The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

A feedback-trained robot task assignment system

Jeremy Straub[1, 2], Eunjin Kim[1]

[1]University of North Dakota, Grand Forks, ND 58202, USA
(Tel: 1-701-777-4107, Fax: 1-701-777-3330)

[2]jeremy.straub@und.edu

Abstract: Many applications (such as search and rescue or planetary exploration) require robots to characterize an environment that they have little or no initial information about. In this type of a scenario, a team of heterogeneous robots can be used to perform discovery and characterization activities. Unlike a collection of homogeneous robots which can be controlled without regards to the particular abilities possessed (e.g., using many common swarm techniques), the effective use of heterogeneous teams requires dynamic assignment based on constantly changing needs, available skills and robot locations. This paper discusses one such control technique that is demonstrated via a collection of small robots with very limited – but heterogeneous – sensing capabilities.

Keywords: heterogeneous robot team control, robot group communications, robotic control techniques.

1 INTRODUCTION

Task assignment to a collection of heterogeneous robots is an ill posed problem that also presents the complexity of continuous change. Assignment algorithms commonly consider distance from a target, closeness of fit for a given task and other metrics in considering which robot or robots to assign. Various assignment approaches are presented in the literature including self-assignment, central-assignment and swarm-based assignment. This paper proposes a feedback-trained task assignment mechanism which is demonstrated via a foraging application.

Obstacles of different heights were placed in the testing area; robots with sensors mounted at different height levels seek to generate a range measurement for the obstacles. A minimum of two robots is required to identify the height of an object (if it is taller than the highest sensor, for example). A second appropriately equipped robot must confirm each classification. Practically, several robots with different height measurement capabilities will be required to determine the lower and upper bound of the height. Then robots with corresponding height measurement capabilities will be required to confirm these findings. This experimental setup mirrors a variety of real-life conditions where alternate sensor types may be required (for example, sensors focusing on an adjacent range of light spectrum, etc.) to complete a task.

In this application, robots begin by searching for targets of interest and return to this when not assigned an alternate task. When a target is found the robot determines whether it is able to adequately assess it and then either issues a request for confirmation of its assessment or a request for an alternately equipped robot to be sent.

A central planner that runs on a selected robot (any robot with sufficient capabilities can potentially take this role) receives all confirmation or alternate robot requests and tasks a robot based on a combination of existing task-load, closeness to the target and closeness of task fit. The assigned robot receives the task and incorporates it into its task list based on path optimization. The central planner occasionally polls all of the robots for movement and task completion times. This data is used to update the controller's time-cost estimates via a weighted feedback incorporation technique. Updated estimates are sent out to all robots which incorporate the new data (weighed against local condition data) in to their internal costing values.

2 BACKGROUND

Collaborative robotic group control has a myriad of prospective applications. Three of these are particularly relevant to the research that has been conducted and the proposed control techniques: planetary exploration, reconnaissance and search-and-rescue.

It has been proposed, by Fink [1, 2, 3], to perform collaborative planetary exploration (inclusive of Earth science) through the use of a multi-tier robotic group. This concept presents an obvious collection of heterogeneous robots. They are differentiated by their movement type (orbital, flying or ground-based) and their particular function within the group. This mission architecture takes a top-down approach to task assignment. Orbital tier members (with great scope of coverage but limited resolution) identify targets of interest for exploration by aerial tier members which direct the efforts of (and may deploy) ground-based robots.

The multi-tier architecture presented for planetary exploration purposes is also highly relevant for reconnaissance. Work on sensornets [e.g., 4] is a demonstration of collaborative control of multiple heterogeneous robots. While many sensornets do not control the actual actions of their component sensors (robots), they do control the interaction between the data consumer and various sensors for the particular application. The sensornet concept can be looked at as a special case of the multi-tier mission architecture which may focus only on a particular subset of tiers and evidence only limited control.

