Framework for the Long-Term Operation of a Mobile Robot via the Internet

Shervin Emami emami@itee.uq.edu.au
Gordon Wyeth wyeth@itee.uq.edu.au
Michael J. Milford milford@itee.uq.edu.au
David P. Prasser prasserd@itee.uq.edu.au

School of ITEE
University of Queensland
St Lucia QLD 4072 Australia

Abstract

This paper describes an autonomous docking system and web interface that allows long-term unaided use of a sophisticated robot by untrained web users. These systems have been applied to the biologically inspired RatSLAM system as a foundation for testing both its long-term stability and its practicality. While docking and web interface systems already exist, this system allows for a significantly larger margin of error in docking accuracy due to the mechanical design, thereby increasing robustness against navigational errors. Also a standard vision sensor is used for both long-range and short-range docking, compared to the many systems that require both omni-directional cameras and high-resolution Laser range finders for navigation. The web interface has been designed to accommodate the significant delays experienced on the Internet, and to facilitate the non-Cartesian operation of the RatSLAM system.

1 Introduction

To navigate effectively in any environment, a mobile robot needs both a map of that environment as well as the ability to position itself within that map. Furthermore, if the environment is initially unknown, the robot must explore and map out the environment before it can perform useful navigation tasks. This can only be achieved through a continuous mapping and localisation process known as Simultaneous Localisation and Mapping (SLAM). SLAM is considered one of the most significant problems that must be solved before mobile robots become truly autonomous. A large number of solutions to the problem have been developed by various researchers. RatSLAM is a biologically inspired SLAM system comprising of a number of neural networks that are based on models of the rodent Hippocampus [Milford, 2005]. RatSLAM can explore and map large-scale indoor and outdoor environments, and navigate to goals. However, the system has not yet been used in experiments lasting longer than several hours. It is unclear whether the RatSLAM methodology contains any long-term stability issues or learning problems. There are two main practical considerations stopping longer tests from being run; the battery life of the robot and the lack of a simple and easily accessible user interface.

This report describes the development and implementation of an automated battery charging system and a web interface module designed to solve these problems and provide a test bed for long-term experimentation using the RatSLAM system.

Robots that operate continuously are already used in the real world - in automated factories, they operate continuously for 24 hours a day. In contrast, mobile robots are typically only capable of short-term operation. Virtually all mobile robots require both software and hardware to be shut-down before its batteries can be manually recharged. To overcome that problem, this work has focussed on designing a docking station and associated software to allow a mobile robot to recharge its own batteries whenever required, before continuing its normal operations with minimal interruption.

The docking station system was designed for a Pioneer 2DX robot running RatSLAM during long-term indoor experiments. The assumed sensory input for this design was a robot-mounted camera, a range detection device such as a sonar ring or Laser range finder, and wheel motor encoders. These sensors are found on virtually all mobile robots however, so the system could easily be adapted to suit the majority of indoor robots. Furthermore, the docking station has been designed to be light and portable so that it is easily repositionable, allowing it to be placed in arbitrary positions and even moved during experiments.

Autonomous docking stations are not a new concept – there is a range of designs in existence, but they often require a special omni-directional (360˚ view) camera as well as a digital compass. The docking station discussed in this report is proposed as an alternative to the few systems that do operate based on vision alone. To improve robustness against failure as well as applicability for different types of robots, docking operations were designed to operate independently of the RatSLAM system once the robot had navigated to the local region containing the docking station. This allows reliable docking despite minor errors in the RatSLAM system. The system was also designed with angular and positional robustness in mind, so that the docking operation would be reliable despite sensor noise.

To promote the development and testing of the RatSLAM system's long-term capability, a web-interface is being created to allow the public to view and manipulate a robot using RatSLAM from within their web browsers. This will be the first public test of the RatSLAM system and will show whether it is viable for the control of mobile robots. Creators of existing web-based mobile robots often feel the most significant limitation of these robots is their limited battery life and that 24-hour operation would be of great benefit [Simmons *et al.*, 1999]. This would be overcome by the combined docking system and web interface.

RatSLAM's non-Cartesian representation of the world has prompted a web interface that requires much more than a simple broadcasting camera and manual controls. The user indicates a target location to which the robot will navigate by itself using RatSLAM. The act of navigating to a goal provides both a means to demonstrate and validate RatSLAM's representation of the world and a method of robot control that is robust to the varying delays of network communications. The typical delays experienced with the world-wide-web limit continuous control of mobile robots, as there may be periods where feedback from the robot is unavailable. Since goal navigation does not require user intervention, the robot can continue regardless of the speed or reliability of the network.

2 Background

2.1 Docking System Background

Currently very few mobile robots have the capability to recharge without manual intervention. These systems are still under development, and while specific implementations vary, they tend to use the same set of features.

Requirements for Automated Battery Charging:

1) The robot must detect when the batteries need recharging. A threshold may be applied to the battery voltage, or the operating time can be limited. Another technique is to accurately measure the amount of electrical current being drawn. By tracking this, a robot can predict how long the batteries should last before they lose energy [Hada and Yuta, 1999].

2) The robot must be placed relatively close to the docking station. Different navigation methods include SLAM, wall or line following, or beacon searching, each with their own limitations.

3) Control of the robot must be transferred to the docking system before it can move. Either motor control is disabled from the regular navigation system, or it is shut down and restarted when docking is complete [Silverman et al., 2003].

4) The robot needs a method of locating the docking station markers. Current systems often perform marker detection with either a Laser range finder [Silverman *et al.*, 2002; Oh *et al.*, 2000] or a vision sensor. They are based on either appearance (eg: Optical Flow Analysis [Barnes and Liu, 2001] or Image Warping [Franz *et al.*, 1998]) or feature extraction (eg: Active Markers [Cassinis *et al.*, 2005], or Laser barcodes [Oh *et al.*, 2000]).

5) The robot needs a navigational behaviour to get from its current position to the docking station. This should allow the integration of obstacle avoidance. Docking behaviours may be distinctly classified as requiring absolute calculations of pose (Metric approaches) or based on relative changes to sensor input (Reactive approaches). Metric approaches tend to be variations of the Potential Fields method [Arkin and Murphy, 1990; Rizzi *et al.*, 1998] and require feature extraction, while Reactive approaches are generally region tracking [Mitchell and Labrosse, 2004; Cassinis *et al.*, 2005 unpublished] or using Optical Flow [Santos-Victor and Sandini, 1994] and may be based on appearance or features.

