

Proposed Methodology for Application of Human-like gradual Multi-Agent Q-Learning (HuMAQ) for Multi-robot Exploration

Dip Narayan Ray, Somajyoti Majumder

SR Lab, CSIR-CMERI, Durgapur, West Bengal, India 713209

{dnray,sjm}@cmeri.res.in

Abstract. Several attempts have been made by the researchers around the world to develop a number of autonomous exploration techniques for robots. But it has been always an important issue for developing the algorithm for unstructured and unknown environments. Human-like gradual Multi-agent Q-learning (HuMAQ) is a technique developed for autonomous robotic exploration in unknown (and even unimaginable) environments. It has been successfully implemented in multi-agent single robotic system. HuMAQ uses the concept of Subsumption architecture, a well-known Behaviour-based architecture for prioritizing the agents of the multi-agent system and executes only the most common action out of all the different actions recommended by different agents. Instead of using new state-action table (Q-table) each time, HuMAQ uses the immediate past table for efficient and faster exploration. The proof of learning has also been established both theoretically and practically. HuMAQ has the potential to be used in different and difficult situations as well as applications. The same architecture has been modified to use for multi-robot exploration in an environment. Apart from all other existing agents used in the single robotic system, agents for inter-robot communication and co-ordination/ co-operation with the other similar robots have been introduced in the present research. Current work uses a series of indigenously developed identical autonomous robotic systems, communicating with each other through ZigBee protocol.

1. Introduction

The research on mobile robotics has now reached a stage, where it is being considered as an obvious choice for many applications, such as office/factory/hospital automation, personal care, domestic aids, exploration in land, water and other planets. Several mobile robots as well as navigation algorithms have been developed over the time by different researchers around the world. Some of them are autonomous and some of them are not only autonomous, but also intelligent in nature. Still there are several reasons where single robot cannot be preferred. The main tasks of these mobile robots for most of these cases are primarily autonomous explorations, general assistance, data collection of the unknown or slightly known environment. In such a situation predefined and preprogrammed approaches are inadequate to meet the challenges posed by the environmental dynamics. Another major constraint encountered by mobile robotics researchers is the robot's capability to process large amount of information in real time, obtained from various sensors in a dynamic and an unstructured environment. Also if the area is too large, it is difficult to explore the whole area by a robot in time due to limited on-board power. This time constraint becomes more important when a robot is deployed for security and surveillance purpose, such as landmine detection or bomb/ explosive detection in public places.



For the above reasons instead of using single robot, multiple robots can be used. This may effectively decrease the computational load due to handling large amount of data obtained from different sensors as well as drastically reduce overall exploration time for a large stretch of area. Still the challenge of exploration in completely unknown and unimaginable environments (that the programmer cannot think of) remains unanswered.

This has led towards the development of a generalized navigational (exploration) algorithm for safe navigation (towards the goal), that can be divided into multiple sensor specific subtasks. So that, in case of indoor exploration, these subtasks can be reordered in terms of priority as check battery power, seek goal, follow walls and corridors, detect door, detect obstacles and look for objects of interest, for example. For outdoor exploration, these can be re-ordered as check battery power, seek goal, detect obstacles and look for objects of interest. Capability of such approach is essential for the reason that in natural outdoor environment clearly lacks very common and easily identifiable geometric features such as corners or lines for example.

It is also a necessity to reduce the computational complexity and processing over load to achieve a reasonable response time for the system. This has been achieved by using the concept of multiagent system together with behaviour-based paradigm for certain states. The resultant performance has been found to be very satisfactory. Again for each of the sub-tasks, agent based approach has been implemented whose functional responsibility is to gather the state (s) from the sensors and suggest/predict action (s). All the different actions from different agents are then modified through a coordination mechanism to produce a single definitive action. The interaction of the system with the environment makes the system to act rationally and effectively across various operating conditions starting from structured indoor environment to much unstructured outdoor environment under varying terrains and dynamic conditions.