Terrestrial search and rescue applications are also similar to the aforementioned applications. While this effort may be occurring in what was previously a well-known space, there is no guarantee that its configuration is as-previously-known and no guarantee that any particular infrastructure will be available. In fact, urban search and rescue presents a particular challenge as the environment may contain both physical and electromagnetic hazards to collaborative robotic operation. Robots deployed into search and rescue applications must be self-sufficient and be prepared to encounter electromagnetic interference to group communications and a changing array of hazards. They must also operate under a level of time pressure that is not present in many other robotic applications.

3 COLLABORATIVE CONTROL

To simulate the real-world conditions typical of the operation of heterogeneous group robot operations, each robot in the experimental group is controlled by a separate process that communicates with other robot control processes via message passing. Due to onboard processing limitations of the very basic robot hardware that was used for this experiment and the desire to capture performance information in real time, these processes were all physically located on a single computer that communicates with the robots via Bluetooth. This wireless protocol has limited range, but it was sufficient for the experimental environment used – specifically a small classroom-sized room. The experimental area was a rectangle approximately 10 feet by 20 feet. Three robots were operated concurrently in this area; they searched for and characterized pseudo-randomly placed obstacles.

The robots each begin in exploration mode, but will switch in to exploring traversal mode when another robot detects an object of interest and requests assistance to characterize it. When a robot is in close proximity to an object (or the believed position of an object) it switches into characterization mode. Once the robots have characterized the area completely (no area with a radius greater than a specified value remains unexplored) they switch in to terminal mode.

3.1 Exploration Mode

In exploration mode, the robot moves in a search pattern that is determined based on the presence of known and unknown areas surrounding it. The robot will avoid known obstacles, but prefers unknown (no pheromone) grid locations to those with known (positive pheromone) traverse-cost values. The robot will move will move to maximize its expected utility; however, in this mode, unknown grid locations will be treated as having +10 pheromone, impassible ones will be treated as having -100 and known locations will be treated has having -1 pheromone. The robot will continue in a utility maximizing search pattern until an uncharacterized obstacle is identified, the grid space is completely explored or another robot requests characterization assistance that it is well positioned to offer.

3.2 Exploring Traversal Mode

When a robot receives a suitable request, it evaluates the closeness and priority of the request relative to others that it is presently processing. If the new request has a lower combined (importance weighted against distance and current completion progress) selection value, it is queued; otherwise, the robot begins moving to the new location.

$$SEV = a \bullet I_t - b \bullet D_t - c \bullet C_{cur} \quad (1)$$

A robot that is in exploration mode will service any request that it is suitable to service (that is, it meets the instrumentation requirement and is not beyond the maximum traversal distance away from). Once a robot selects a request, it enters exploring traversal mode. In this mode, the robot selects the lowest-cost route. Unknown grid locations are assigned a cost of zero while those which slow movement (which was not a condition that was tested in this experiment) or contain an obstacle are considered based on their assigned negative utility value. The robot also continues its exploration activities along the path. If an object of interest is detected, it will compare the need to characterize this new object (that it is obviously in close proximity to) to the combined selection metric for its current task. Based on this, it may characterize this newly discovered object or queue it for later revisiting.

3.3 Characterization Mode

When the robot is in close proximity to an object that it has discovered and decided to characterize or one for which it has received a characterization or verification request, it will enter characterization mode. Characterization mode has two goals. First, the robot attempts to perform the characterization or verification activity required. Secondarily, it aims to align its grid with that of any previous robots that have visited this obstacle.

The robot moves around the obstacle and attempts to locate (trigger the bump sensor) from various directions. Through this, the robot characterizes the size of the object (i.e. how many and which grid squares it occupies). This is stored and transmitted to the central control process. If a previous robot has visited, the shape, size and location are compared and the correlation matrix between the grids of the robots are updated.

3.1 Terminal Mode

The robot enters terminal mode when it believes that its exploration area is completely characterized and there are no pending characterization assistance or verification requests remaining. In terminal mode, the robots return to their starting location and attempt to locate each other (through bump sensor triggering). This step is used to perform a final alignment on the grid correlation matrix between robots and once the robots locations are known to each other, they return to their ending formation.

