GO-2 :- AN AUTONOMOUS MOBILE ROBOT FOR A SCIENCE MUSEUM

Ray Jarvis and Alan Lipton
Intelligent Robotics Research Centre
Monash University
Wellington Road
Clayton, Victoria 3168
Australia
ray.jarvis@eng.monash.edu.au

Abstract

Whilst many fragmented capabilities in the areas of mobile robot localisation, environmental modelling, path planning and motion control have been reported in the literature, as have also some complete systems which perform adequately in short-term laboratory demonstrations [1], this paper describes the rationale behind and implementation details of a complete and robust autonomous mobile robot navigation system which will operate for long periods with minimal human intervention under the gaze of a critical public audience at a science museum (ScienceWorks in Spotswood, Melbourne).

1 Introduction

It is well known that autonomous mobile robot navigational capabilities depend critically on the interplay of localisation, environmental modelling and path planning systems along with necessary mobility control and communications considerations. The function of the robot is also directly relevant.

In known environments, environmental modelling can be reduced to representational aspects. In the system considered here, however, the obstacle field and surrounding wall 'constraint' environment is not only initially unknown but can be time-variant to some extent; that is, environmental modelling and path planning must be dynamic and thus ongoing.

Three basic considerations were required of the robot system in the context of its expected operation within a science museum on a continuous daily basis over a number of months. The first was that it should do something of interest to the visiting public which would indicate intelligent behaviour and in an entirely autonomous mode. The system described in this paper goes to (GO-2) a specified goal via a collision-free path in an initially unknown obstacle field environment. The second was that it should be robust and need minimal attention by museum staff over lengthy periods of operation. The third was that the system should neither damage itself nor be a threat to the safety of the public.

It was decided to also include a teleoperation mode with on-board stereo cameras used to give the operator a view of the environment from the robot's perspective and a joy-stick for motion control. The logic here was that the teleoperation mode would allow an operator (public spectator/participant) to experience the complexity of obstacle avoidance and path planning first hand to better appreciate the capability of the robot in autonomous mode and thus be aware of what kind of intelligent behaviour was being exercised.

An important extra constraint was imposed by the circumstance surrounding the project - the whole system had to be operational within four months of the 'go ahead' given by the museum authorities. The 'show' had to start on time.

Figure 1. Shows the GO-2 robot without its cover on.

2 Power and Computing Requirements

It would have been nice to have the mobile robot entirely self contained. However, this would have meant recharging and/or replacing batteries on an inconveniently regular basis which would have introduced interruptions to service and required the attention of museum staff. Thus, at an early stage, it was decided, with some reluctance, to use a tether despite its intrusion being somewhat inelegant. This decision was also influenced by being able, with a tether, to keep the powerful computer which was to operate the system off-board and thus make its monitor available for displaying aspects of environmental modelling, path planning and localisation during the progress of a demonstration. The tether also allowed the on-board stereo camera signals to be viewed on a separate monitor without the use of noisy radio video transmitters.

On-board batteries driving the wheels of the robot and supplying power to on-board sensors (including the stereo camera pair) are trickle-charged on a continuous basis using a battery charger, with mains voltage supplied through the tether.

A Silicon Graphics Indy Workstation was used as the only computational resource for the system. Its powerful computing capability and excellent graphics made it ideal for the purpose. However, one serious limitation was that it had only two RS232 ports and these were prone to fail with spurious signal overloads. Both optical isolation and diodes protection were introduced after an incident of failure caused by capacitively coupled transient 'spikes'.

3 Locomotion

Standard wheelchair controller/motor/gear/wheel sets were used in differential steering mode with the drive wheels centrally placed and castors used to stabilise the chassis. The wheelchair gear was known to be reliable and capable of hauling a human weight for up to six hours without battery recharging. Furthermore, the controller included proportional, integrative and differential (p.i.d.) control, dynamic breaking and a safety requirement that both wheels must be in neutral before they can be driven using separate analog voltage signals either from a joy-stick or other source (e.g. D/A converter on a computer). The differential steering mode allows turning on the spot as well as a variety of curved trajectories both forward or backward. The tyres were filled with solid rubber inserts so as to obviate the need for pumping them up occasionally.

4 On-Board Sensors

Putting aside for the moment the stereo camera pair (which is not part of the autonomous navigation system), two on-board sensors were used. These were chosen because of their suitability based on earlier experience and that they were both capable of transmitting their data via RS232 serial line protocol. The first is a Denning scanning localiser which is essentially a bar code reader with the bar codes used as passive beacons at known positions up to 100 feet away. The angles and identities of three or more such beacons are sufficient to provide (using simple geometry) the position and orientation of the robot up to 10 times per second. In fact, redundancy is exploited to improve accuracy and guard against faulty data. Typically 6 to 10 bar code beacons are used for this task if the environment is contained within basically one space. Navigation between rooms etc. can be supported by the use of up to 32 distinct bar codes, some perhaps repeated in various scan orderings to allow sufficient discrimination between contexts.