Several works on multi robot exploration have been reported in literatures [1], [2], [3]. Different unsolved problems in this field emphasizing the theoretical issues have been presented in [4] along with a critical survey. Dudek et al. [5] has described the taxonomy for multi-agent robotics. The exploration strategies have been inspired from the fields of Artificial Intelligence, game theory, distributed computing, theoretical biology, animal ethology, artificial life etc. Initial inspiration as found in [6] was from the animal world. Arkin proposed a methodology for foraging and retrieving objects in a hostile environment by multiple mobile robots through co-operation without communication. Motor schemas, such as move-to-goal, avoid-robot, avoid-static-object and maintain-formation are used with the help of an arbiter to reach the goal. This work has been further extended for a mobile robot team by Balch & Arkin [7]. The work of Kube and Zhang [8] is based on the behaviours of different social insects, mostly ants and bees. They have implemented the box-pushing task using subsumption architecture [9], [10] as well as adaptive logic networks (ALN). Fukuda et al. [11] have also described similar studies using analogs to animal behaviour. A three layered control methodology for cooperation between heterogeneous mobile robots has been proposed in [12] for indoor environment. Those three layers are functional level, control level and high-level decision making planner level.

Recently, researchers are engaged to develop methodology for exploration by multiple mobile robots while making a group formation. Such line formations and general formations of non-holonomic robots have been studied by Yamaguchi [13] and Yoshida et al. [14]. Decentralised control laws using potential field approach has been implemented in [15] and [16] to guide the mobile robots away from obstacles. Similar approach for controlling a formation, guarding a perimeter and surrounding a facility has been described in [17] along with the stability analysis. Burgard et al. [18] have described an approach for the coordination of multiple robots for exploring an unknown environment taking into account the cost of reaching a target point and its utility. Similar approaches for exploration and mapping by multiple mobile robots have been presented in [19], [20], [21], [22]. Still the developments of different control laws/ architectures for autonomous navigation and exploration are going on as found in [23], [24]. HuMAQ is the new addition in this list. Initially the development of

HuMAQ was for Multi-agent systems, not necessary multiple systems, but finally it has been found that it has the potential for being used for multiple systems without any change.

2. Human-like-gradual Multi Agent Q-learning (HuMAQ)

Human-like Gradual Multi Agent Q-learning (HuMAQ) is a methodology already applied in [25] for single robotic system. It uses multiple agents as shown in Fig 1(a) for describing different behaviours/tasks of the robot depending up on its type of applications.

Suppose a system has 'N' numbers of agents responsible for different states generated by the sensors and the internal states. In a simple system, two states of the robot can be handled by each agent. In case of a complex system, each agent can take care of 'n' states. Depending on the states, each agent selects different actions from the state-action table (Q-table) and interacts with the Co-ordinator. This co-ordinator arranges the agents to be invoked according to the predefined priority as shown in Fig 1(b). Additional details have been discussed with specific examples in [26].

Here the Co-ordinator plays an important role as action modifier, also referred as Modifier for simplicity. The modifier refines the actions proposed by different agents depending upon their priorities. For example, if the goal-seeking agent has directed the robot to move in forward direction, but the obstacle-avoidance agent has detected obstacle in the front. This clearly shows a conflict resolution required which is done by the modifier. It then searches for possible alternate position from where the goal is nearer and refines the action to move towards this alternate position. The modifier in essence tries to find a common area of interest where the states from most of the different agents are available and then refines the action from the descending list of states from that common space accordingly. It is not mandatory that all the states from all the agents should be present, because at any given instance of time all the agents may not be active. With the increase in number of agents, the complexity of the Co-ordinator and modifier also increases. Different rules for selection of the refined action by the modifier can be formed depending up on the type of sensors and applications. However, the basis of such rules should always be common space sharing.

