMR is R -> L0. It is assumed that there is no obstacle in the high level threat region.

Then the following formulas are valid.

1. $(H_k = 1) \Rightarrow \bigvee_{i=0}^{8,\, i\neq k} RHi$

(one obstacle in Eight possible regions)

2. $(H_{k1}\ \&\ H_{k2} = 1) \Rightarrow \bigvee_{i=0}^{8,\, i\neq k_1, k_2} RHi$

3. $(H_{k1}\ \&\ H_{k2}\ \&\ H_{k3} = 1) \Rightarrow \bigvee^{8,\, i\neq k_1, k_2, K_3} RHi$

i=0

$$4.\quad (H_{k1} \,\&\, H_{k1} \,\&\, H_{k1} \,\&\, H_{k1} = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k_1,\ k_2,\ K_3,\ K_4} RHi$$

(Four obstacles in Eight regions)

$$5.\quad (H_{k1} \,\&\, H_{k2} \,\&\, H_{k3} \,\&\, H_{k4} \,\&\, H_{k5} = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k_1,\ k_2,\ K_3,\ K_4,\ k_5} RHi$$

$$6.\quad (H_{k1} \,\&\, H_{k2} \,\&\, H_{k3} \,\&\, H_{k4} \,\&\, H_{k5} \,\&\, H_{k6} = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k_1,\ k_2,\ K_3,\ K_4,\ k_5,\ k_6} RHi$$

(Six obstacles in Eight regions)

$$7.\quad (H_{k1} \,\&\, H_{k2} \,\&\, H_{k3} \,\&\, H_{k4} \,\&\, H_{k5} \,\&\, H_{k6} \,\&\, H_{k6} \,\&\, H_{k7} = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k_1,\ k_2,\ K_3,\ K_4,\ k_5,\ k_6,} RHi$$

$$8.\quad (H_{k1} \,\&\, H_{k2} \,\&\, H_{k3} \,\&\, H_{k4} \,\&\, H_{k5} \,\&\, H_{k6} \,\&\, H_{k6} \,\&\, H_{k7} \,\&\, H_{k8} = 1) \Rightarrow \bigvee_{i=0}^{8,\ i\neq k_1,\ k_2,\ K_3,\ K_4,\ k_5,\ k_6,\ k_7} RHi$$

$$9.\quad (H_{k1} \,\&\, H_{k2} \,\&\, H_{k3} \,\&\, H_{k4} \,\&\, H_{k5} \,\&\, H_{k6} \,\&\, H_{k6} \,\&\, H_{k7} \,\&\, H_{k8} \,\&\, H_{k9} = 1) \Rightarrow \bigvee_{i=0}^{8,i=k1,\ k2,\ k3,\ k4,\ k5,\ k6,\ k7,\ k8,\ k9} RHi$$

(Nine obstacles in Eight possible regions)

These quantifier free constructive logical formulas form the control signals that change the state vectors of the AMR moving in space. These Я$_{\omega|}$ -provable (∃, ∀)-free constructive logical formulas form the control signals that change the state vectors of the AMR moving in space. The results and further discussions are given in the next section.

5.1 Equally likely strategy

For convenience, let us assume that there is no obstacle in L0 position as well as in the high level threat regions. Then the first eight formulas given in Case #1 are valid. Now let us refer to figure 4 L2, L4, L6 and L8 are the direct neighbors to L0 whereas its indirect neighbors are L1, L3, L5 and L7. Further, we assure that the robot is moving towards L0. Now, if there is an obstacle in a direct neighborhood, then its distance L0 is assumed to be 1 unit. On the other hand, if an obstacle is found in an indirect neighborhood, then its distance from L0 is assumed to be 2 units. If one obstacle is found in a direct neighborhood and one obstacle in an indirect neighborhood, then the total distance form these two obstacles to L0 would be calculated as 3 units[11]. There are 9 cells in the discrete region as defined in figure-4. The scalar metric from all obstacles at Lj (j >- 1) to an empty cell Li is denoted as D(Li), where D(Li) = d(Li,Lj); j>-1; here Li is the position of the empty cell and Lj is the position containing an obstacle.

For example, consider an empty cell at L0. Assume that obstacles are in L1, L3 and L5. Then D(L0) = d(L0, L1) + d(L0, L5) = 6. Now, the probability of collision due to obstacle on L1 is denoted as P(1). Based on these basic assumptions, the following algorithm ccould be used to fix the MD of the robot.

L1	L2	L3
L8	L0	L4
L7	L6	L5

Figure-4: Discrete region model

6. Algorithm

Given the number of obstacles and their respective positions., the scalar metric of each empty cell is calculated[2] including say, L0. The position Li with the largest D(Li) value is taken as the safest direction to avoid collision. There are eight control formulas and so, there are 256 possibilities the robot has to investigate before it proceeds further.

When an obstacle is found in L1, the probability of collision at position L0 is directly proportional to the D value of L0 D(L0) = d(L0, L1) = 2. Let us denote the probability corresponding to this metric as P(1). P(1) is the probability of collision due to obstacle in L1. Note that the probabilities P(1), P(3), P(5) and P(7) are equally likely. Similarly, the probabilitiies P(2), P(4) and P(6) are equally likely. In the same manner, one can verify that the probabilities P(2,4), P(2,6), P(2,8), P(4,6), P(4,8) are equally likely.

