t-bots: a coordinated team of mobile units for searching and occupying a target area at unknown location

Tullio Facchinetti
University of Pavia, Italy
tullio.facchinetti@unipv.it

Abstract

This paper describes t-bots, a team of co-ordinated mobile units designed for the collaborative search of a target goal at unknown location within a workspace where obstacles limit the mobility of robots. The final goal of the team is to make all the team robots to reach the final target location and to enter a delimited target area. The strategy of each robot depends on the information that are gathered during the exploration. Such information can be acquired by sampling onboard sensors or can be received from other mobile units, thus exploiting the communication capabilities of robots.

A Finite State Machine (FSM) determines the mobility strategy, the co-ordination and operational modes. The FSM's state change is triggered by the information collected by the robots and by the information received from other robots using wireless communication devices.

1 Introduction

t-bots is the team of simulated robotic units developed at the Robotics Laboratory of the University of Pavia for participating to the simulated contest of CiberMouse'09.

The objective is to find an unknown target location, which is placed within a workspace where the robot's mobility is limited by obstacles. The final goal is to make all robots to reach such location and to enter a delimited area. The target location can be detected in two ways: a ground sensor detects if the robot is over the target area; a wireless beacon is sent from the center of the target area, and the direction (not the distance) to the beacon transmitter can be sensed when the robot is close enough to the transmitter.

Robots can base the workspace exploration upon information read from onboard sensors or received from other robots through the onboard wireless communication device. The set of sensors include: proximity sensors; beacon detection sensors; an absolute positioning sensor, which simulates a GPS receiver; ground sensors. *t-bots* do not make use of the onboard compass, since the current heading is retrieved from the GPS data; even though the compass gives more accurate measurements, the limitation on the number of sensors requestable for each sensor reading suggested to limit the number of exploited sensors to the minimum possible for achieving the desired operations.

t-bots do not implement complex features like mapping or recostruction of the environment. This allowed to focus on simple mobility and communication strategies that take into account the availability of multiple robots, the size of the arena, and the fact that each run of the simulation will not make use of information collected during previous runs. On the other hand, in other scenarios where collecting information about the workspace would be useful, a more advanced mapping would be required.

The main features of a *t-bot* are the following:

- the behavior is based on a Finite State Machine (FSM) that controls the operational modes of each *t-bot*;
- the mobility strategy is strongly based on location-awareness;
- location-awareness allows to implement the go-to-point strategy, which is the building block of most operational modes;
- the communication is used to broadcast very essential information useful to update the operational mode of each *t-bot*.

This paper will illustrate the architectural design of *t-bots*, including: the operational modes determined by the FSM, described in Section 2; the go-to-point behavior, in Section 3; the usage of communication in Section 4. Each section will describe the problems faced during the system design and the approaches adopted to solve them. Finally, Section 5 concludes the work by presenting some possible improvements.

Further details and the full source code of *t-bots* are available at [1].

2 The Finite State Machine and operational modes

A Finite State Machine (FSM) is used to keep track of the current operational mode of a robot. There are 4 operational modes:

1. QUEST: the robots move around the arena independently, on predefined paths, searching for the beacon transmitter;
2. BEACON FOUND: the robot found the beacon transmitter, or it has received such information from other robots;
3. GROUND FOUND: the robot has received the information about the exact location of the target area from other robots;
4. GROUND ACCESS: the robot is over the target area, waiting and performing suitable operations to allow all the robots to access the area.

When all robots have entered the target area, they all stops at the same time. Figure 1 shows the FSM and outlines the most common state transitions. It is worth to notice that the FSM is a bit more complicated, with many substates and including several error management conditions, but this sketch should help to understand the overall flow of the robot's behavior.

The strategy to find the target is as follows:

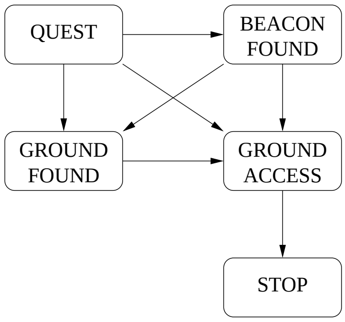

Figure 1. The Finite State Machine states.

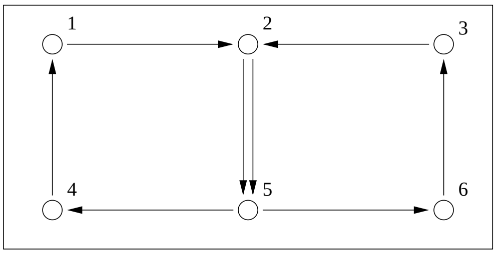

Figure 2. The path followed during the QUEST mode.