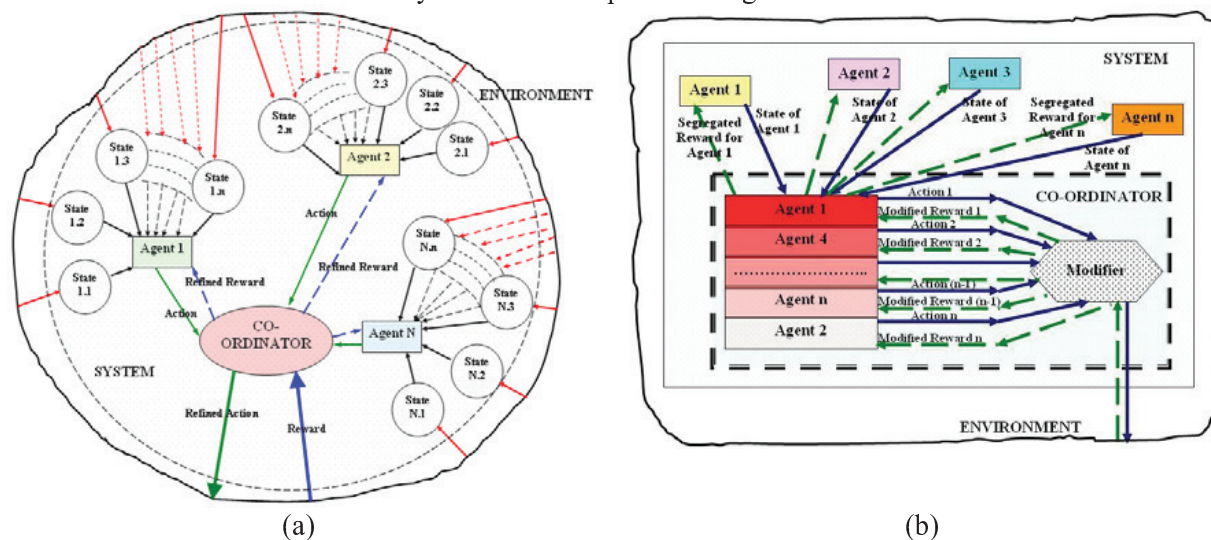


Fig – 1: (a) Generalised Multi-agent Q-learning model is divided in two main parts: different Agents and the Co-ordinator (b) Generalized Co-ordinator for HuMAQ, also known as modifier, modifies the actions for different agents depending upon the priority and distributes the rewards accordingly. The refined action is now performed by the actuators and the obtained reward is again sent to the modifier to distribute them as per the priority of the agents only to those, which were active during determination of states. Therefore, instead of a single reward as in the case of Q-learning, a distributed and gradually reducing weightage system has been adopted here. A variable weightage system has been found to be the most suitable and therefore used to account the validity in priority of the agents.

For example, if there are four agents and the total reward obtained is R for any instance of time, when all the four agents were responsible for a refined action, it can be distributed as:

$$R = w_1 R_1 + w_2 R_2 + w_3 R_3 + w_4 R_4 = \sum_{i=1}^4 w_i R_i \quad \dots\dots\dots (1)$$

Where $w_1 > w_2 > w_3 > w_4$; w_1, w_2, w_3, w_4 are the distributed weightage for rewards for first agent (highest priority), second agent, third agent and fourth agent (low priority) respectively. R_1, R_2, R_3 and R_4 are the individual rewards for first agent, second agent, third agent and fourth agent respectively obtained directly due to system-environment interaction. With these separate rewards $w_i R_i (i \in 1, 2, 3, 4)$, the separate Q-tables for different agents update themselves gradually. After a prolong trial run in any exploration task, if the system needs to be recharged or powered off for any reason, the final Q-table at that instance is stored and will be used as initial table for next run onwards.

