International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Email:editorijoer@gmail.com http://www.ijoer.in

Vol.4., Issue.2., 2016
(Mar-Apr)

RESEARCH ARTICLE

ISSN: 2321-7758

ROBOTIC FAILURE COORDINATION IN MULTI-ROBOT HUNTING

GEBREHIWOT TEKLEHAYMANOT ABRAHA, POORVA AGRAWAL

Department of Computer Science Engineering, Symbiosis Institute of Technology, Mahrashtera, Pune, India

ABSTRACT

Multi robots are cooperated to take out simple and complex tasks of human activities ranging from simple to very huge and complex tasks. Inside a multi robot the tasks such as; coordination, path planning, collision avoidance, searching has been among the areas of interest. Beside this, robots also should have to take extensive action towards the robot breaks down. In multi robot sometimes robots fail to achieve their chores because of many interior and external causes. This may lead to task interruptions in the robots overall goal achievement. The primary focus of this work is on coordination of multi-robots when one of the member robots failed to catch an evader in unknown environment. Due to different physical, human and dependability factors robots may fail before reaching their ultimate goal. Therefore, robots should have to be guaranteed, in order to perform their action without any interruption, since the interruption of the system task will result in high loose of resource and time. Due to this reason, the contribution of this paper is how to guarantee functionality and performance of the whole system as any member robot is failed from the alliance. The problem in large highly coupled system is at the time of failure, as any failure is occurred the whole system failed. To overcome this problem the systems should designed by maintaining the system cooperation and decreasing the failure dependability of the system. In this paper we propose two approaches on how to guarantee the system at the time of failure by adding robustness in multi-robot to accomplish the hunting task using strategy of down continuity and replacement strategy. Finally, the proposed approach aims to guarantee the hunting task to accomplish with the available resource and time.

Key Words—Robotic failure, failure Independency, Idle robot, Robotic replacing, Hunting

1. INTRODUCTION

The advancement in the robot technology came up with the making ease of human undertakings. Difficult conditions are able to practice with robots ranging from small to large and complex, such as bomb exploration[1], field/space robots[2-3], medical tasks [4-7], cooperative hunting[8,9-15]and many more. Some of the advanced ways of using of robot is the use of multi robots. As different studies show the use of multi robots[16-20]makes ease of the complex activities, such as hunting, space exploration and other cooperative tasks. To do such kind of tasks robot should be guaranteed with high performance and backup of robot reserve. The backup robots are robots in idle state in which they are assigned to perform the task if working robots are failing to function. Today, robots are able to participate in the hunting[8, 21-26]of the invaders in an unknown environment [8, 27]wherein the robots perform efficiently and complete their tasks successfully.

In the last decade, much research has dealt with multi robot, coordination, communication, path planning and failure of robots[8, 21-27]. Robots can coordinate, interact and communicate with each other to perform specific common goal. The coordination of multi robots helps them to execute an action in time with high assurance.

The availability of the robots in the alliance should be assured and have to work without failure. As the research of Carlson J*et al*[28-29]shows the Mean-time-between-failure (MTBF) for robots of many manufacturers is 6 to 20 hours. They tried to improve ratio to 24 hours and the availability is 54%. Based on this they suggest that the probability of MTBF it is still too low. Finally, they conclude by saying that, the gap between the mean times to failure and mean time to repair, downtime varies widely. This study shows that since failure is obvious and the repair time takes time in conditions that are difficult to repair the robot. The alternative solution is to assign idle robots for backup or to perform the task with $N-i$ robots where; N is Number of robots and i is the failed robot.

Robotic failure may occur due to different reasons such as; physical failure, human failure, and dependability computing practices [28]. Physical failures may occur due to the problem with the sensor, effectors, power, control system, and communications. Human failures are failures raised due to the problem in design and Human Computer Interaction (HCI). Dependability computing practices are failures occurred due to the complete or partial association of robots to perform some task. Although dependability is quite obvious in mobile robots, the solo robotic failure and operation is also a big concern. Therefore, by taking this issue into consideration; our research is able to maintain the cooperation so as to manage the task by remaining robots which in turn decreases the dependability on the failed robots.

An extensive research has been performed with robotic hunting by Ni J, *et al* [8].The authors have considered the case where the robots break down before and after the evader is caught. They resolved the problem by reforming and restarting the hunting task. Restarting of the system is easy in the simulation project, but to be implemented in the actual universe, it is the wastage of resource and is too costly.

Another research of Khan T., *et al.*[30]overcomes robotic failure of failed robots by fixing of welding spots at each station .If any robot is failed the robot is repaired at that spot and failed robot is replaced by the working robot at that station .Research paper by Twala B.,*etal*[31] discusses on the prediction of execution failure of robots using incomplete data input to the robot.

The main contributions of this paper are summarized as follows. 1) On the robotic failure solutions: - assuring the reliability of the system, by granting the system with robotic backup. 2) Catching of the evader with $N-i$ robots where N the total number of robots in the failed and$i$ the number of failed robots. 3) Using the biologically inspired neural network to complete the hunting task.

The presentation is organized as follows. In Section 2, the problem statement is given. Section 3, presents the proposed approach to solve the multi robot robotic failure, Section 4 Result and discussion finally, the conclusion is given in Section 5.