6) A docking station with an automated method of electrically connecting & disconnecting with the robot. This often involves a protruding male connector on the robot and a female connector on the docking station. This connection must allow electrical power to be transferred to the robot, hence requires a minimum of two electrical connections. A crucial feature of the connecting system is that a certain level of misalignment must be acceptable, based on the accuracy and reliability of the guidance system. Current docking systems typically allow ±5° and ±5cm of acceptable error.

7) An automatic battery charger must be placed on either the robot or the docking station. The charger is usually part of the docking station, as this reduces the weight and size of the robot, and of course, lethal voltages will not be required on the docking connector. However if the robot contains an onboard charger, standard mains electrical connectors can be used, such as the standard wall plug [Yamada *et al.*, 2005] or an IEC "Kettle" plug [Austin and Kouzoubov, 2002], both of which are typically found in many indoor environments.

8) The robot requires a method of detecting the establishment of a connection. Many systems use an infrared sender and receiver to determine when the robot has physically connected to the docking station [Hada and Yuta, 1999], and assume that electrical connectivity is established upon physical connectivity. A detection system is required because there may be occasions when the robot mistakenly believes it has docked.

9) A truly robust system must also integrate obstacle avoidance and error handling (eg. misalignment or failing to detect a marker) through the whole process to allow for unexpected events that will invariably occur in real situations.

The most crucial aspect of the docking system is marker detection (stage 4), for two reasons:

Firstly, the features of the marker detection system (such as range, accuracy, and reliability) and the forms of outputs it provides (whether it obtains absolute or relative bearing, distance, position, orientation, or physical measurements) will determine the possible docking behaviours (stage 5) that may be used.

Secondly, marker detection systems are sensitive to image noise, lighting conditions and hardware properties, with many errors in the marker detection system typically being reproduced or even amplified by the navigation and docking strategies.

The marker detection system for a docking system requires both long range ability and short-range accuracy since it is required to accurately dock (typically with centimetre accuracy) from several metres away. Virtually all sensory devices are only reliable in either long-range detection or short-range accuracy; therefore most systems use two different sensors and methods during the docking process (eg: [Silverman *et al.*, 2002]), performing a two stage process of approximate docking and high-precision docking. Approximate docking typically involves Laser or sonar range finders, vision processing [Silverman *et al.*, 2002], infrared beacons [Oh *et al*., 2000], a visible light source [Walter, 1953], or line following. High-precision docking typically involves a Laser range finder [Silverman *et al.*, 2002; Oh *et al*., 2000], reflective tape [Hada and Yuta, 1999], vision processing or ultrasonics [Arkin and McKenzie, 1994].

Due to the requirement of virtually every docking station for the robot to enter along a certain axis, an important role of the docking behaviour is that it must not only bring the robot to the docking station, but it must also make sure the robot enters the docking station from a certain direction and orientation. In fact, majority of the current docking stations require the robot to enter from within ±5° and ±5cm of the docking station's axis, some of them even requiring millimetre accuracy [Austin and Kouzoubov, 2002]. The alternative method of battery recharging is to use a docking station that allows entrance from any direction, thereby allowing a simpler docking strategy. This is essentially the way that electric bumper or "dodgem" cars obtain their power – by using a metal plate both on the ground and above the robot, and the robot uses brush contacts to transfer power across the flat surface [To and Mann, 1998 unpublished].

There are also alternative approaches to a static docking station for mobile robots. Zebrowski and Vaughan [2005] have simulated a novel recharging solution by using a mobile docking station (a large 'tanker' robot that is capable of powering other robots) that will search for robots with low battery power and dock with them, as well as its own battery charger. This is similar to a Marsupial system [Minten *et al.*, 2001] where the Daughter robot searches for and returns inside the large Mother robot.

Industrial robots currently performing automated docking use simple but reliable techniques to find the docking station, such as following embedded electrical wires in the factory floor [Schilling et al., 1993], and the robot is provided with a map of the environment while using dead-reckoning (high precision motor wheel encoders) to return to a fixed position. However these systems require a fixed environment, and often don't allow for obstacles or simple changes in the environment. Hence, these systems are not applicable to the mobile robots under discussion.

2.2 Web Control Background

The simplest form of tele-operation is direct closed-loop control of the distant robot by the user, such as that used in hazardous environments. However if delays are long or unpredictable, the system can become unstable. A better approach is supervised control, where the user guides the robot from a distance but the robot has local control such as obstacle avoidance [Sheridan, 1992].

Robots have been tele-operated over distances for more than 50 years [Goertz and Thompson, 1954]. These systems all used dedicated communication channels between the robot and the operator, thereby relying on a continuous and guaranteed delay and transfer rate. However all Internet based telerobots operate within an environment with a large and unpredictable variation in both delay and transfer rate [Oboe and Fiorini, 1988], as well as unpredictable data ordering and data loss if using a simple protocol such as UDP instead of TCP.

This places new requirements on web-based robots compared to traditional telerobots, as manual control of a mobile robot can become unstable due to variable Internet delays [Hirukawa and Hara, 1998]. Additionally, users were trained scientists and engineers whereas typical users of a web robot have minimal technical skill and don't wish to read complex instructions [Schulz et al., 2000]. Typical web users are also very temporal, expecting results within their web browser (rather than stand-alone programs) and interactivity within three minutes [Taylor and Dalton, 2000].

3 Web Interface

A web interface has been created to allow remote operation of the robot via the Internet. The web interface is currently functional, however the user interface will be modified to incorporate a new version of RatSLAM. This may require a major redesign since the RatSLAM system is partially non-Cartesian. While a simple menu could be implemented allowing the user to click on buttons for various room locations, this will be less exciting for the web user compared to having a 2D map that allows moving to any desired location.

3.1 User Interface Issues

Clearly the user interface for an indoor web robot could have a map of the floor showing all the obstacles and walls, and allow the user to simply click on a location in the building to place a goal, telling the robot that it should try to get to that location. However RatSLAM's non-Cartesian model (based on the Hippocampus) is partially topological rather than Cartesian [Milford, 2005]. Effectively this means that a standard overhead map of obstacles and walls may be unusable for the navigation, as two adjacent points in the map may not necessarily be representing two adjacent points in the environment. Therefore a custom user interface will be needed that provides the user with sufficient control and flexibility but without any technical experience necessary. The goal seeking navigation of the robot is still under development, hence a final user interface has not been decided on, as this will depend on the capabilities of the navigation system. A temporary interface is currently in-place to allow testing of the system.