4 COMMUNICATIONS

A simplified version of the communications architecture presented in [5] was used for message passing between the robot control processes and the central controller. Three message types (and their respective response) were used: object located, object characterized and request. Messages are processed by the central controller and forwarded out (containing the ID of the initiating robot, which is important for grid matching use) to the other robots to update their internal state database.

4.1 Object Located Message

The object located message is used to send a preliminary point obstacle to the central control process for distribution to the rest of the robots. This allows other robots to consider the presence of this obstacle when planning their routes during the characterization process. It also communicates that characterization of this object is in process and will prevent another robot that detects the same object during the characterization process from beginning to characterize it.

4.2 Object Characterized Message

When an object has been characterized (or verified) completely (i.e., from all directions), the object characterized message is sent to the central controller. This message contains all grid spaces that the object has been identified to occupy. When the central controller receives this message, it will send the preliminary data to all of the other robots and automatically generate characterization assistance or verification messages (depending on whether the characterization by the current robot was definitive – or simply determined that an alternately configured robot is required for further characterization).

4.3 Request Message

The request message is sent by the central controller once it receives an initial characterization completion notification (via the object characterized message). It will then, if this characterization was definitive (i.e., no other robot is required to characterize the object) issue a verification request; if further characterization is required it will issue a characterization request stating the required characterization capabilities (e.g., sensor above 2.2 cm, etc.). All request messages contain the grid location or locations of the object that requires the robot's characterization efforts and the reporting robot (for grid matching purposes).

4.3 Current State Message

The current state message is sent regularly by each robot to the central controller. The central controller distributes this message to any other robots that are operating in close proximity to the sending robot. This message is designed to allow robots to be aware of each other to prevent collisions.

Fig. 1. Picture of one of the robots that was used for testing the system. This robot has its second bump sensor set at the lowest setting (directly above the lower sensor). Other robots had their sensors at higher settings.

5 COMBINED SYSTEM OPERATIONS

The robotic control in this system is event driven. Once a path is selected for robot exploration or traversal, the control software will continue to execute the movement actions required to follow the designated route until interrupted by the detection of an obstacle, arrival in proximity to a characterization or verification target, receipt of a request that exceeds the selection value of the present task, or the determination that the entire exploration area has been completely explored.

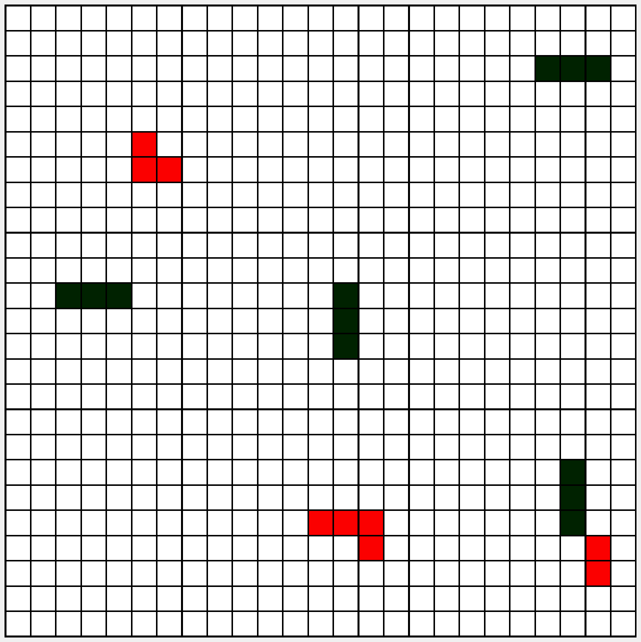

Fig. 2. Partial grid of exploration area showing obstacles (red / light) and demonstrated-traversable areas (green / dark)