2. Problem Statement

There is no sufficient research on the robotic failure adaptive cooperative hunting even the advancement of the robotic technology is increasing at an instant. In robotics, a lot of studies have been done on the cooperation and better communication of multi robots hunting and few on reliability and robotic failure [1, 26-27, 30-31]. However, the focus of these researches was not on the fault of member robots in multi robot cooperative hunting. Some of the researches undermine the fault of the robots, and others use different mechanisms to solve the problem of robotic failure to the whole system. Multi robots accomplish specific goal orientated tasks simply than single robots in terms of speed and confidence to accomplish the task. However; robotic failure coordination was being a problem in multi robot(s) cooperated hunting. Although the procession of robotic technology, dynamic from time to time the mechanism for guaranteeing of robots with high confidence at the time of failure is not notwithstanding achieved.

Our research mainly focuses on this issue that how hunting robots accomplish their action despite the failure of individual robot. The working robots should able to overcome the failure happened in the group. We propose two solutions to solving this issue, i.e. performing the hunting task with $N-i$ robots where $N$the number of robots is and $i$is the failed robot. For $R_1 = \{i_1, i_{2,}i_3 \; \ldots.. i_n\}$ working robots in searching for evader $E = \{e_1, e_{2,}e_3 \; \ldots.. e_n\}$ and $R_2 = \{j_1, j_2 j_3 \; \ldots.. j_n\}$ idle robot(s) for backup if any robot from R_1 is failed to function.

where R_1&R_2 are the working and Idle robots ,E is the evader ,and the R_1 (i) ,R_2 (j)&E (e) are the members of the group R_1,R_2,E respectively. When any robot breaks down from R_1so the temporary commander of the group will broad cast help to the idle robots R_2 and the R_2 (j) joins the group.

The number of idle robots to be used as backup depends upon the importance of the goal. If the goal we wish to realizes best than the worth and alternative technical issue of the robot(s) then supported this we have a tendency to decide range of idle robots. If we tend to unravel this drawback; the job of multi robot hunting is starting to be rigid; if any robot fails the whole system functioning to be discontinued. At this time the only choice is restarting the system or within the field then attempting to reform the alliance again [8]. This results in extra resource and time wastage. Here is the summary of the failure adapting hunting task. All the robots and evader has 360^0 visual capability. The evader is different only in speed and communication capability [1]. The robots have nine sensors each sensor has 40^0 visual capability. The robots are tray to search for the evader the first robot which find the evader will be the temporary commander (TMC); and broadcast a message to all remainder robots. The message to be broadcasted is the location of the evader and the id to identify from its other evader. The robots replay the status flag, and the location. If any robot is failed during the searching or pursuing then the two actions are performed; to continue the hunting task with $N-i$ or to ask for help from the backup for idle robots. Finally, if all the robots in catch state and if there is no way for the evader to escape then the evader is caught. The simple explanation of the summery is shown in Fig. 1.

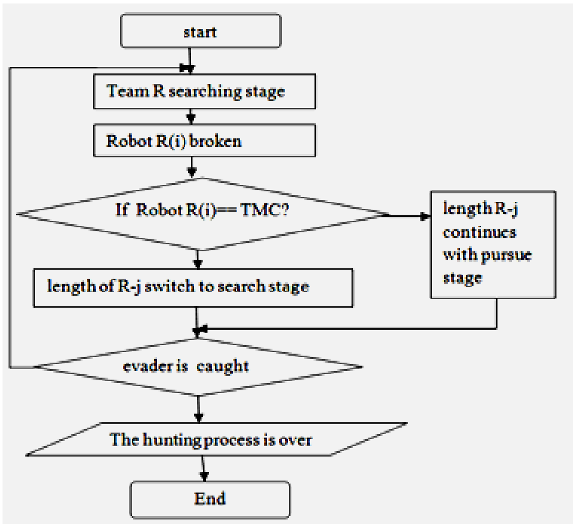

Fig.1. Flow diagram of the robotic hunting with failure

Proposed Approach

Robots are not only cooperating to accomplish a specific task, but also robots should cooperate in case of failure. The paper mainly focuses on how the robots are able to coordinate to function in hazardous situations. Due to this we propose two approaches. The foremost one is how to accomplish hunting task without interruption even the robots are less than the mandatory number of working robots. The second one is how to replace faulty robots.

Our research is based on the concept of bio inspired neural networks. The research on bio inspired neural network enables robots to have specialized path planning without the need for prior knowledge, cost function estimation [8, 21-22].We use three novel algorithms of Ni J, *et al*[8] Strategy of dynamic alliance, Pursuing Strategy Based on a Biologically Inspired Neural Network and Strategy of formation Strategy of dynamic alliance to form an alliance.

3.1 Performing hunting task with N robots

Strategy of dynamic alliance:-If any of the robots in the field finds the evader this robot will be the temporary commander r_{temp} ; then all the robots send the status flag (the state of the robot whether searching, failed, pursuing or catching) and the coordinates to r_{temp} .Based on this r_{temp} will calculate the distance from each r_i to e_j .

$$\text{dist}\left(p_{ri}, p_{ej}\right) = \sqrt{\left(x_{ri} - x_{ej}\right)^2 + \left(y_{ri} - y_{ej}\right)^2} \quad \ldots.(1)$$

Where$p_{ri,}$ and p_{ej} are position of robot and position of evader respectively. $dist(p_{rj}, p_{ej})$ represents function of distance from point p_{rj} to p_{ej} where point x and y represents the geometrical potions of each of the evader and the robot.