The main impact of a non-Cartesian map is that the environment (the walls and obstacles) may look extremely strange and incomprehensible to the untrained eye, therefore the map is likely to contain various markers to show where physical locations are on the map, exploiting the topological nature of the navigation system (Figure 1).

To allow a graphical user interface that supports topological navigation but with adequate functionality and user appeal, each web client will automatically download a Java Applet, capable of communicating with the central server, displaying the graphical interface and processing user input.

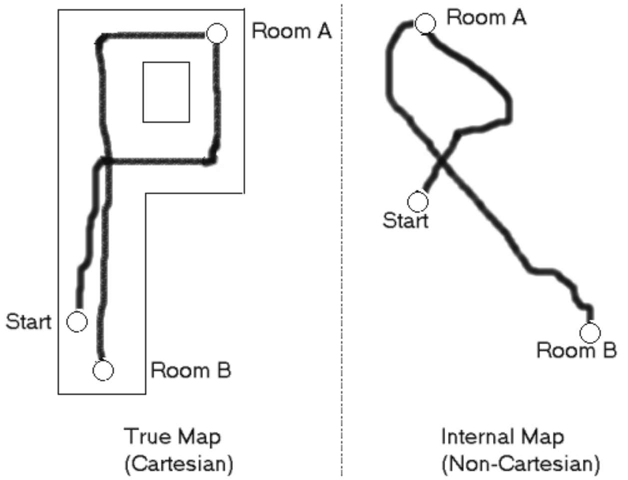


Figure 1: In this example, the starting position and Room B are in fact next to each other, yet the internal non-Cartesian map may treat them as dissimilar. The reverse condition may also occur, therefore the map of the environment used for this web interface cannot show (user friendly) walls.

3.2 Communication

The architecture of the RatSLAM system performs the majority of the processing on a remote PC through a wireless network to the robot's onboard PC (see Figure 2). An intermediary program is executed on the departmental web server (using Unix Solaris 8) which allows reliable communication between the RatSLAM system and the world-wide-web. This software (named 'Relay' since its main purpose is to relay information between the robot and the web users) must provide access of the various resources of the robot to multiple simultaneous web clients, without causing instability to the RatSLAM system or itself whenever a web client crashes or sends corrupt data. Therefore it has the most vital and complex role of the web interface system.

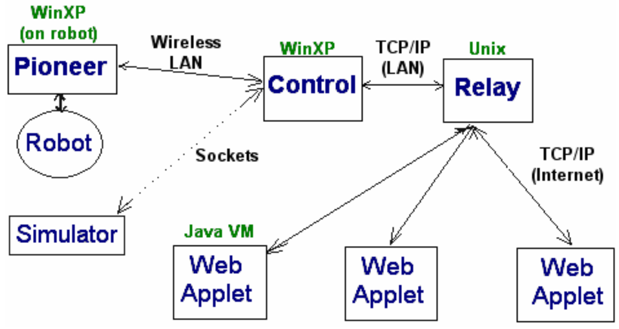
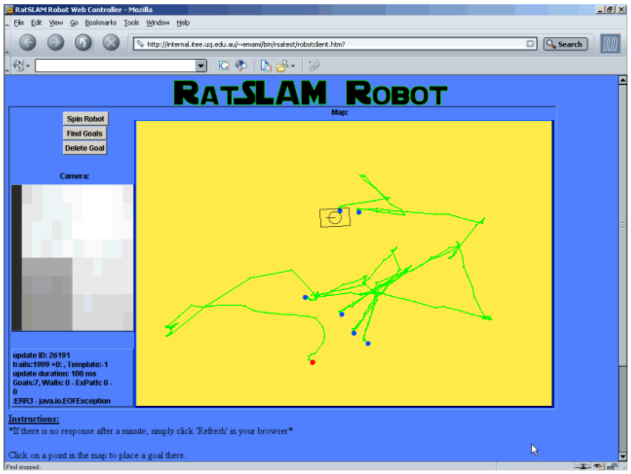


Figure 2: Communication between various software modules. "Pioneer" performs low-level processing, "Control" performs RatSLAM, obstacle avoidance and communication with the Web Interface. "Relay" synchronises multiple web clients for access to the robot.

3.3 Explored Path

To show users where the robot has recently navigated, the environment map shows an exploration trail over the past 30 minutes (see Figure 3). To keep the network data to a minimum, the trail data is sent as a list of (x,y) coordinates to be displayed by the Java applet. Rather than sending thousands of coordinates (upto 16kB of data) to each web client for each update, an efficient system has been implemented that keeps the Relay web server updated with the latest explored path but only sends each web client any exploration data that it is missing. For example, if the navigation system is obtaining data every 100ms, assuming there is one web client with a 50ms refresh rate and another with a 500ms refresh rate, then every 100ms the web server will be sent one new robot position, the fast web client will be sent a new position just once every 100ms, and the slow web client will receive the past 5 robot trail positions with each update.



update ID: 26191
trails:1999 +0:, Template:-1
update duration: 100 ms
Goals:7, Walls: 0 - ExPath: 0 -
0
:ERR3 - java.io.EOFException

Instructions:
If there is no response after a minute, simply click 'Refresh' in your browser

Click on a point in the map to place a goal there.

Find stopped.

Figure 3: Typical screenshot of the current web interface. Notice that there is no environment or floor plan due to the non-Cartesian navigation system.

3.4 Robustness

To allow robust behaviour under the many different potential errors and failures that should be expected with unregulated web users around the world, the software contains a vast level of data verification, enabling a web client to suddenly crash or send corrupt data without compromising the stability of the robot and other web users. This is a very complex task, involving various timeouts, data integrity checks and connection verifications for each send or receive operation, and only field tests can show the true robustness of the system.

Although various levels of error checking have been implemented on each of the 4 places of communication (Control's server, Relay's client, Relay's server, and the Java Applet's client), it has to be assumed that any of the software modules may still crash for unknown reasons. Therefore, Control's server will automatically shutdown and restart whenever connection to Relay is lost (such as when Relay has crashed), and Relay will automatically reconnect to Control's new server if it has been restarted. This is in the hope that Relay can be run as a system service just once on an Internet server, and even when the robot is turned off or Control is shutdown due to software changes, Relay will continue running and will reconnect once the robot's Control system has restarted. On occasions when any of the 4 communication modules are restarted, the web user will simply need to click on their web browser's "Refresh" button to continue operation.