- at the beginning the 5 robots move around the arena, following a path made by a sequence of predefined target locations;
- this behavior gives the probabilistic chance to become in contact with the beacon signal;
- when one of the robots become in contact with the beacon transmitter, it starts broadcasting the current location and direction to the beacon, and goes towards a new location which is an estimation of the beacon trasmitter location, calculated from the robot's current location and moving into the direction to the transmitter;
- if a robot enters the target area, it starts broadcasting the exact location;
- when a robot receives whether the estimated or the exact location of the target area, it moves towards such location;
- when a robot enters the target area, it waits until all robots have also entered, to allow an entering strategy that allows all robots to enter.

The following sections describe each single operational mode.

2.1 QUEST operational mode

While in the QUEST mode, the *t-bot* moves according to a predefined path that allows to visit some key locations in the arena, searching for the beacon transmitter. The predefinded path is depicted in Figure 2. The sequence of visiting locations is constant, equal to

1–2–5–6–3–2–5–4–1

A different initial location is given to each *t-bot*, to achieve a sufficient initial spread of robots.

Each visiting point is reached using the go-to-point approach described in Section 3. When a location is visited, the next point becomes the new goal for the go-to-point routine. Clearly, the robot must avoid obstacles while moving towards the goal. Therefore, the trajectory is not necessarily straight between two locations. This is a desirable side-effect since, making the robot to go along a longer path, it could increase the probability to detect the beacon transmitter.

In this way the robot does not perform an exaustive search of the arena. Instead, it performs the probabilistic search of the beacon transmitter. This strategy is motivated by the following considerations:

- due to the arena's size, five robots can cover large part of the arena if they are sufficiently spreaded;
- the beacon transmission range is large with respect to the size of the arena's size;
- the radio transmission range is also large with respect to the arena's size, allowing a reasonable level of connectivity while travelling the arena.

For the above reasons, the fixed-point path works reasonably well in the given scenario, and allows to find the beacon reasonably soon without complex workspace exploration.

The probabilistic search stops when the robot determines the location of the target area. This can happen whether by a direct sensing of the beacon or the ground sensor. Moreover, such information can also be received from other robots. The state change is as follows:

- if the robot detects the beacon trasmitter or receives such information from an other robot, it goes into the BEACON FOUND mode;
- if it directly enters the target area, it switches into the GROUND ACCESS mode and starts broadcasting the co-ordinates of the target area;
- if it receives the information that some other robot is entered the target area, it goes into the GROUND FOUND mode.

2.2 BEACON FOUND operational mode

In the BEACON FOUND mode, the robot is aware of an estimation of the beacon transmitter location. This is possible due to a direct sensing of the beacon, or the reception of the information that an other robot detected the beacon.

In both cases, the available information are:

- the location where the beacon has been detected (x_{bcn}, y_{bcn});
- the direction towards the beacon transmitter d_{bcn}, which is the only information retrieved from the beacon sensor.

Using such information, the robot sets an intermediate goal equal to (x_{estim}, y_{estim}) as follows

$$x_{estim} = x_{bcn} + dist * \cos(d_{bcn})$$

$$y_{estim} = y_{bcn} + dist * \sin(d_{bcn})$$

The value of $dist$ is constant, and determined on the basis of the target area radius, the target area size, and the beacon trasmitter range.

The robot then uses the go-to-point algorithm to reach the intermediate goal. During the travel, it is likely to receive more accurate information (the exact target area position), or to pass over the target area. In the former case, the rotot switches to he GROUND FOUND mode, while in the latter it switched to the GROUND ACCESS mode.

The information about the beacon trasmitter location can be either directly obtained from the robot that detected the beacon or through a multi-hop communication from other robots.

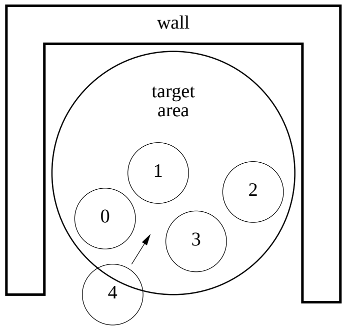

Figure 3. Example of situation where a coordinated entering strategy is required.