3. Single Robot Vs. Multiple Robots

3.1. The Robotic Prototypes

As already mentioned, earlier works [24], [25] on HuMAQ used single robotic system. For this purpose the indigenously developed robot, called ARBIB (Autonomous Robot Based on Intelligent Behaviours) was used. The basic hardware module of ARBIB consists of a power supply, controlling and processing unit, sensor suite, communication module and the motors. The sensor suite includes three different pair of sensors i.e. light sensor (left and right), IR sensor (left and right) and battery-level indicator. The power supply is provided from an on-board 12 V Li-Ion battery bank. Only the battery-level indicator is connected directly to the power supply. Other sensors are obtaining power from the controlling/ processing board. Battery-level indicator constantly monitors the on-board battery voltage and informs the controlling unit when it goes below a pre-defined threshold level. The data from the sensors are fed to the controlling unit through one of their three terminals. The communication with the base station/ operator console has been established using commercially available Bluetooth module. The two motors are controlled via the H-bridge amplifier. The power supply for the motor is fed into the H-bridge and the control signal for the motor along with the change of directions is provided from the controlling/ processing unit to the H-bridge amplifier.

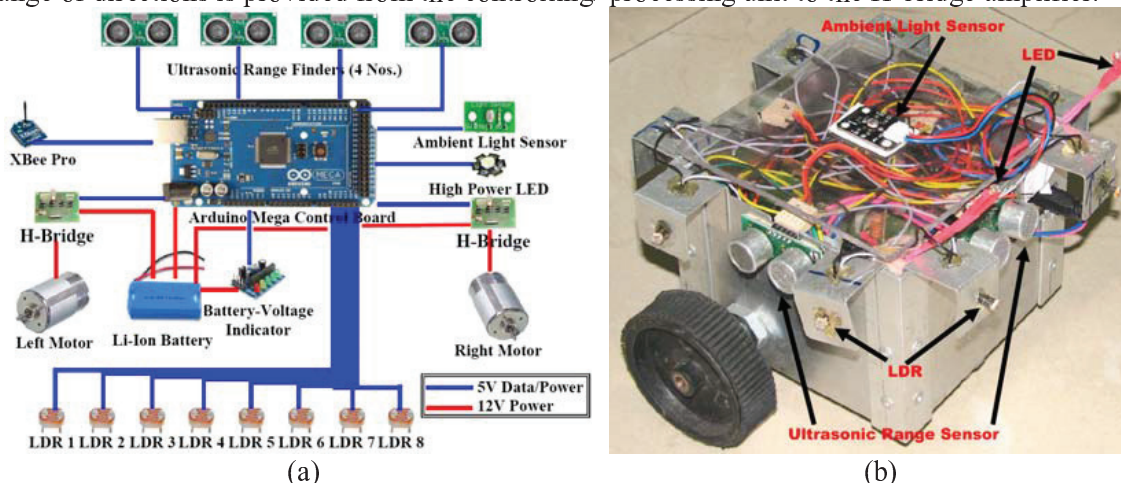


Fig – 2: (a) The hardware architecture (b) Prototype of a single robot showing major sensors

The present work uses ARBIB IV another upgraded version for testing the proposed multi-robot navigation methodology. The basic architecture of single robotic system is same throughout the ARBIB series of mobile robots. All the single robots are identical in all respect (in terms of mechanical structure, hardware and software architectures). All of them have on-board battery supply,

controlling and processing unit, sensor suite, communication unit and two DC motors. Here on each single robot a high-power LED has been mounted for inter-robot communication. Sensor suite of the single robotic system includes four ultrasonic sensors (on each side), eight LDR (two fixed at the top along the side boundary of each side) and one ambient light sensor. The motors are controlled through H-bridge as done in earlier case. Fig 2 (a) and (b) depict the hardware architecture and the prototype of ARBIB IV respectively.