4 Implemented Docking System

The docking system created is based on feature extraction using an existing colour segmentation library developed by Prasser and Wyeth [2003], and performs real-time processing via the onboard AMD K6 400MHz processor. The library has been significantly modified for the purpose of docking (requiring markers to be at eye-level with the camera, using flat markers instead of cylinders, searching for multiple markers, and supporting partially obscured markers), and can obtain the approximate position and direction (pose) of the robot relative to a docking station.

The two landmarks used are thin 20x10cm markers placed 60cm apart, where each marker is split into two colours (see Figure 7). During colour segmentation, results are filtered to find markers where both colours are of similar size and in alignment with each other. Sporadic noise is reduced by a mean filtering of the results over five frames, and false classifications are reduced by using the vertical position of the marker as a heuristic when matching landmarks, since they should be at approximate eye level with the camera.

Since the calculated pose has a higher susceptibilty to noisy data compared to the original distance and bearing measurements from the landmarks, the docking behaviour used is a combination of Metric and Reactive approaches (see Item 5 of Section 2.1), where a Ballistic movement is performed from the robot's current position to a target point 1.5 metres in front of the docking station. This is followed by a Controlled movement from the target point directly into the docking station (see Figure 4). During the Ballistic stage the robot is typically moving in a perpendicular direction to the docking station, and is unable to see the docking station. Whereas the Controlled stage is a closed-loop movement since the robot will be moving towards the markers. This is based on biological motion (such as human movement) which can be characterised as two fundamental types: Ballistic and Controlled [Woodworth, 1899]. Woodworth found that almost all voluntary movements were characterised by these two behaviours, from large-scale to small-scale tasks. Further research on Cebus monkeys have shown that the calculations performed in the Ballistic stage are too fast to be made in real-time, hence are a pre-calculated open loop task [Brooks *et al.*, 1973].

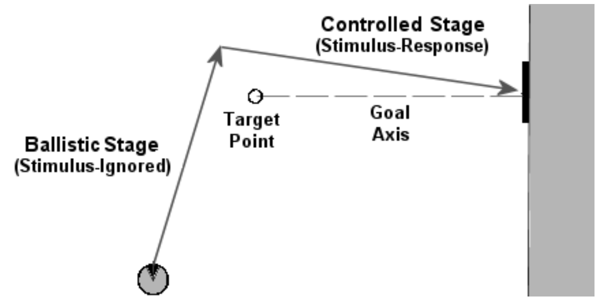
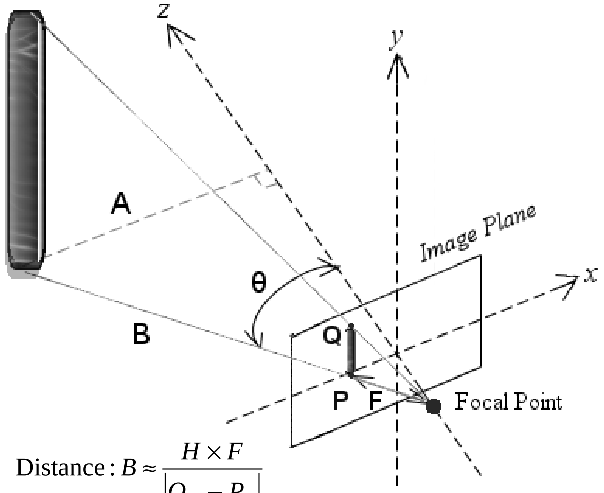


Figure 4: The two-stage docking behaviour, based on biological movement.

4.1 Pose Estimation

Distance and bearing to each landmark is calculated based on the most significant feature. This results in a distance calculation based on the known height of the tall thin landmark in most situations (see Figure 5), but using the width of the short fat landmark when the robot is significantly close to the landmark (where part of the landmark will be out of the camera's field of view). Bearing is calculated based on horizontal position alone.



$$\text{Distance}: B \approx \frac{H \times F}{\left|Q_y - P_y\right|}$$

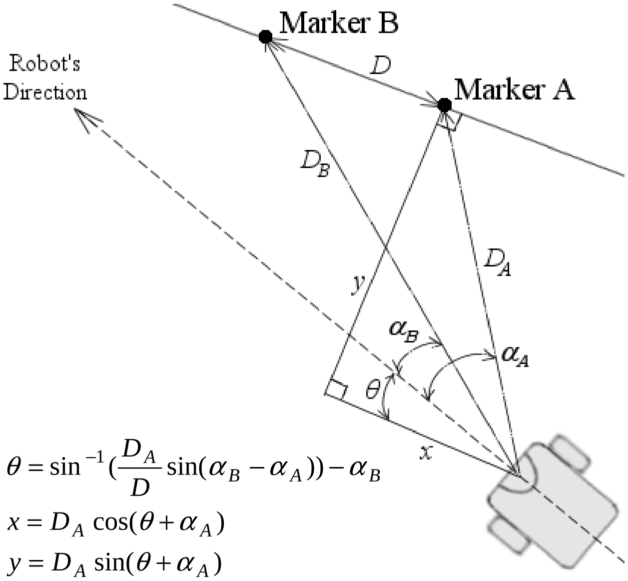
$$\text{Bearing}: \theta \approx \sin^{-1}\left(\frac{P_x}{F}\right), \text{where H and F are known.}$$

Figure 5: Distance and bearing from a single landmark.

Since the docking station requires the robot to enter from a certain position and direction, a coordinate frame is used to allow the robot to enter only through the goal axis. Once distance and bearing is known for two separate landmarks that are placed a known distance apart, pose *(x,y,θ)* of the robot relative to the goal reference frame may be extracted, therefore the goal axis will be known (see Figure 6). This process is highly vulnerable to noise, however it is only used for the Ballistic stage of docking; the following Controlled stage overcomes the large errors in the Ballistic stage.

Rather than using different sensors and markers for the Ballistic and Controlled docking stages, a single sensor and set of landmarks are being used, but with different navigation strategies. This greatly simplifies the hardware and setup required for the robot docking system. To support long-range approximate docking, the two markers are large enough for detection from a distance of 5m and adequately spaced to reduce noise in the calculation of pose. However when the robot is near the docking station, only one of the markers will be within the camera's field of view. This is typically the stage where other docking systems would switch to the use of a Laser or other high-precision sensor for the final docking stage.