Pursuing Strategy based on a Biologically Inspired Neural Network; after the alliance is formed the next step is to follow the evader; all the robots follow the evader based on the special type of path planning. This type of path planning is based on the shunting equation [8, 21-22] in which all the communication is based on the neural landscape. The failed robot broadcasts its message to the nearest possible neuron around.

$$\frac{dx_i}{dt} = -Ax_i + (B - x_i)\left(([I_i^e]^+ + \sum_{i=1}^{k} W_{ij}[x_j]^+) - (D + x_i)[I_i^o]^-\right) \ldots(2)$$

The shunting equation already discussed [8, 21-22]but the shunting equation in this paper is working in beside the point discussed it focuses on the failed robots.

Variables $I_i^e$and I_i^o are the external input to the i^{th} neuron from the evader and obstacle, respectively. Strategy of formation; after the parsing strategy if all the robots are at the stage they can see the evader; the robots are arranged in a way the evader can't escape [8, 21-22].

Robots are able to communicate and coordinate with each other the robotic failure of the allies may result in interruption of the hunting task. There are two conditions by which failure may occur. 1) Failure of robot before having information about the evader 2) Failure occurred after Pursuing is occurring. The first condition occurs before the robots find the evader any robot may fail or break down on the way. At this time if the robot is failed to work or breakdown one of the robots from the idle robots will take the responsibility and move to the region and join the group to start hunting. The second condition occurs at the stage when the robots pursuing the Evader. Braking down of robot at this stage is difficult and there is probability to escape for the evader. Due to this we propose two scenarios. The first scenario is how to perform a task with remainder robots if any robot is failed from the allies; the second one is what if the restrobotsaren 'table to accomplish the task inside the desired time and obtainable resource. In this issue the idle robot(s) are released and join the group to replace the failed robot. These two approaches are discussed below.

3.2 Strategy of Down Continuity

In the real world, it is feasible to catch an evader with $N - i$ robots. There is no hard rule how and when to catch the evader .We can catch the evader in a way the evader can't escape. As an analogy to the real world when any robot is failed it is feasible to catch the evader with the remainder robots. This scenario enables the robotic hunting to perform the required task without need of additional resources. In this case there are three cases where robots $N{-}i$ can catch the evader:

Firstly; if the robots are at catch stage; means within some distance the robots can sense the evader or within the maximum range of the evader can't escape. The maximum range

(Max_{range}) is the maximum sensor capability at which the robot can sense its surroundings.

$$\forall robots if\ (Max_{range} > dist from equ(1) \ldots\ldots\ldots\ldots(3)$$

Second; if the evader is rounded by the $N - i$ robots and the obstacles and nowhere to escape.

$$E^{o}_{dist}(p_{ei}, p_{oj}) = \sqrt{(x_{ei} - x_{oj})^2 + (y_{ei} - y_{oj})^2} \ldots\ldots\ldots(4)$$

$$\forall robots if (equ(3) \&\& E^{o}_{dist} < min_{dist}) \ldots\ldots\ldots\ldots(5)$$

Where; E^{o}_{dist} is the distance from the evader to the obstacle and $x, y, e and o$ are the positions, evader and obstacles. min_{dist} is the minimum threshold distance from the obstacle to the evader where the evader can move without collision. Third; if the evader revolves within the same position for some elapsed time and at least one of the working robots can see the evader.

Evader

$$caught = \begin{cases} if \Delta Position_{evader} \leq 0, and \\ for any workig\ robot \\ if dist(p_{ri}, p_{ej}) < min_{dist}\ and \\ if elapsed_{time} > time_{max} \\ to\ revoleve\ on\ the\ same\ postion \end{cases} \ldots\ldots(6)$$

This may be caused due to the evader failure to escape from the hunter robots. The evader may fail due to different physical factors [26]such as; system power, sensor, effectors failures and so on. Therefore; if any robot is able to see the evader in such condition the robot in the catching stage broadcasts the message to the robots in pursuing stage and the remainder robots know that the evader is caught and hunting task is over. This serves to avoid unnecessary movements by the robots.

Algorithm for hunting with N-i robots

```
for i=1:length(robots)
    if(robots(i)) ==state_Break )
        then
        length(robots)= length(robots-j)
      for i=1:length(robots)
        Use formation algorithm to cover[1] the place of failed robot
          If (De<max_Range
             then
robot is in catch state.
                Hunting task is completed
                end if
          if(De<max_Range and Δposition of evader≈0   where elapsedtime= timeMax
               then
               evader is caught
             end if
          end for
       end if
    end for
```

where:
#j=failed robot
#D_e distance from robot to evader
#$time_{max}$ =max threshold time
#Max_{range} =sensor capablity of its envomment

3.3 . *Replacement Strategy*

This scenario applies when the robots in scenario one are unable to catch the evader within the maximum threshold time and when the temporary commander broadcast message to the idle robots. In addition to this taking the importance of the goal in to consideration is an important factor before idle robot replacement. This helps to avoid the coast and time over head of insignificant hunting goal (cost profit analysis is done manually), i.e. the value added from the hunting task is compared with the total cost of resource (technical and material).