3.2. Communication

The overall communication has been divided in two different categories, inter-robot communication and the robot - base station/operator console communication as shown in Fig 3(a). For inter-robot communication the LED – LDR module has been employed. For sending the signal from the co-ordinator robot, LED of the co-ordinator robot is switched ON. The LDRs on the other robots are attracted towards the light emitting from the LED of co-ordinator and act accordingly. The communication between the robots and the base station/ operator console is established using ZigBee protocol. However, this communication is one way i.e. from the robot to the base station and only for sending data. Data from the robots have a definite header which separates the robots from each other. Initially for inter-robot communication the approach of distance measurement using signal strength was used. But due to carry out the experimentation in a small (a room of 10 Ft X 12 Ft dimension) most of the times, the signal strength was not significant enough for locating the robots as well as their directions.

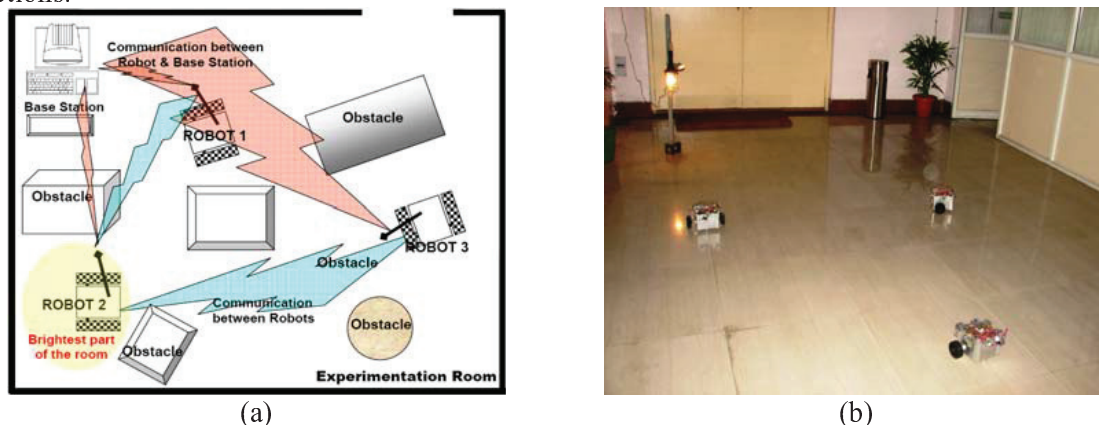


Fig – 3: (a) The inter-robot communication and the robot-base station communication strategy for multi-robot exploration (b) A view of the experimental area

4. Proposed Methodology

Here three homogeneous robots are deployed in a room as shown in Fig – 3(b) to find the brightest part of the room. As already mentioned four different sets of sensors and four related behaviours have been used. The four behaviours according to descending priority are Hunger, Obstacle-avoidance, Co-ordination and Goal-seeking. Hunger is related to battery level indicator and highest priority behaviour. This is referred by sub-states ‘0’ (below threshold value) or ‘1’. Corresponding actions are ‘shut down’ or ‘keep the system on’ respectively. Obstacle-avoidance behaviour is associated with the Ultrasonic range finders and helps the robots to keep away from static and slow-moving obstacles. These four sensors are numbered 1, 2, 3, 4 clockwise when viewed from the top starting from the side having the castor wheel just at the bottom and the obstacles are defined at near [less than 20cm] (representing sub-state ‘3’), far [20 – 60cm] (‘2’) and farthest [above 60cm] (‘1’). Accordingly the action is chosen in terms of direction and speed as shown in Table - II. Any of the three robots starts acting as co-ordinator when it reaches the predefined threshold value of the light read by the ambient light sensor. As soon as it reaches the area the eight LDRs mounted on the sides are made OFF and the LEDs are switched ON. Otherwise all the time the lowest priority behaviour, goal-seeking, is active. If there is no co-ordinator it uses the ambient light sensor to find out the maximum intense zone. It

compares the current reading with the previous one. It may be either greater (sub-state '2') or lesser ('1') or same ('0'). If co-ordinator is present it will follow co-ordinator using LDRs to reach the goal. The LDRs are also numbered in similar way as done in case of the ultrasonic range finders. Here the sub-states are '1' (for output voltage of 0.1-1V), '2' (2.1-3V), '3' (3.1- 4V), '4' (4.1 -5V) and '0' (0V). The action for this agent is chosen in terms of direction as shown in Table – III.