To allow using a single sensor for this docking system, one of the markers is placed directly inline with the desired docking position, allowing a visual servoing process to coordinate the Controlled stage of docking with just the single marker based on distance and bearing (see Figure 7). Both of the markers are used for distance & bearing until approximately 1m from the docking station when a single marker is used for docking, to a distance of roughly 30cm when the marker is only partially within the field of view. In this close proximity where the top and bottom of the marker are out of view, distance is calculated based on the width of the marker rather than its height. The width of the marker is visible up to a distance of approximately 10cm, when the left and right sides of the marker are also obscured, hence no geometry is extractable. By this stage the robot will have already docked, since the camera is mounted in the centre of the robot.



$$\theta = \sin^{-1}\left(\frac{D_A}{D}\sin(\alpha_B - \alpha_A)\right) - \alpha_B$$
$$x = D_A\cos(\theta + \alpha_A)$$
$$y = D_A\sin(\theta + \alpha_A)$$

Figure 6: Position and heading (x,y,θ) relative to the goal axis is obtained solely from two landmarks. Noisy feature extraction can affect this significantly and so this is only used for the initial Ballistic stage, while distance and bearing to single landmarks are used as input for the remaining operations as this has greater stability.

Figure 7: The green and red marker (on the right) is directly above the target position of the docking station.

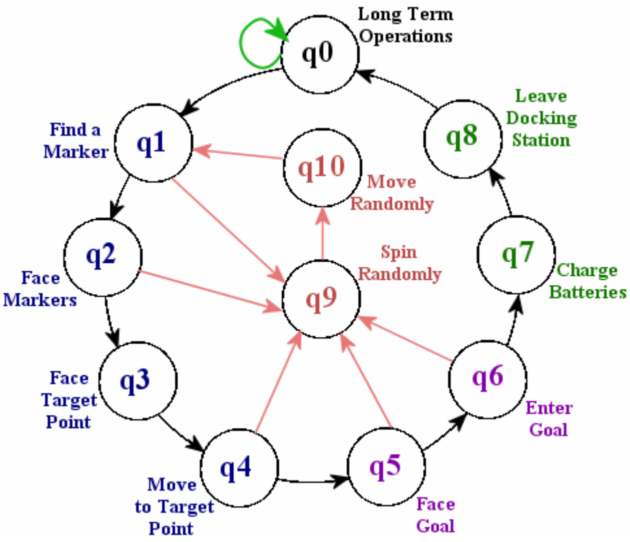
4.2 Stages in the Docking Process

The docking behaviour has been implemented as a Finite State Automaton, where the robot moves in a sequence of discrete states to successfully transition from having low batteries to having recharged batteries (see Figure 8). Since the docking operation will be called upon by a higher-level navigation system when batteries are low, the docking system assumes that the robot has been placed in the general area of a docking station before commencing.

There will often be occasions when the robot will perform a scan for the docking station but not detect the landmarks, either because they are not visible from that location or because the marker detection system behaved incorrectly. Incorrect marker detection is generally due to noisy image segmentation since the robot is moving while it is performing the image capture and segmentation. This can be reduced by reducing the robot's speed, or by using an omni-directional image sensor.

To allow for circumstances when the docking station will be hidden by obstacles (or even in the opposite side of the room), the robot will search for the docking station by moving in random directions, performing a 360° scan at each new location. As the robot is used in indoor, closed environments, this procedure is bound to find the docking station, albeit slowly. Since the docking station may be obscured by both temporary obstacles and large fixtures such as walls or desks, this process of searching in random locations is used whenever an obstacle is detected or the docking station hasn't been found.

The docking behaviour combines both Metric and Reactive approaches, in that states alternate between performing a (metric) operation based on motor encoders or performing a closed-loop (reactive) operation where previous misalignments and drifting errors are overcome.



Ballistic stage: {q1, q2, q3, q4}
Controlled stage: {q5, q6}
Error handling: {q9, q10}

Figure 8: State Machine of the docking process. The robot typically performs long-term operations (RatSLAM, Web Interface, etc) until its batteries are low, when it leaves state q0 to perform the recharging process.

Error handling (moving to a random location) is entered when the robot has either rotated or moved too far for the state it is in (based on the motor encoders), spent too long trying to move (based on time in that state) or has encountered an obstacle (based on the Laser and Sonar range finders).

4.3 Physical Connection System

The connecting system used is an extremely simple design that allows for very large misalignments in both position and orientation. Many docking systems require an accuracy of less than ±5cm and ±5° (some systems requiring millimetre accuracy [Austin and Kouzoubov, 2002]). Requiring such high precision reduces the reliability of the system as small errors are likely to result in an unsuccessful docking. Therefore a docking station has been created that allows a significantly large error margin of ±20cm and ±55° (see Figure 9). A successful electrical connection is far more likely than with a system that only allows a very small margin of error.

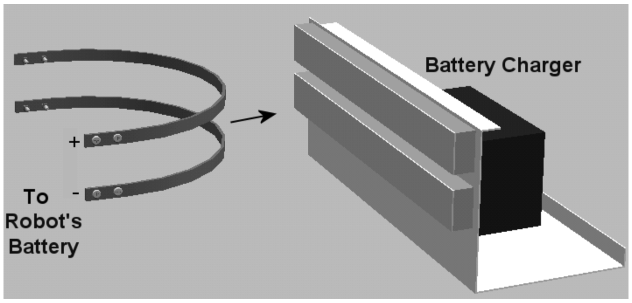


Figure 9: **Left:** The two metal rails attached to the robot, connected through a protection diode to its battery. **Right:** The two flat metal plates held up by soft sponge material, connected directly to the battery charger. This very simple design has an extremely high tolerance of error in both positional and angular misalignment.

Figure 10: Photo of the rails mounted on the robot, entering the docking station.

4.4 Limitations

Image segmentation is used to find distance and bearing for each marker, which is then used to calculate the position and orientation of the robot. Any error in this initial image segmentation may be further amplified by the pose calculation, resulting in potentially large positional errors. This is further exacerbated by the Ballistic stage since it involves a large open-loop movement based on the robot's calculated pose. However, the docking system behaviour uses lower-level sensor measurements whenever possible (alignment based on landmark bearing rather than the calculated pose) to reduce the effects of segmentation noise. The final stages of docking are all based on low-level sensor-feedback loops to remove the inaccuracies of the Ballistic stage, therefore allowing the system to overcome a significant level of noise in the majority of cases. In situations when a stage in the docking process fails due to misalignments, noise or obstacles, the docking process is repeated from a new random location, thereby allowing problematic obstacles and noise to be overcome after one or more retries.