As the working robots are set to search the evader; the idle robots (IR_1, IR_2, IR_3, IR_4) are positioned at the boundary as we see from Fig.3 $P_1(x,y)$,$P_2(x,y)$; where $f(x,y)$ is the position of the evader after failure is happened. If working robot $WR\ (i)$ failed at position $z(x,y)$ distance measured from $P1$,or $P2$ to f ;then nearest location is selected fromP_1,P_2 as follows:-

$$\mathrm{dist}(p_1, \mathrm{f}) = \sqrt{\left(x_{p1i} - x_{fj}\right)^2 + \left(y_{p1i} - y_{fj}\right)^2} \quad \ldots \ldots \quad (7)$$

$$\mathrm{dist}(p_2, \mathrm{f}) = \sqrt{\left(x_{p2i} - x_{fj}\right)^2 + \left(y_{p2i} - y_{fj}\right)^2} \quad \ldots \ldots (8)$$

$$f(x_{sel}) = \begin{cases} P_1 \,, if(dist(P_1, f) < dist(P_2, f) \\ P_2 \,, if(dist(P_1, f) > dist(P_2, f) \end{cases} \ldots \ldots \ldots \quad (9)$$

The selected $f(x_{sel})$ robot(s) are released to replace the failed robots based on the queue index as in Fig 2. The numbers of idle robots in the queue are positioned according to their index in the two positions around system boundary as in Fig. 3. The robot with the first index is selected i.e., First in First Out (FIFO) strategy is applied to select the next active robot from the idle robots.

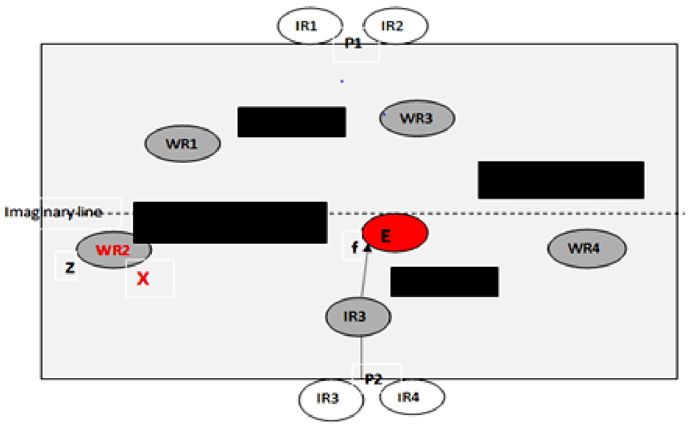

Fig. 2. Idle robot positing in the field and replacement at the time of failure

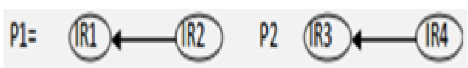

Fig. 3. Queue positioning of robots according to FIFO index

As we see from Fig. 4 failure of robot may be in two cases; failure to the temporary commander (TMC) and failure of any robot $R\ (i)$. If the failed robot$R(i)$ is TMC, the hunting task will be little bit difficult as the TMC is the leader, especially at the maturity stage (pursuing and catching stage) all the robots following the TMC are unable to communicate. Although the robots $R-1$ are unable to communicate with TMC, it doesn't mean that the robots are stopping functioning. The $R-1$ robots continue as allies by switching the current state to search beside this; any of these robots will broadcast a message to the idle robots (IR) accordingly one idle robot will join the group to replace the failed TMC. If any robot rather than TMC is failed, there are two cases; whether continue with the current state if it is in the maturity (Pursuing or catching) stage or to replace the robot if it is in the infant (searching stage).

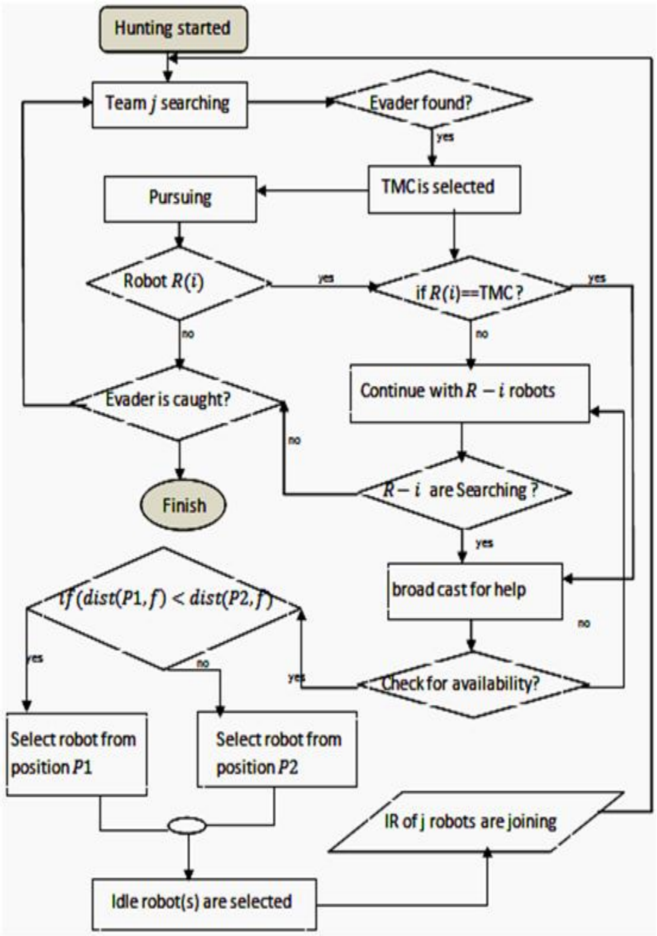

Fig. 4. General flow chart diagram of Robust Multi robot hunting

3. Experiments & Results

According to the above description the research is implemented using SimIAm in MATLAB programming language. Some of the simulation results are shown below.