Table – I: The actual representation of the state table as viewed by the robot

Sub-states for the eight LDRs for consecutive four sides								Sub-states for the URFs mounted on four sides				Battery -level	Current light value greater or not?
4	3	2	2	1	1	0	0	0	0	0	0	0	1
LDR [scale is 0, 1, 2, 3, 4 for eight LDRs]								Ultrasonic Range Finder [Scale is 0, 1, 2, 3 for four nos. of URFs]				Battery Level Detector [Low or high]	Ambient light sensor

Table – II: Action Table for Obstacle-avoidance Agent

Position/distance of obstacles	Near (less than 20cm)	Far (between 20 - 60 cm)	Farthest (more than 60cm)
RPM of DC Motors	15	30	60

Table – III: Action table for the Agent associated with goal-seeking behaviour

Serial of LDRs	0	1	2	3	4	5	6	7
Clockwise rotations	0 (0 × 45°)	45° (1 × 45°)	90° (2 × 45°)	135° (3 × 45°)	180° (4 × 45°)	225° (5 × 45°)	270° (6 × 45°)	315° (7 × 45°)

The above mentioned (sub) states and actions are denoted only in terms of numbers. Further details can be obtained from [25]. The most common action is chosen by the modifier as described earlier and the rewards obtained are updated in the corresponding state-action table (Q-table). The weightages for the most active agents (i.e. obstacle-avoidance and goal-seeking) are 0.6 and 0.4 respectively.

5. Experiments, Results & Discussion

A number of experiments have been carried out in a room of the laboratory as shown in Fig 3(b). A 100W lamp has been used as the light source in a slight dark environment. All the three robots are deployed at one place away from the light source facing each other in closed triangle. On reaching the predefined threshold value of the ambient light the robots stop at that position.

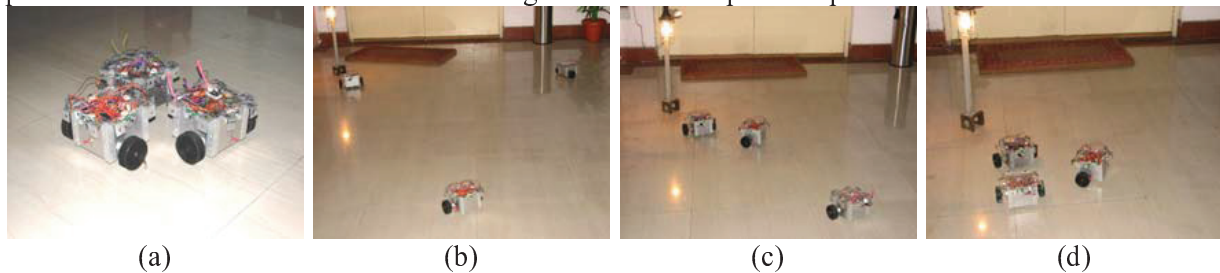


Fig – 4: (a) At the starting (b) First robot has reached the goal and started acting as co-ordinator (c) Second robot has reached the goal (d) Third robot has also reached the goal

As shown in Fig 4(b) the first robot has reached the goal and started acting as co-ordinator. The second and third robots reached the goal afterwards following the signal from the co-ordinator. The data generated are directly transmitted to the base station through XBee. As soon as any robot reaches the goal it stops sending data. These data are further processed and represented in Table – IV.