Grease or dirt on the docking rails of the robot or docking station may interfere with the supply of electrical current from the battery charger and therefore cause eventual failure of the automated recharging process (particularly in industrial environments). This can be easily overcome by occasional cleaning of the contacts by a human. However it should be noted that if the robot is expected to operate completely unaided and autonomously for long periods, then this may be an eventual reason for failure.

An additional problem is that coloured objects in the background may interfere with the image segmentation. This is a partially unavoidable vulnerability of colour segmentation. For example, if a red item is above a red and blue marker, the current vision system uses an average of the sizes of the two coloured segments, therefore the red and blue marker will be detected as taller than it should be, causing miscalculations in later stages. This has recently been reduced by using the known aspect ratio of the marker as a heuristic to determine whether the red segment or the blue segment is more likely to be in error. A large border around the coloured markers also reduces the effect of background colours.

A basic level of obstacle avoidance is implemented into the docking system, however this should be improved for the system to be considered robust in typical conditions. There are two expected modes of failure of this docking system:

1) If an obstacle is not seen by the Laser or Sonar sensors (such as obstacles that are under 15cm tall), then the robot may be halted and the long-term operation will fail.

2) If a considerably large obstacle is placed on the target point directly in front of the docking station (eg: a 1m diameter container placed 1.5m in front of the docking station) then the robot will be incapable of docking until the obstacle is removed, as this is the only target point the robot will try to reach. While it would be common knowledge for the robot operators to never place a large obstacle in this location, it could potentially occur in real-life situations and therefore should be overcome. This could involve dynamically adjusting the target point where the robot tries to dock from if it has found obstacles after several unsuccessful trials. Note that small or temporary obstacles are avoided by simply moving to a new random location and trying again.

5 Results and Discussion

5.1 Docking Results

The purpose of this docking system is to allow unaided long-term operation of the robot. Several tests have been performed to show the typical results obtained by the docking system. In practice, there are high variations in docking duration, particularly when obstacles are introduced. When the robot begins docking from a position where the docking station is visible, then the docking operation is performed quickly on almost all occasions. However, if the robot begins in a location where the landmarks aren't yet visible, then the robot moves to random locations until it can find the docking station. This can take an unbounded length of time, however it will be typically found within 10 minutes when in the same room, but is highly dependent on the environment. Since a high-level navigation system (RatSLAM) is expected to bring the robot close to the docking station every time, this searching behaviour is an additional fail-safe, so the docking system will not usually require long searches for the docking station.

Obstacle	*Avg Duration (mins)*	*Success*	*Trials*
None	2.41	100.0%	75
Large	2.07	100.0%	25

Table 11: Duration and success rate after 100 docking trials with and without obstacles, while also performing RatSLAM and the Web Interface. The high success rates are due to the robustness and very high tolerance for error.

Table 11 shows that some obstacles can in fact improve the average response of the system. Since an obstacle was placed near the desired target point, the robot was more likely to stop when near the correct position. This resulted in faster response times in most cases, however there were cases requiring upto 10 minutes for a single dock, due to the obstacle. When the robot is placed on the opposite side of a large cluttered room, docking typically requires 30 minutes for completion. Table 11 also shows a 100% success rate across 100 trials, showing the reliability of the system. However it is important to note that these tests were performed in a brightly lit room with easily detectable obstacles. In real environments, obstacles pose a real threat to the long-term operation of the system. If the robot is expected to operate completely un-aided for long-term operations, the environment must be made clear of obstacles that aren't detectable by the Laser range finder or other sensors.

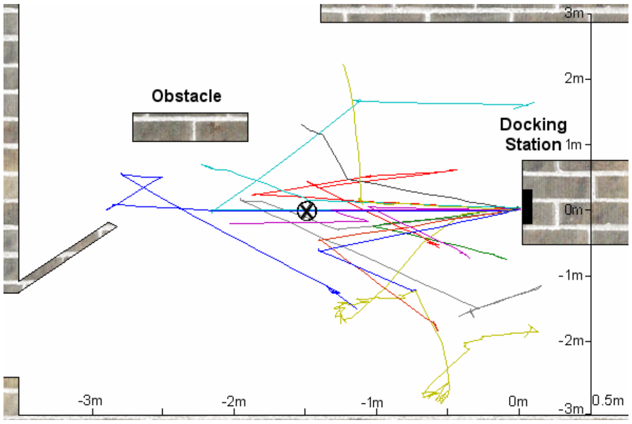


Figure 12: Typical docking trajectories. X marks the target point that the Ballistic stage attempts to move to.

5.2 Visibility of the Markers

As discussed earlier, docking requires both short-range and long-range landmark detection, hence many designers use seperate sensors and strategies to cover the wide range. By using only a vision sensor, the docking behaviour must be capable of accepting limited visibility in various zones. For example, there will be locations where one of the markers is visible but the other is not.

	Range	*Visibility*	*Abilities*
Zone 1	5m - ∞	No markers	Wander randomly
Zone 2	1 - 5m	Both markers are visible	Calculate Pose from markers
Zone 3	0.3 - 1m	Just one marker is visible	Get Distance & Bearing from marker
Zone 4	0.1-0.3m	Marker is only partially visible	Get Distance & Bearing from width of marker

Table 13: Zones of visibility when using a standard camera with a ±22° view.

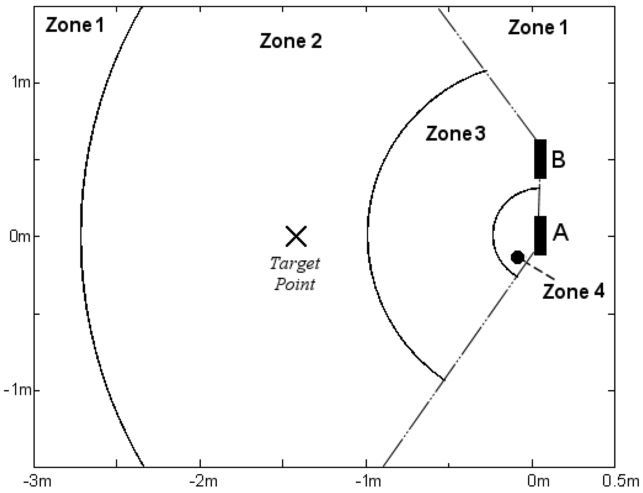


Figure 14: Zones of visibility using a standard camera.