The second sensor is an Erwin Sick scanning time-of-flight range sensor capable of providing a set of range readings at $1/2^0$ intervals over a 180^0 sheet scan in 1 second with ±3 cm accuracy with up to 50 metres maximum range. This sensor was considered ideal for providing environmental modelling data for obstacle fields made up of objects whose cross sections at the horizontal level of the swept beam from the range-finder, when projected on the floor, provided a conservative estimate of forbidden areas into which no part of the robot could intrude. Objects like boxes are ideal in meeting this requirement as are vertical walls. The scanning beam was placed approximately 10cm above the floor with a horizontal sweep facing forward.

5 Motion Control

Having allocated the two existing RS232 ports of the workstation to acquiring data from the Denning localiser and the Erwin Sick range-finder, the analog signals required to drive the two motor/gear/wheel sets separately had to be provided by a different means unless one or other of the RS232 ports were to be used in a multiplexed mode. This latter solution was not considered sufficiently simple as to allow for easy fault diagnosis and correction and was not pursued. Instead, a row of 18 phototransistors, in two groups of 9, were mounted on a circuit board and placed horizontally up against the bottom of the workstation screen. These feed two 9 bit D/A converters which drive the wheels. Thus graphics on the screen can drive the wheels. This method not only proved easy to use and debug/calibrate but also gave the electrical protection of optical coupling. Together with the optical coupling provided for the RS232 ports, all links between the Indy and the robot tether are electrically isolated, thus affording a considerable degree of protection against electrical surges and cross interference.

The robot is provided with a bumper in case of collision from any direction but more likely in the front. A bicycle inner tube filled with water is used both as a passive buffer and as a collision detector. A transparent tube attached to the tyre valve outlet (valve removed) is oriented vertically and the water level arranged so that an opaque float in the tube obscures the passage of light through it, as detected by a light emitter/detector pair, when the tyre tube is impinged upon. Power to the wheels is disengaged for a preset period of a few seconds. The navigation system knows a collision has occured when the robot stops moving for a preset period (say 1 Sec.) when it is meant to be moving.

6 The Navigation System

Now the stage is set to display the integration of instrumental data and algorithmic constructions in providing a total autonomous mobile robot navigation system whose only 'a priori' knowledge about the environment is the location of beacons and fixed surrounding walls and the compliance of the obstacles to the requirements of the simple mode of horizontal sheet range scanning provided by the Erwin Sick range-finder.

The floor space is regarded as a blank sheet (with fixed walls indicated for simplicity) marked off in a regular tessellation of grid squares which define cells. One or more goal point cells are specified by the operator. A tour of goals, nearest first at each stage, will result. The robot must go to (hence the robot's name GO-2) the nearest goal point if it can, mapping its environment and planning its path on a quasi-continuous basis along the way whilst avoiding cells indicated as occupied by obstacles. In fact, whilst the localiser data is used more or less as fast as it can be acquired to correct the trajectory of the robot along straight line segments of the globally planned current path, new range data is incorporated into the incrementally developed and corrected environmental map only at the beginning and at bends in the path whilst the robot is stationary. The monitor screen shows the position and orientation of the robot at all times as well as the current environmental map, including the 'growth' of obstacle cells by the radius of the robot to allow it to be considered as a point in 'configuration space', and the current planned path to the goal.

The path planning is based on distance transform methodology developed by the first author and other colleagues over a number of years [2,3]. An example of its capability are shown in Figure 2. The distance propagation algorithm is sufficiently fast as to permit redetermination of the optimal global path at each adjustment of the environmental map, whether additive or subtractive or both. Thus dynamic environmental changes can be accommodated without compromising the global stance in path planning.

A number of approaches to accommodating time variance in the obstacle field could be entertained. In this implementation only what can be 'seen' by the range-finder is subject to change whether it be additive or subtractive. Thus, if an obstacle which was once observed but is currently out of sight of the range-finder were to be removed, the planner would not immediately be aware of this change. Obviously, various styles of forgetting time constants could be introduced to overcome this shortcoming. However, to date, the simple scheme adopted has proven reasonably reliable if not always optimal. Figure 4 shows a typical navigation run.

7 Evidence-based Environmental Modelling

In this application, the workspace is modelled as an occupancy grid in which each cell either contains an obstacle or free space. However, determination of a cell's status is affected by both environmental dynamism and uncertainty in sensor information. So each occupied cell can be given an evidence (of occupancy) value which the system uses to estimate the likelihood of occupancy. The notion is that reinforcement of sensor information increases the evidence whereas contradictory sensor data decreases it.