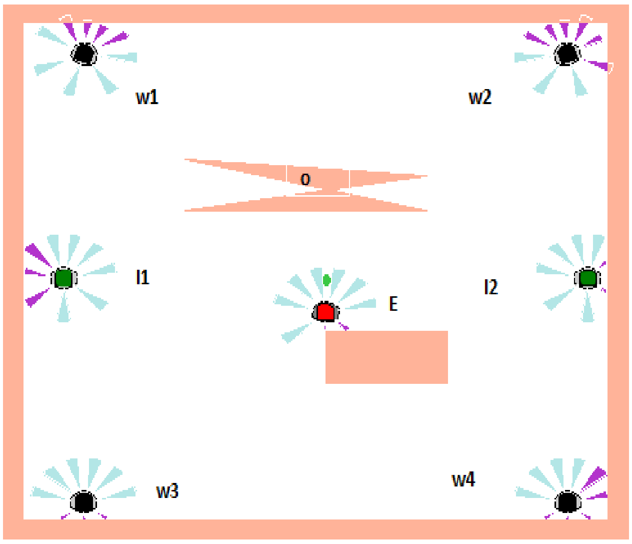

Fig. 5. The simple simulation model of the system

Where $W1, W2, W3$, and $W4$ are working robots, $I1, I2$ are idle robots, O and E are the obstacle and evader. The working robots are placed at any arbitrary location; and the exact location of the evader is not known to them. However; the robots assume that the evader is found in the boundary.

Initially angular and linear velocity of the idle robots is set to zero.

Velocity= $\begin{cases} v = 0 \\ w = 0 \end{cases}$

Where; v is the linear velocity and w is angular velocity.

The idle robots are initially positioned at the center of $(0, -y)$ or $(-x, 0)$ and $(0, +y, or\ (+x, 0)$ of the boundary. If any robot is failed and the remainder robots unable to catch the evader the robot with the closet distance (according to Equation 7, 8,9) to the evader after failure is going to replace the failed robot.

As we see from Fig. 6 the evader is bounded with three of the robots and the obstacle before any failure is occurred; there is no room for other robots to join the group even the allies. Three of the robots and the obstacles are within the evader maximum escaping range; so the evader can't escape anywhere. The hunting task is over without the demand for other late member robot. The robot with the minimum steps(time) to the evader is considered as the temporary commander(TMC) .The TMC should have to be in Watching state until the other robots join and the situations to catch the evader are meet.From the above Fig.7 Simulation result of the robots shown that, Robot3 is selected as TMC .Robot3 reaches early at catching stage as shown by the green dotted line all the robots are hiding toward the evader. The robots are going through stages of avoiding obstacle and avoiding obstacle and going to goal (AOGTG) still the robots are within the Max_range of the individual robot capability.Finally; three of the robots are catching the evader at 5.567 simulation ticks as shown above. However robot4 doesn't reach the catching stage.

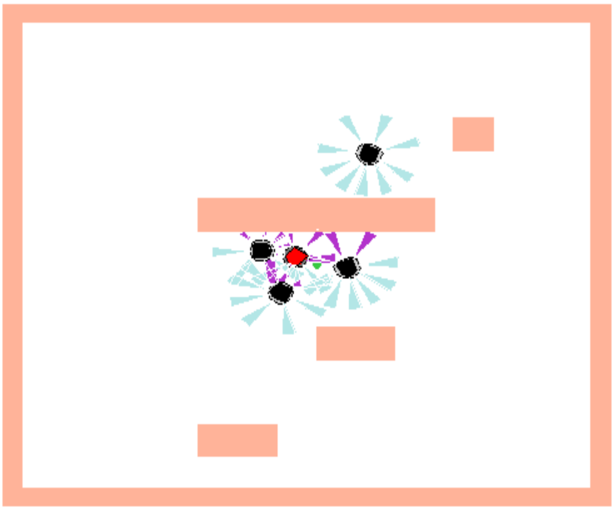

Fig. 6. Probability of catch with $N - i$ robots

Although the obstacles are cause communication overhead, they can also create opportunity for the robots to catch the evader. As the simulation shows there is probability that the evader be in difficult situation. These situations are the evader may be trapped at any boundary/obstacle by two or even one robot. This shows even we have N number of robots participate in the hunting field, the $N - i$ robots should have to continue without affected by the failed i robot.

As we see from the from Fig. 8(a), 9 even the probability is less robots failed due to collisions with obstacles, other robots or many other reasons. So we should have to guarantee the continuity of the hunting task by running the task with the remainder robots as in Fig .8(a)or replacing the failed robot as in Fig .9. The simulation result in Fig 8(b) shows the movement of the robots i.e. the blue the four blue lines shows the four robots and the red line (mark) shows the evader movement. The simulation result of Fig. 9 works as follows; first the four robots are assigned to perform the hunting task. During the searching states robot1 and robot3 are crashed on the way, the remainder two robots try to catch the evader. As the maximum threshold time is reached the first closest idle robot (using equation 6, 7, 8) (robot5) is assigned and join the alliance, but even now the robots can't able to catch the evader. Finally then the next robot (robot6) is assigned and join the alliance catch the evader with four robots.