5.3 Docking Accuracy

Since the mechanical docking system created allows reliable operation despite significantly large misalignments and low accuracy, the docking behaviour doesn't require a high level of precision. Nevertheless, the docking accuracy has been tested.

It is found that positional error is very minimal, always within ±4cm of the allowable ±18cm range. This is due to the fact that the robot enters the docking station in a virtually straight line from the target point, 1.5 metres away. The centre of the visible landmark is used for visual servoing in this final stage, and thus will always dock in the centre of the docking station.

However, angular error can be significantly larger, since it is difficult to ascertain whether the robot is accurately facing the docking station without a digital compass or similiar measuring instrument. By using noisy image data alone, angles within ±10° are generally achieved (see Figure 15). This is adequate for this system since the physical docking station supports reliable docking from ±55°, hence in over 100 docking operations there has yet to be an occasion of a failed docking due to inaccuracy.

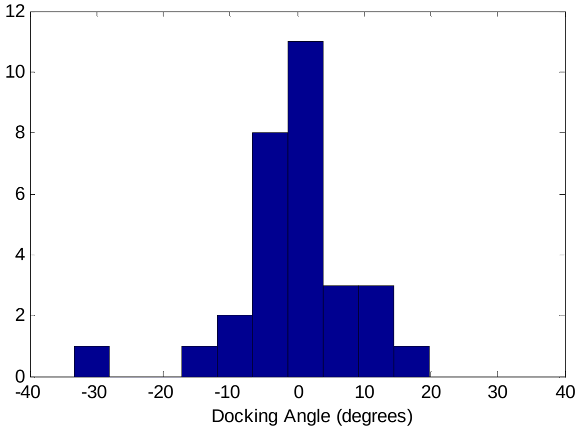


Figure 15: Histogram of typical docking angles. Docking typically occurs within ±10°, however there are occasions of upto ±40°. The docking station allows the robot to safely dock upto ±55°, therefore docking has always been a success. In the unlikely event that the robot tried to dock further than this, the docking station will appear as an obstacle, thus the robot will retry from a different angle.

5.4 Web Interface Robustness

Although the GUI for the web interface being created is not finalised, the current system has been stable for extended periods under harsh conditions, such as 50 simultaneous web clients running on the same or different machines, multiple web clients simultaneously crashing, power suddenly being switched off on a machine, switching power off on the robot, causing RatSLAM to crash, etc. These tests and long-term tests of 50 simultaneous web clients for several days have yet to crash the web interface system. The only issues seen are when a large network delay causes a web client to timeout. This will disconnect a web client, simply requiring the user to click on the ‘Refresh’ button of their web browser, but what is of importance is that the Relay web server will remain unhindered.

Nevertheless, it would be foolish to consider the system as stable simply because it has not crashed under laboratory testing. It will need testing over the Internet where delays are long and unpredictable. Problems should be expected to appear once the system is released to the World-Wide-Web, where there may potentially be hundreds of simultaneous web users, with far longer network delays, with different hardware and software configurations, and without appropriate training in the user interface. This will be the only true test of the system’s robustness.

6 Future Work

6.1 Verification of a Successful Connection

Currently the system verifies a successful docking based on the calculated distance to a landmark. However situations may arise where the robot has reached the docking station but a stable electrical connection is yet to be established. Testing for a stable electrical connection will overcome this vulnerability. A successful electrical connection will be detected by the presence of electrical current across a reverse-polarity protection diode between the robot’s batteries and the docking rails. This measurement will provide reliable testing for a succussful docking completion.

6.2 Cooperative Sharing of a Docking Station

There has been continued interest in cooperative robots to provide novel solutions to many data gathering or distributed problems. The current docking system is capable of allowing a single robot to use multiple docking stations, however the docking of multiple robots using one station will require a sharing protocol between the robots. This may be implemented by a radio transmission to specify whether a docking station is free or occupied. Alternatively, multiple robots could use a higher level management system to decide when each robot should dock, allowing only a single robot to dock at a time.

6.3 Web Interface Improvements

The web interface will recieve a considerable amount of modification for simple use of the non-Cartesian system. This may include a partially symbolic but significantly Cartesian map with various marked key-points (such as offices) with the ability to click on any previously visited position. A further improvement will be a significantly improved level of video feedback (see Figure 16). The existing system is only providing an extremely low resolution of 8x12 greyscale pixels, however the robot’s onboard hardware and networking system is currently being drastically upgraded to support high-resolution full-colour video streaming using the new JPEG2000 image compression format. This will provide a far greater experience for the web user, since visual feedback will be the most important factor to many web users.

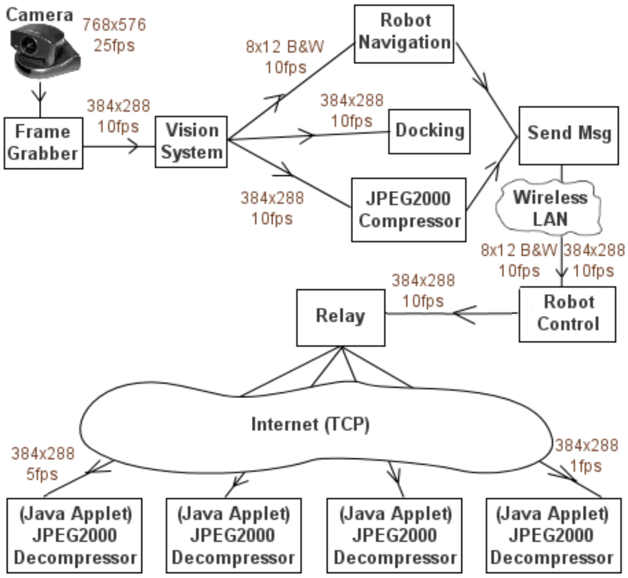


Figure 16: The new video streaming system will allow significantly higher quality video feedback.

7 Conclusion

As explained through this document, an automated docking system has been created that allows a robot to reliably perform long-term operations un-aided without dependence on the high-level navigation system. Also, the frame-work for a reliable web interfacing system has been created to allow human control of the robot through the Internet across large network delays. These are combined to allow long-term continuous control of the robot by web users across the world.