The probability of collision is 0 when there is no obstacle and it will be a certainity that no obstacles are found in all positions from L0 to L8. Consequently, the remaining 254 collision possibilities form the real time collision processes. All these 254 possibilities are categorized into 23 equally likely classes. When a collision problem falls in a particular equally likely class, then any of the equally likely positions can be chosen for forward motion of the robot[4]. In real world situation, the robot can meet with an obstacle in the L0 position too. In such a case, the collision possibilities would rise to 512 instead of 256.
Assume that the robot has risk from the high level threat region also. Then, the eight control formulas as stated in case #2[13] are valid, thus giving rise to 256 collision possibilities which are categorized under 23 equally likely classes.

Note that the risk probabilities have been computed[3,14-15] till now by treating low level and high level threat regions independently. In practice, this would prove to be erroneous.

In fact, the complete solution to the collision avoidance problem is that there are 512 x 256, that is 1,31,072 possibilities of collision that an AMR has to encounter[17] if the dynamics is modeled in the discrete space and there are 529 collision avoidance strategies each with equally likely probabilities.

7. Conclusions

A novel technique with a fast algorithm for path fixing of Autonomous Mobile Robot for obstacle avoidance using a framework of cellular logic is presented. The images of the unknown environment obtained from local perception devices are scanned by an array of neighborhood windows and processed by the techniques of contouring, skeletonizing and centroid determination. High risk regions and Low risk regions are considered for nearby and distant obstacles and a framework of cellular logic is used to determine the probability of collision. The mission direction is accordingly fixed instantaneously to meet the challenges of obstacle avoidance. The fast algorithms for the instantaneous fixing of collision free paths by AMRs in unknown environment had been the major output of this intended research.

8. Acknowledgements

This paper is an abridged version of a RESPOND proposal titled Modelling of Collision Processing submitted to Vikram Sarabhai Space Center, Thiruvananthapuram. Most of the research work has been carried out at Pentagram Research Center Private Limited, Hyderabad and at the Advanced Center for Automation Research. Vasavi College of Engineering The authors thank the Managing Director, Pentagram Research center, Hyderabad for permitting them to present and publish a part of the research.

REFERENCES

[1]. Stanislay Vassilyev, Logical Approach to Intelligent Control, Proceedings of IEEE SMC meeting on CESA '98, Tunisia, April 1-4, 1998.
[2]. Beeson M.J, Foundations of Constructive Mathematics,Springer Verlag,Heidenberg,1985.
[3]. Markov, A. A., Normal Algorithms which Compute Boolean Functions, Soviet Math. Dokl., 5, 1964.
[4]. Grzegorczyk, A., An Outline of Mathematical Logic, D. Reidel Pub. Company, 1974.
[5]. Rajan, E.G. Modelling of Collision Processes, Technical Report PRC/ PRG/TR-02/98, Pantagram Research Center Private Limited, Flat No. 201, Venkat Homes, MIGH 59, Mehdipatnam, Hyderabad 500 028, Andhra Pradesh, India.
[6]. Rajan, E.G., Symbolic Computing - Signal and Image Processing, B.S.Publications, Hyderabad, 2003
[7]. Rajan, E.G., and Ramprasad, V.V., Pattern-Directed Array Processing, Technical Report, TR/IP-5-91. Department of Electrical Engineering,

Indian Institute of Technology, Kanpur, January, 1991.

[8]. Rajan, E.G.,High-Throughput Cellular Logic Array Processing of Remote Sensed Imageries, International Conference on Remote Sensing and GIS, ICORG-92, J.N.T. University,Hyderabad, Proceedings, Tata McGraw Hill Publishing Company Pvt. Ltd., pp-347-440.

[9]. Rajan E.G., High-Throughput Cellular Logic Array Processing of Satellite Data for Geophysical Surveying, Paper No A.1-S.1.08. The World Space Congress, Washington, D.C., U.S.A., 28 August- 5 September 1992.

[10]. Rajan, E.G., Object Recognition and Tracking in a Multiple Target Environment, 1994 IEEE International Conference on Systems, Man and Cybernitics, San Antonio, Texas, U.S.A., October 2-5, 1994, pp-2390-2395.

[11]. Wolfrom, Stephen, Theory and Applications of Cellular Automata, World Scientific Publishing Company, 1986.

[12]. Rajan, E. G., Cellular Logic Array Processing, Techniques forhigh throughput Image Processing Systems, SADHANA, Special issue on Computer Vision, Volume 18, Part 2, June 1993 pp.279-300, Indian Academy of Sciences.

[13]. Marko, A.A, Theory of Algorithms The Israel Program for Scientific Translation, Jerusalem 1961.

[14]. Antonio Benitez, Maria del Carmen Ramirez, Daniel Vallejo, Collision Detection using Sphere-tree Construction International Conference on Electronic Communications and Computers, IEEE 2005.

[15]. Lin,M.C., Mancoha, D.., Carry J.F. Fast contact determination in dynamic environments. In International Conference on Robotics and Automation,. Pp.602-608. IEEE 1994.

[16]. Pin, J.E., Varieties of Formal Languages, Plenum Press, New York, 1986.

[17]. F.Policarpo, A . Conct. Real-Time Collision Detection and Response. Proceedings of the Brazilian symposium of Computer Graphics and Image Processing(SIBGRAPT'01), IEEE, 2001.