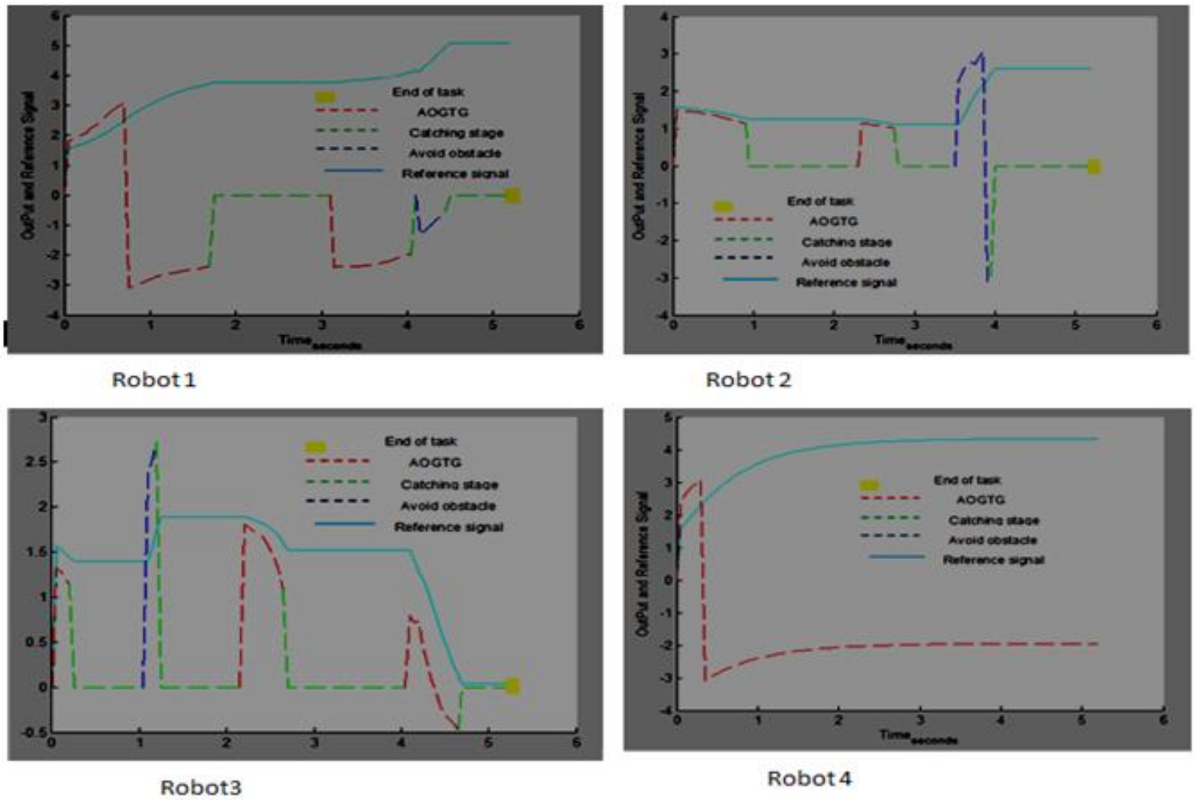

Fig. 7. The Simulation of the hunting task of Fig .6 catching evader with three robots before

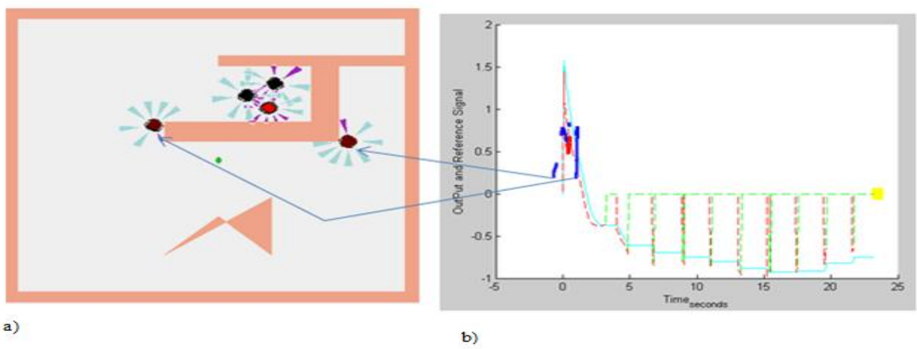

Fig.8. The Evader is caught with only two robots with help of obstacle after two robots are failed.

We held total of 20 simulation experiments; initially four robots are going to search for the one evader and two backup robots .The results of the simulation are discussed in the table 1 below.

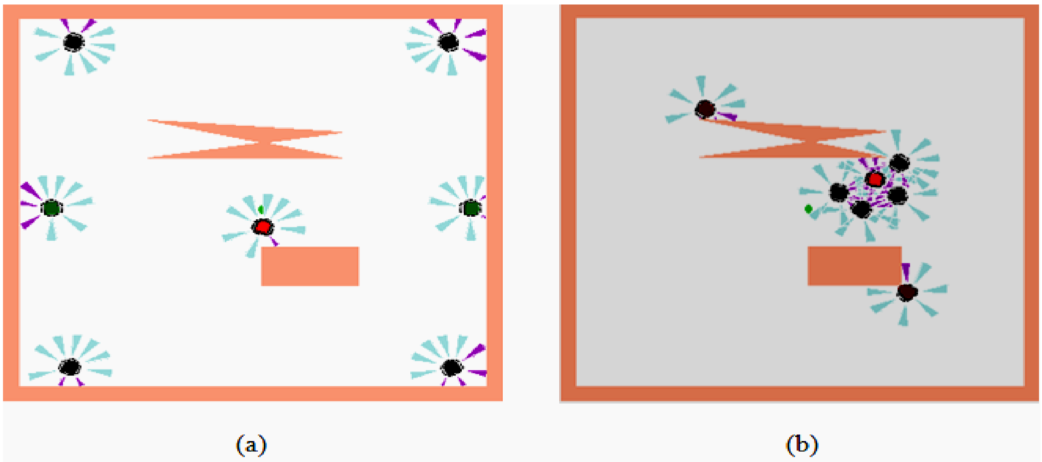

Fig.9. (a) The default positioning of the robots where black, green, red robots are the working, idle, and the evader respectively (b) The Evader is caught with four robots after failure of two robots (robots in maroon color) is occurred

The four robots are tray to catch the evader; out of the 20 simulation experiments the robots catch the evader 8 times (40%) without fault. Three robots are able to catch the evader 4 times (20%) and 3 times (15%) before and after failure respectively. The hunting task is accomplished with four robots after the failed robot is replaced five times the robots able to catch the evader it is about 25%.