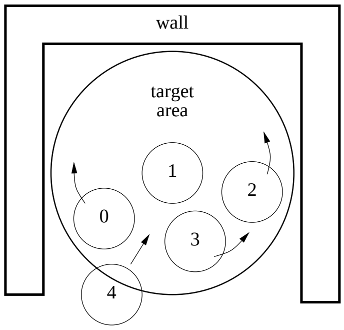

Figure 4. The strategy to “make space” for the 5-th robot.

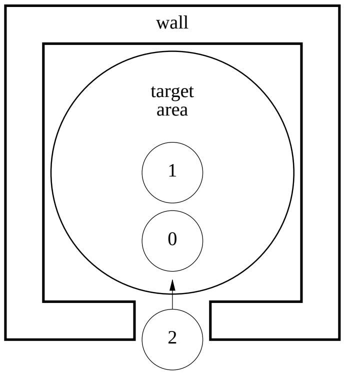

Figure 5. A worst case scenario that is not yet covered by the current implementation of entering strategy.

2.3 GROUND FOUND operational mode

The GROUND FOUND mode is very similar to the BEACON FOUND mode, in the sense that the robot sets a goal point to be reached using the go-to-point algorithm. However, while in the BEACON FOUND mode the robot has only an estimantion of the goal point, in this mode the robot has the information about the exact location of the goal. This mode can only entered upon the reception of information from other robots, since when a robot actually enters the target area, it switches to the GROUND ACCESS mode.

Even in this mode, the information about the target area location can be obtained directly from the robot that entered the target area or through a multi-hop communication from other robots.

2.4 GROUND ACCESS operational mode

There is a dedicated operational mode for accessig the target area when only one robot remains outside. This specific mode has been introduced for two reasons:

1. given the scoring rules, where 100 points are substracted from a robot’s score if it reaches and enters correctly the target area, it is far better to dedicate extra time to achieve that all robots will enter the target area;
2. we noticed that up to 4 robots can enter the target area without any problem just using the QUEST, BEACON FOUND and GROUND FOUND modes. However, the 5-th robot may experience several problems in entering the target area without a dedicated strategy.

For example, in the situation depicted in Figure 3, robot 4 can no longer enter the target area if all the robots simply terminate their operations as soon as they enter the target area (to save time and achieve a better score). In this case, the score of the 4 robots within the target area will be better with respect to the case in which they wait for the last robot, but the total score could be far worst since node 4 will not benefit of the -100 points *bonus*. This situation considers a very pessimistic workplace, but the team should be able to face all the conditions that are likely to happen.

For the above reasons, a dedicated strategy is targeted to let the last robot enter the area.

First of all, all robots remain activ even if they already entered the target area. They only terminate their execution when all robots have entered. This condition is detected by inspecting the information received from other robots.

The first robot that enters the area becomes the leader, and obtains the right to reach exactly (with appreciable precision) the center of the target area. This action is targeted to leave space to the next robots for entering the area.

When the last robot tries to enter the target area, if there is an obstacle made by one or more units, the robots within the target area performs the proper actions to “make space” for it. Those robots start rotating around the area’s center until there is space enough for the robot to enter.

The robots exchange their angular displacement in their messages. They use such information to infer if the rotatation around the area’s center should be clockwise or counterclockwise, depending on their angular displacements relative to the one of the last robot.

For this purpose, the angular displacement with respect to the beacon transmitter must be calculated very accurately, so a mobile average filtering is done on the absolute orientation sampled from the GPS and on the relative orientation of the robot with respect to the beacon trasmitter. This operational mode implements all the timing behaviors to let the orientations to remain steady during the filtering.

The described strategy does not face the very worst scenario where only a narrow passage is available for entering the target area (Figure 5), since the “making space” strategy is only applied to the last node which has not entered the target area. Thus, in the situation depicted in Figure 5, the second node that will enter the target area will block any remaining robots. However, the proposed method could be extended to cover such situation, by applying the entering strategy not only to the last robot but to all robots after the first two units have entered the target area.

3 Go-to-point behavior

The go-to-point behavior is the main strategy of a *t-bot*. Given a target location within the arena, the robot tries to reach such location by applying a motion strategy which operates as the Tangent Bug described in [2]. The Tangent Bug uses only local information to reach a global goal (the target location), and can deal with finite range sensing devices. As other Bug algorithms, has the nice feature that, for 2-dimensional workspaces, it always finds the path to the goal if it exists.

The technique is based on two sub-strategies: motion-to-goal and boundary-following. The following sections briefly describe the go-to-point algorithm, focusing on the techniques adopted for their implementation.