The robot is able to successfully dock despite noisy images or obstacles in the path, and can gradually find the docking station if the high-level navigation system incorrectly placed the robot out of view of the docking station. It is modular and reusable in the sense that it uses no prior knowledge of the environment, obstacles, goal position & axis, or robot position & axis.

References

[Arkin and Murphy, 1990] R.C. Arkin and R.R. Murphy, “Autonomous navigation in a manufacturing environment”, *IEEE Transactions on Robotics and Automation*, vol.6, pp.445-454, Aug 1990.

[Austin and Kouzoubov, 2002] D. Austin and K. Kouzoubov, “Robust, Long Term Navigation of a Mobile Robot”, In *Proceedings of the IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*, October, 2002.

[Barnes and Liu, 2001] N. Barnes and Z.Q. Liu, “Fuzzy control for active perceptual docking”, In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems*, Vol.3, pp.1531-1534, 2001.

[Brooks et al., 1973] V.B. Brooks, J.D. Cooke and J.S. Thomas, “The continuity of movements”, *Control of Posture and Locomotion*, New York: Plenum, pp. 257, 1973.

[Cassinis *et al.*, 2005 unpublished] R. Cassinis, F. Tampalini, P. Bartolini and R. Fedrigotti, “Docking and Charging System for Autonomous Mobile Robots”, Department of Electronics for Automation, University of Brescia, Italy, 2005. www.ing.unibs.it/~cassinis/docs/papers/05_008.pdf

[Franz *et al.*, 1998] M. Franz, B. Schölkopf, H. Mallot, H. Bülthoff, “Where did I take that snapshot? Scene-based homing by image matching”, *Biological Cybernetics*, vol.79 pp.191-202, 1998.

[Goertz and Thompson, 1954] R. Goertz and R. Thompson, “Electronically controlled manipulator”, *Nucleonics*, 1954.

[Hada and Yuta, 1999] Y. Hada and S. Yuta, "Robust Navigation and Battery Re-charging Systems for Long Term Activity of Autonomous Mobile Robot", In *Proceedings of the Ninth International Conference on Advanced Robotics*, Tokyo, Oct 1999.

[Hirukawa and Hara, 1998] H. Hirukawa and I. Hara, “The web top robotics”, In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 1998.

[Milford, 2005] M.J. Milford, G. Wyeth, and D. Prasser, "Efficient Goal Directed Navigation using RatSLAM," In *Proceedings of the International Conference on Robotics and Automation*, Barcelona, Spain, 2005. http://www.itee.uq.edu.au/~milford/Publications/publications.htm

[Mitchell and Labrosse, 2004] T. Mitchell and F. Labrosse, “Visual homing: a purely appearance-based approach”, In *Proceedings of Towards Autonomous Robotic Systems*, University of Essex, Colchester, UK, 2004.

[Minten *et al.*, 2001] B.W. Minten, R.R. Murphy, J. Hyams, and M. Micire, “Low-order-complexity vision-based docking”, *IEEE Transactions on Robotics and Automation*, vol.17, issue 6, pp.922-930, Dec 2001.

[Oboe and Fiorini, 1988] R. Oboe, P. Fiorini, “A design environment for Internet-based telerobotics”, *The International Journal of Robotics Research*, vol.17, issue 4, pp.433-449, April 1988.

[Oh *et al.*, 2000] S. Oh, A. Zelinsky, and K. Taylor, “Autonomous battery recharging for indoor mobile robots”, In *Proceedings of the Australian Conference on Robotics and Automation*, August 2000.

[Prasser and Wyeth, 2003] D. Prasser, G.F. Wyeth, “Probabilistic Visual Recognition of Artificial Landmarks for Simultaneous Localization and Mapping”, In *Proceedings of the IEEE International Conference on Robotics and Automation*, IEEE Press, Taiwan, 2003.

[Rizzi *et al.*, 1998] A. Rizzi, G. Bianco, and R. Cassinis, “A bee-inspired visual homing using color images”, *Robotics and Autonomous Systems*, vol.25, pp.159-164, 1998.

[Santos-Victor and Sandini, 1994] J. Santos-Victor and G. Sandini, "Visual behaviours for docking", *Computer Vision and Image Understanding*, vol.67, issue 3, pp.223-228, Sept 1994.

[Schulz *et al.*, 2000] D. Schulz, W. Burgard, D. Fox, S. Thrun, and A.B. Creemers, "Web interfaces for mobile robots in public places," *IEEE Robotics & Automation Magazine*, vol.7, issue 1, pp.48–56, March 2000.

[Sheridan, 1992] T.B. Sheridan, *Telerobotics, Automation and Human Supervisory Control*, Cambridge, MA: MIT Press, pp.107, 1992.

[Silverman *et al*., 2002] M.C. Silverman, D. Nies, B. Jung, and G.S. Sukhatme, “Staying Alive: A Docking Station for Autonomous Robot Recharging”, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, 2002.

[Simmons *et al.*, 1999] R. Simmons, J. Fernandez, R. Goodwin, S. Koenig, and J. O'Sullivan, “Xavier: An Autonomous Mobile Robot on the Web”, *Robotics and Automation Magazine*, 1999.

[Taylor and Dalton, 2000] K. Taylor and B. Dalton, “Internet robots: A new robotics niche”, *IEEE Robotics and Automation Magazine*, vol.7, issue 1, pp.27-34, March 2000.

[To and Mann, 1998 unpublished] P. To, G. Mann, “Self-recharging as survival-oriented robot behaviour”, Unpublished Honours Thesis, School of Computer Science and Egineering, UNSW, Australia, 1998. wwwit.murdoch.edu.au/~mann/ROBOTS/robots.html

[Walter, 1953] W.G. Walter, *The Living Brain*, W.W. Norton, New York, 1953.

[Woodworth, 1899] R.S. Woodworth, “The accuracy of voluntary movement”, *Psychological Review 3* (Monograph Supplement), Issue 13, pp.1-119, 1899.

[Yamada *et al.*, 2005] T. Yamada, K. Nagatani, and Y. Tanaka, “Long-term Activities for Autonomous Mobile Robot”, In *Proceedings of the International Conference on Field and Service Robotics*, Port Douglas, Australia, pp.297-308 (preprints), July 2005.

[Zebrowski and Vaughan, 2005] P. Zebrowski, R.T. Vaughan, “Recharging Robot Teams: A Tanker Approach”, In *Proceedings of the International Conference on Advanced Robotics*, July 2005.