Finally; the general findings and comparisons of this paper and previous journals is discussed in table 2.

TABLE I: The advantage of the obstacle to catch the evader

Evader caught with	Condition	Number of robot replaced	Number success
4 robots	Before failure	-	8
3 robots and obstacle	Before failure	-	4
3 robots and obstacle	After failure	-	3
4 robots	After failure	1	5
4robots	After failure	2	3

TABLE II: Comparing of the existing system and the this system in solving the hunting task and general process

Characteristics	Existing[1]	Proposed approach
Idle Robot	-	yes
Functioning with $N - i$ robots	-	yes
Buck up robot localization	In the hunting field searching for evader without joining any group. The state of the robot is working this leads to wasting of robot resource.	The idle robots are placed at the center of the any boundary as in Fig.5 this saves the robot resource battery, risk of failure. The state of the robot is idle.
Resources Minimization	Only tries to catch the evader with N i.e. 4 robots	Can catch the evader with $N - i$ robots as demand to catch the evader are meet

Speed to catch evader	Since the algorithm is rigid i.e. tries to catch evader only with four robots without considering the role of obstacles; the time to catch the evader may take time.	It is fast; it can use situations if the evader can't escape. The role of obstacle to catch evader is considered. and has important role
Using situations(obstacle)	It doesn't use important situations (role of obstacle) only catches with four robots.	Considers obstacle as important factor

5. Conclusion

This paper studied a multi-robot hunting problem under failure of robots. Proposed research tries to complete the hunting task in the presence of failure of one or multiple robots. Simulation result shows that the proposed approach add robustness by completing the task with the remaining robots, thereby minimizing resources being used in the hunting. The robots communicate and cooperate each other to catch the evader even at the time one or more robots are failing from the allies. Furthermore, the proposed approach is flexible enough to solve the problems when the robot breaks down without restarting the system. This research can be used not only in robotic hunting but we can use it in any system from robotic bomb exploration to plain attack.

Proposed work can be further extended using multiple evaders and evaluating the performance on simulator and actual physical test bed.

REFERENCES

[1]. Liu TM, Lyons DM. Leveraging area bounds information for autonomous decentralized multi-robot exploration. Robotics and Autonomous Systems. 2015 Dec 31;74:66-78, DOI: 10.1016/j.robot.2015.07.002

[2]. Huntsberger T, Stroupe A, Aghazarian H, Garrett M, Younse P, Powell M. TRESSA: Teamed robots for exploration and science on steep areas. Journal of Field Robotics. 2007 Nov 1;24(11-12):1015-31. DOI: 10.1002/rob.20219

[3]. Schenker PS, Huntsberger TL, Pirjanian P, Baumgartner ET, Aghazarian H, Trebi-Ollennu A, Leger PC, Cheng Y, Backes PG, Tunstel E, Dubowsky S. Robotic automation for space: planetary surface exploration, terrain-adaptive mobility, and multirobot cooperative tasks. InIntelligent Systems and Advanced Manufacturing 2001 Oct 5 (pp. 12-28). International Society for Optics and Photonics. doi:10.1117/12.444181

[4]. Dharia SP, Falcone T. Robotics in reproductive medicine. Fertility and Sterility. 2005 Jul 31;84(1):1-1, DOI: 10.1016/j.fertnstert.2005.02.015.

[5]. Thangavel K, Balamurugan A, Elango M, Subiramaniyam P, Senrayan M. A survey on nano-robotics in nano-medicine. Journal of NanoScience and NanoTechnology. 2014 Feb 8;2(1):525-8, DOI: 10.1016/j.fertnstert.2005.02.015

[6]. Kehoe B, Patil S, Abbeel P, Goldberg K. A survey of research on cloud robotics and automation. Automation Science and Engineering, IEEE Transactions on. 2015 Apr;12(2):398-409, DOI: 10.1109/TASE.2014.2376492.

[7]. Zheng Y, Bekey G, Sanderson A. Robotics for biological and medical applications. DRAFT REPORT. 2005 Aug:49.

[8]. Ni J, Yang SX. Bioinspired neural network for real-time cooperative hunting by multirobots in unknown environments. Neural Networks, IEEE Transactions on. 2011 Dec;22(12):2062-77. DOI: 10.1109/TNN.2011.2169808

[9]. Cao Z, Gu N, Tan M, Nahavandi S, Mao X, Guan Z. Multi-robot hunting in dynamic environments. Intelligent Automation & Soft Computing. 2008 Jan 1;14(1):61-72, DOI: 10.1080/10798587.2008.10642983

[10]. Zhu D, Lv R, Cao X, Yang SX. Multi-AUV Hunting Algorithm Based on Bio-inspired Neural Network in Unknown Environments, 2015 Nov, DOI: 10.5772/61555.