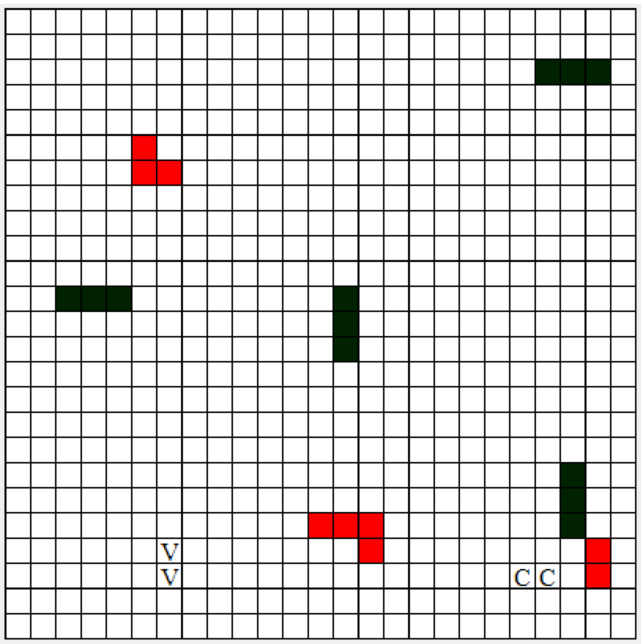

Fig. 3. Partial grid of exploration area showing obstacles, demonstrated-traversable areas and areas that need to be verified (V) or characterized (C)

The system determines the speed of robot movement based on the current mode that the robot is in and the nature of the grid locations that it is traversing. In exploring traversal mode, the robot will increase speed across regions of well-known (demonstrated-traversed) grid locations; however, it will slow when preparing to enter an unknown region or approaching the target to be characterized. The effect of different speed settings was not tested; however, this is an area that learning techniques (which will be discussed in the following section) could possibly be used to increase system performance in future research.

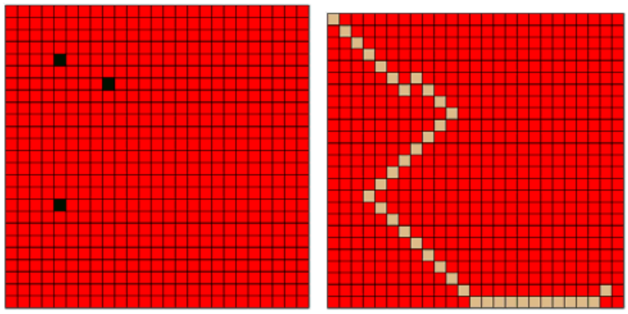

Fig. 4 & 5. Left: Partial grid of exploration area with initial -10 setting for all uncharacterized grid locations (red / light) and three characterized demonstrated-traversable locations (green / dark). Right: Partial grid of exploration area showing how high utility grid locations (as shown in Fig. 4.) impact routing

5.1 Path Planning

Path planning is utility and pheromone driven. The routing engine will make selections based on the lowest cost route that it perceives. Locations with known obstacles will have high negative utility values while those that have been demonstrated to be passable without issue on several occasions will have moderate positive utility values. A modified version of the A* path-finding algorithm is used. The route with the lowest total cost is selected.

6 LEARNING

The system relies on several constant values that are used to guide system operation. These values include the maximum target separation distance for accepting a task, the radius that determines when exploration is complete (when there is no region of this radius left unexplored, exploration is deemed complete) and the values that are used for positive and negative pheromone when regions are deemed impassible or well-traveled-enough to be demonstrated-traversable.

A learning algorithm based on testing incremental changes and evaluating their outcome has been developed and limited testing has been conducted. Minor changes to these values have been shown to have minimal effect; however, future research may focus on changing these values in conjunction. Additionally, correlation of particular value sets to preferential performance in various scenario types should be investigated.

7 EXPERIMENTAL SETUP

For testing, three robots were built which could be transformed into a variety of configurations. For characterization testing, the robots were set up with three different height levels for the upper bump sensor. For verification testing, two robots were assigned the same height level and the third was assigned a level that would only trigger characterization help requests (its upper sensor was shorter than the other group and also shorter than all obstacles).