Table – IV: Nos. of updates for reaching the goal for different trial runs

Nos. of updates	Run 1	Run 2	Run 3	Run 4	Run 5
Robot 1	298	239	216	232	182
Robot 2	283	291	263	274	237
Robot 3	254	281	248	220	256

It has been found that robot which acts as the co-ordinator takes less time than others. In the first run Robot 3 reached the goal first, thereafter Robot 1 and lastly Robot 2. Similar inferences can be made for other runs. If the numbers of updates for different runs are taken into consideration, it can be found the numbers of updates for most of the robots (except Robot 3) are lesser than the Run 1. This has been possible due to implementation of human-like gradual Q-learning. The numbers of updates for different robots versus the trial runs has been presented in Fig 5(a). The initial and final Q-table for different agents will clearly show refinement of state-action pairs which is a proof of learning. The initial and final Q-table for the obstacle-avoidance agent has been presented in Fig 5(b) and (c).

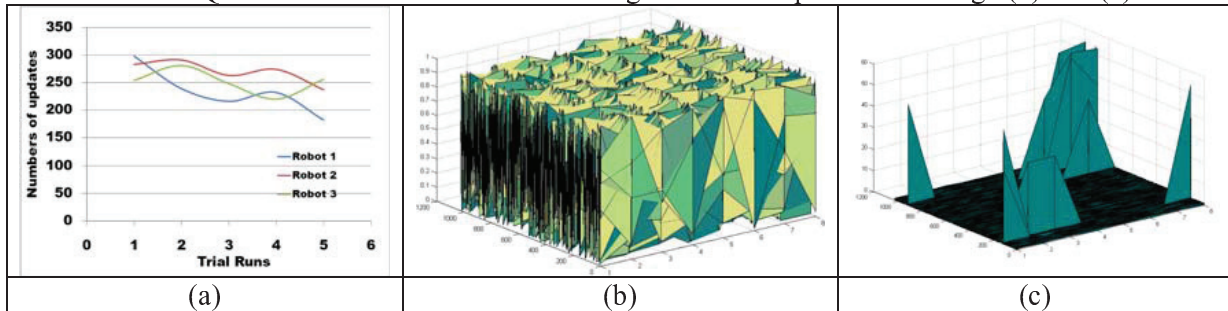


Fig – 5: (a) The number of updates for different robots versus trial runs (b) Initial Q-table for obstacle-avoidance agent (c) Final Q-table for obstacle-avoidance agent

6. Conclusion

Human-like Gradual Multi Agent Q-learning has already been proven for multiagent single robotic system. It has been tested in real time in both indoor and outdoor environments. Proof of learning has also been established. Present work aims to use the proposed HuMAQ for light exploration by a team of homogeneous robots. These indigenously developed robots use four different sensors and four different behaviours for safe exploration. Two different modes of one way communication have been implemented for successful operation. The proposed methodology has been successfully tested in an indoor environment in a limited space. The future scope of the work will include establishment of two way communications between the robots and base station, use of heterogeneous robotic systems and testing in rugged outdoor environments.

References

- [1] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, "Cooperative mobile robotics: Antecedents and directions," in Proc. 1995 IEEE/RSJ IROS Conf., pp. 226–234.
- [2] L. E. Parker, "Current state of the art in distributed autonomous mobile robotics," in Distributed Autonomous Robotic Systems 4, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 3–12.
- [3] J. Ota, "Multi-agent robot systems as distributed autonomous systems," Advanced engineering informatics (2006), pp. 59–70.
- [4] Y. U. Cao, A. S. Fukunaga and A. B. Kahng, "Cooperative mobile robotics: Antecedents and Directions," Autonomous robots, MA, USA (1997), Vol. 4, pp. 7–27.
- [5] G. Dudek, M. R. M. Jenkin, E. Milios and D. Wilkes, "A taxonomy for Multi-agent Robotics," Autonomous Robot (1996), Vol. 3, pp. 375–397.
- [6] R. C. Arkin, "Cooperation without communication: Multiagent schema-based robot navigation," J. Robot. Syst., vol. 9, no. 3, pp. 351–364, 1992.
- [7] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," IEEE Trans. Robot. Automat., vol. 14, pp. 926–939, Dec. 1998.
- [8] R. C. Kube and H. Zhang, "Collective robotics: From social insects to robots," Adaptive Behavior, vol. 2, no. 2, pp. 189–218, Fall 1993.
- [9] R. A. Brooks and A. M. Flynn, "Fast, cheap and out of control: A robot invasion of the solar system," J. Br. Interplanet. Soc., vol. 42, pp. 478–485, 1989.