[11]. Cao Z, Zhou C, Cheng L, Yang Y, Zhang W, Tan M. A Distributed Hunting Approach for Multiple Autonomous Robots. International Journal of Advanced Robotic Systems. 2013 Jan 1;10 , DOI: 10.5772/53410.

[12]. Madden JD, Arkin RC, MacNulty DR. Multi-robot system based on model of wolf hunting behavior to emulate wolf and elk interactions. InRobotics and Biomimetics (ROBIO), 2010 IEEE International Conference on 2010 Dec 14 (pp. 1043-1050). IEEE. DOI: 10.1109/ROBIO.2010.5723472

[13]. Belkhouche F, Belkhouche B, Rastgoufard P. Multi-robot hunting behavior. InSystems, Man and Cybernetics, 2005 IEEE International Conference on 2005 Oct 10 (Vol. 3, pp. 2299-2304). IEEE. DOI: 10.1109/ICSMC.2005.1571491

[14]. Benbouabdallah K, Qi-dan Z. A Fuzzy logic behavior architecture controller for a mobile robot path planning in multi-obstacles environment. Research Journal of Applied Sciences, Engineering and Technology. 2013 Apr;5(14):3835-42. DOI=10.1.1.476.4439

[15]. Huang TY, Xue-Bo CH, Wang-Bao XU, Zi-Wei ZH, Zhi-Yong RE. A self-organizing cooperative hunting by swarm robotic systems based on loose-preference rule. ActaAutomaticaSinica. 2013 Jan 31;39(1):57-68. doi= 10.1016/S1874-1029(13)60007-5.

[16]. Parker LE. Multiple mobile robot systems. In Springer Handbook of Robotics 2008 (pp. 921-941). Springer Berlin Heidelberg, doi=10.1.1.381.1344.

[17]. Chaimowicz L, Sugar T, Kumar V, Campos MF. An architecture for tightly coupled multi-robot cooperation. InRobotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on 2001 (Vol. 3, pp. 2992-2997). IEEE, DOI: 10.1109/ROBOT.2001.933076.

[18]. Agrawal P, Agrawal H. A REVIEW ON MULTI ROBOT COOPERATION USING BIO INSPIRED NEURAL NETWORKS. 2013 Nov , 9(4). doi=10.1.1.671.6640

[19]. Saeedi S, Paull L, Trentini M, Li H. Neural network-based multiple robot simultaneous localization and mapping. Neural Networks, IEEE Transactions on. 2011 Dec;22(12):2376-87.DOI: 10.1109/TNN.2011.2176541

[20]. Sharan P, Mutyal M, Agrawal H, Agrawal P. Simulation study of multi-robot for Intelligent Transportation System. In India Conference (INDICON), 2014 Annual IEEE 2014 Dec 11 (pp. 1-5). IEEE. DOI: 10.1109/INDICON.2014.7030617

[21]. Yang SX, Meng M. Neural network approaches to dynamic collision-free trajectory generation. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on. 2001 Jun;31(3):302-18. DOI 10.1109/3477.931512.

[22]. Yang SX, Meng MQ. Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach. Neural Networks, IEEE Transactions on. 2003 Nov;14(6):1541-52.; doi: 10.1109/TNN.2003.820618.

[23]. Kwon JW, Kim JH, Seo J. Multiple Leader Candidate and Competitive Position Allocation for Robust Formation against Member Robot Faults. Sensors. 2015 May 6;15(5):10771-90.; doi:10.3390/s150510771

[24]. Kwon JW. Cooperative environment scans based on a multi-robot system. Sensors. 2015 Mar 17;15(3):6483-96, DOI=10.1.1.470.132

[25]. Khalid Hasnan, Qadir Bakhsh, Aftab Ahmed, Bhagwan Das, SanamAyub. A Novel Hybrid Locomotion Mechanism for Small Mobile Robot, Indian Journal of Science and Technology, 2015, 8(17), pp. 1-5.

[26]. K. Kanaka Vardhini, T. Sitamahalakshmi. A Review on Nature-based Swarm Intelligence Optimization Techniques and its Current Research Directions. Indian Journal of Science and Technology, 2016, 9(10), pp. 1-13.

[27]. Balch T, Arkin RC. Behavior-based formation control for multirobot teams. Robotics and Automation, IEEE Transactions on. 1998 Dec;14(6):926-39, doi=10.1109/70.736776

[28]. Carlson J, Murphy RR, Nelson A. Follow-up analysis of mobile robot failures. InRobotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on 2004 Apr 18 (Vol. 5, pp. 4987-4994). IEEE. DOI=10.1.1.1.9684

[29]. Carlson J, Murphy RR. Reliability analysis of mobile robots. In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on 2003 Sep 14 (Vol. 1, pp. 274-281). IEE, DOI: 10.1109/ROBOT.2003.1241608.

[30]. Kahan T, Bukchin Y, Menassa R, Ben-Gal I. Backup strategy for robots' failures in an automotive assembly system. International Journal of Production Economics. 2009 Aug 31;120(2):315-26.DOI: 10.1016/j.ijpe.2007.09.015.

[31]. Twala B. Robot execution failure prediction using incomplete data. InRobotics and Biomimetics (ROBIO), 2009 IEEE International Conference on 2009 Dec 19 (pp. 1518-1523). IEEE., doi=10.1109/ROBIO.2009.5420900