A variety of obstacles were placed in the exploration area prior to each test; there location was measured so that the robotic results could be compared to the actual placement of obstacles. Obstacles were all heavy enough to avoid being moved by robot impact and no changes were made during each testing run.

The robots were placed in a central location facing in different directions. Once the test begun, the robots all moved off in their initial direction and begun random search activities.

Several scenarios were created which differed primarily in the density of obstacles present. The starting location of the robots was also varied as they were placed arbitrarily each time; however, the initial formation-placement of the robots was always very similar.

8 PERFORMANCE EVALUATION

The system was demonstrated to perform exceptionally well. In all test scenarios presented, the system identified all obstacles present in the exploration area. Additionally, this was done with virtually no manual involvement (in a few limited instances the robot dislodged one of its bump sensors requiring manual reattachment; however, this is a construction issue and not a control one). Despite having no mechanism for detecting side or rear impacts, the robots managed to successfully their exploration activities after rear collisions or side-swipes. Settings that allowed only short rearward movements for backing away from an obstacle worked as intended. The occasional non-bump-sensor-triggering collision (i.e., a collision with the front treads, etc.) or side swipe would temporarily cause a divergence of the robot's internal state believed-position and its actual position; however, this was generally resolved reasonably quickly by encountering an obstacle or wall and 'snapping' back to the grid based on this.

The small exploration area and high-traction carpet that it was covered with were undoubtedly factors in this performance. If the robot were more prone to slipping (or even could be spun by an impact) it is likely that re-snapping would have taken significantly longer or not occurred. The sensor configuration onboard the robots makes them have an exceptionally low tolerance for capture-and-place activities as they must physically explore a large area and encounter several known obstacles to reorient their internal position state.

9 CONCLUSION

The research conducted has demonstrated that limited communication and a set of well-defined behaviors are sufficient for characterizing an unknown environment. The robots in this experiment were purposefully simplistic in sensing capabilities and intentionally did not have any way to determine their true position (e.g., GPS). They demonstrated that with a reasonable level of relative movement knowledge, robots could assist each other in characterizing an environment that each could not sufficiently characterize on its own.

The control techniques demonstrated in this experiment are highly applicable to a variety of real world applications. Additional research, however, is required to validate their suitability and to identify required application-specific algorithm augmentations and constant values. A variety of general enhancements are also of research interest. These include adaptive learning (where constant values are updated based on performance in near-real-time by the evaluation system) and enhancements to deal with sensor data of varied levels of accuracy (i.e., visually sensed data versus bump-sensed data).

ACKNOLEDGEMENT

Thanks is given to the University of North Dakota Computer Science Department, especially Tom Stoke and Dr. Ronald Marsh, for providing the components that were used in this research.

REFERENCES

[1] Fink, W (2006), Generic Prioritization Framework for Target Selection and Instrument Usage for Reconnaissance Mission Autonomy, Proceedings of the International Joint Conference on Neural Networks, pp. 11116-11119.

[2] Fink, W, Dohm, JM, Tarbell, MA, Hare, TM, & Baker, VR (2005), Next-Generation Robotic Planetary Reconnaissance Missions: A Paradigm Shift, Planetary and Space Science, pp. 1419-1426.

[3] Fink, W, et. al. (2011), Robotic Test Bed for Autonomous Surface Exploration of Titan, Mars, and other Planetary Bodies, Proceedings of the IEEE Aerospace Conference, pp. 1-11.

[4] Tsatsoulis, C (2008), Sensor Webs as Multiagent, Negotiating Systems, *NASA Earth Science Technology Conference.* Adelphi: National Aeronautics and Space Administration.

[5] Straub, J (2011), A Modular, Application-Agnostic Distributed Control Framework for Robotic Applications, Accepted for publication in the Proceedings of the International Conference on Information and Communication Technologies and Appliations.