3.1 Motion-to-goal

During the motion-to-goal phase, the robot drives towards the current goal. The motion is not necessarily on the straight line between the current robot position and the goal, since proximity sensors are used to move around obstacles. However, the robot remains in the motion-to-goal phase until the distance between the current position and the goal is decreasing.

When the distance from the goal increases, it means that the robot can no longer move towards the goal, due to the presence of obstacles. In this case, the boundary-following phase in entered. The robot remembers the location where the boundary-following phase is entered as x_{bf}.

One problem experienced during the implementation of the motion-to-goal phase arises from the evaluation of the trend of the distance variation. Due to the errors of the GPS sampling, it is infeasible to check the distance variation by simply comparing the distances calculated at two subsequent sampling instants. For example, even though the true distance is decreasing, measurement errors of the robot position may lead to the wrong conclusion that the distance is increasing, causing the robot to wrongly switch to the boundary-following phase, thus jeopardizing the whole go-to-point algorithm.

To overcome the above descripted problem and thus to correctly evaluate the distance variation trend, each robot stores the history of the distances from when the motion-to-goal phase has been started. It uses such information to calculate a linear regression of the latest n values, and it concludes that the direction is decreasing if the linear regression has a negative slope; otherwise the distance is increasing. Larger values of n results in higher filtering of the measuring errors, but introduce a higher latency in determining the correct distance variation trend.

3.2 Boundary-following

The boundary-following phase is needed to exit from situations where obstacles impose to the robot to get farther from the goal in order to avoid the obstacle itself. The robot follows the obstacle boundary until it reaches a location x_{closer} that is closer to the goal with respect to x_{bf}. In this case, it returns to the motion-to-goal phase. If the robot returns to x_{bf} without having found any x_{closer} location, it means that no path to goal exists. In this latter case, the robot switches to an operational mode that depends from the current operational mode.

4 Communication

The robots make a very simple usage of the communication, broadcasting a limited number of information. This is done to avoid overloads on the trasnmission medium, which limits to 100 the amount of characters that could be sent at every simulation instant.

The values sent at each trasmission are the following:

- the FSM state and a *bit mask*, that are used together by the receiver to infer the current behavior of the sender;
- the location of the sender;
- the values of (x_{bcn}, y_{bcn}) and d_{bcn}, which are used during the BEACON FOUND mode;
- the co-ordinates of a node when it is over the ground area; such co-ordinates is used to forward the information in a multi-hop fashion from the node which entered the target area to all the remaining nodes (exploited in the GROUND FOUND mode);
- the current direction of the robot towards the beacon trasmitter, which is used only during the GROUND ACCESS mode;
- the ground flag, which is used to determine if all the robots are over the target ground, and so it is possible to stop the team's activity.

The *bit mask* contains a set of flags that are used to perform agreements on some common tasks (i.e., during the GROUND ACCESS), and are managed as described in [3], even though the real-time evaluation performed in the paper is not adopted here. The most meaningful flags are:

- a set of bits to indetify whose robots are aware of the beacon detection;
- the same for the ground detection;
- a bit indicating whether the leader has been elected;
- a bit that indicates that the robot is the leader.

The mask is used to let the transmission of all the information even when they are not meaningful (i.e., the ground location is sent even when it has not yet been discovered), and allow the receiver to consider the information only when they are meaningful. This simplifies the contruction and the parsing of the message. It is worth to note that more complex communication schemes could be adopted, but have not been investigated in this work.

The direction of the robot is used only during the GROUND ACCESS mode. It contains the filtered absolute angular displacement of the robot with respect to the beacon transmitter. This value makes sense only when the robot is not rotating: since the filtering is needed to achieve an accuracy in the order of ten-th degree, it can not be achieved if the robot is rotating. The accurate value is needed by the "making space" procedure, which requires an accurate estimation of the relative angular displacements among the robots.

5 Conclusions

This paper presented an overview of *t-bots*, a team of coordinated robotic units dedicated to the seek and reach a target location. The paper focuses on the approach and the solutions adopted to solve the problems encountered during the development of the co-ordinated strategy.

References

[1] http://robot.unipv.it/toolleeo/contrib/t-bots.

[2] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. The MIT Press, 2005.

[3] T. Facchinetti, L. Almeida, G. Buttazzo, and C. Marchini. Real-time resource reservation protocol for wireless mobile ad hoc networks. In *Proceedings of the IEEE Real-Time Systems Symposium*, December 2004.