- [10] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robot. Automat.*, vol. RA-2, pp. 14–23, Mar. 1986.
- [11] T. Fukuda, H. Mizoguchi, K. Sekiyama, and F. Arai, "Group behaviour control for MARS (micro autonomous robotic system)," in *Proc. Int. Conf. Robotics and Automation*, Detroit, MI, May 1999, pp. 1550–1555.
- [12] F. R. Noreils, "Toward a robot architecture integrating cooperation between mobile robots: Application to indoor environment," *Int. J. Robot. Res.*, vol. 12, no. 1, pp. 79–98, Feb. 1993.
- [13] H. Yamaguchi and T. Arai, "Distributed and autonomous control method for generating shape of multiple mobile robot group," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, vol. 2, 1994, pp. 800–807.
- [14] E. Yoshida, T. Arai, J. Ota, and T. Miki, "Effect of grouping in local communication system of multiple mobile robots," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, vol. 2, 1994, pp. 808–815.
- [15] P. Molnar and J. Starke, "Communication fault tolerance in distributed robotic systems," in *Distributed Autonomous Robotic Systems 4*, L.E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 99–108.
- [16] F. E. Schneider, D. Wildermuth, and H.-L. Wolf, "Motion coordination in formations of multiple robots using a potential field approach," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 305–314.
- [17] J. T. Feddema, C. Lewis and D. A. Schoenwald, "Decentralized control of cooperative robotic vehicles: Theory and application," *IEEE transactions on Robotics and Automation*, Vol. 18, Albuquerque, NM, USA (2002), pp. 852–864.
- [18] W. Burgard, M. Moors, C. Stachniss and F. Schneider, "Coordinated multi robot exploration," *IEEE traction on robotics*, *IEEE journals and magazines*, Vol. 21 (2005), pp. 376–386.
- [19] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz and B. Steward, "Distributed multi-robot exploration and mapping"
- [20] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun and H. Younes, "Coordination for multi-robot exploration and mapping "
- [21] W. Burgard, M. Moors, D. Fox, R. Simmons and S. Thrun, "Collaborative multi-robot exploration," *IEEE international conference on robotics and automation* (2000), Vol. 1, pp. 476–481.
- [22] Z. Yan, N. Jouandeau and A. A. Cherif, "Multi-robot decentralized exploration using a trade – based approach," *Proceedings of the 8th international conference on informatics in control, automation and robotics*(2) (2011).
- [23] L. E. Parker, "Designing control laws for cooperative agent teams", *Robotics and automation* (1993), Vol. 3, pp. 582–587.
- [24] T. Laengle, T. C. Lueth, U. Rembold, H. Woern, "A distributed control architecture for autonomous mobile robots – implementation of the Karlsruhe multi-agent robot architecture kamara," *Advanced robotics* (1997), pp. 411–431.
- [25] D. N. Ray, Amit K. Mondal, S. Mukhopadhyay, S. Majumder, "A Proposed Methodology for Behaviour-based Multi-Agent Q-learning for Autonomous Exploration", *26th International Conference on CAD/CAM, Robotics and Factories of Future 2011*, Kuala Lumpur, Malaysia, July 26 –28, 2011, Vol. 2, Page – 583 – 593, Eds. Dr. M. Khurshid Khan & Dr. Ni Lar Win
- [26] D. N. Ray, "Autonomous Navigation of Mobile Robots: A Fusion of Behaviour-based Robotics and Reinforcement Learning", *Ph D Thesis*, National Institute of Technology, Durgapur, India, 2012