In an application such as this, in which the general layout of the workspace is known (save for some obstacles whose positions cannot be relied upon), there should be a mechanism for defining grid cells as certainly occupied. In this implementation, such cells are tagged as **permanent** and are not updated by the system. Thus the stationary walls are considered permanent fixtures.

There are four distinct classes of environmental sensing events which can be used to update the evidence value of a cell. They are:

- *contact* - an obstacle is detected in a previously free cell,
- *reinforcement* - an obstacle is detected in a previously occupied cell,
- *contridiction* - no obstacle is detected in a previously occupied cell which is clearly visible' to the sensors, and
- *maintenance* - a previously occupied cell is not visible' to the sensors. (ie, it is behind the robot or obscured by some other obstacle).

The impact each type of event has upon the evidence associated with a cell depends on different factors. A *contact* event provides an initial evidence value E_i (the *contact* constant) and depends on the reliability of sensors. If they can be throughly trusted, this value can be quite large. However, if the sensors are very error prone, this value may be quite small and the cell will require *reinforcement* to achieve a large evidence value.

A *reinforcement* event should increase the evidence value of a cell and a *contradiction* event should lower it. Assuming there is a time delay of Δt_S between sensor readings, time constants can be defined to handle these modelling events. Consequently, a *reinforcement* event will increase the evidence of a cell by $\Delta t_S / \tau_r$ and a *contridiction* event will decrease the evidence by $\Delta t_S / \tau_c$, where τ_r is the *reinforcement* time constant, and τ_c is the *contridiction* time constant. The two main factors which affect the choice of these constants are sensor reliability and environmental dynamism. Unfortunately, these factors represent something of a dichotomy. If sensors are unreliable, the time constants should be large to prevent instability in the model. However, if objects in the workspace are prone to frequent movement, then at least τ_c (evidence time constant for contradiction)should be quite small or there is a large time lag between the movement of obstacles and updating the model resulting in obstacle 'smearing'.

A *maintenance* event occurs when there are occupied cells which are beyond the field of view of the sensors either because they are behind other occupied cells (occluded) or beyond the viewing limits of the sensors. In this case the cell's evidence could decay by $\Delta t_S / \tau_m$ where τ_m is the *maintenance* time constant. Choosing a value for this time constant presents a dilemma. If it is small, unseen obstacles will quickly fade and the robot may become caught in a limit cycle wherein the robot elects a path to avoid an obstruction and then forgets about the obstruction and elects to turn around again - *ad infinitum*. At the other end of the scale, a large time constant will mean that obstacles are never (or at least - very slowly) forgotten. This means that any environmental dynamism which occurs outside the robot's field of view will not be considered in the model. This means that the robot may consider itself to be obstructed by obstacles which are no longer present, and it will not be able to find a path. Choosing a value for this constant is a delicate balance which depends upon the speed of the robot, the amount of dynamism in the environment, the size of the workspace and the type of tasks being undertaken by the robot.

To determine a cell's evidence value, consider that after its initial *contact* event, it has undergone N_r *reinforcement* events, N_C *contridiction* events, and N_m *maintenance* events. Its evidence value is then given by

$$E = \begin{cases} 0 & , E < 0 \\ E_i + \frac{N_r \Delta t_s}{\tau_r} - \frac{N_c \Delta t_s}{\tau_c} - \frac{N_m \Delta t_s}{\tau_m} & , 0 \leq E \leq E_{max} \\ E_{max} & , E > E_{max} \end{cases}$$

where $E_{\max}$ is a maximum allowable evidence value.

In this application, sensors are fairly accurate - the Denning localiser can position the robot to within 2cm and the Sick range finder is accurate (depending on the robots motion) to within 3cm. The update rate Δt_s is about 1s (depending on the robots motion), but the performance of the robot is not time-critical, so this update rate is acceptable. For these reasons, it was decided to completely trust the sensors. This means that when an cell is deemed to be occupied, it is immediately given the value $E_{\max}$ and when is is unoccupied it immediately given the value 0. To achieve this, both $E_{\max}$ and E_i are chosen to be 1, and both τ_r and τ_c are chosen to be Δt_s The main advantage of this is that the model will immediately react to changes in the environment directly in view of the robot. So anything happening right in front of the robot will be quickly reflected in the environmental model.

A more difficult question is how to deal with environmental information which is not in the robot's field of view. That is, how to choose τ_m. The nature of the project means that it is impossible to predict how time consuming a path planning experiment will be or how frequent and how extensive environmental dynamism will be, so a statistical approach to choosing τ_m is inappropriate. The solution which was opted for was to assume that unseen obstacles do not change - that is $\tau_m = \infty$. This is clearly an unjustifiable assumption because the robot will inevitably find its path blocked by obstacles which are no longer present. However, this situation is acceptable in the context of this project because there will always be a human operator present and when this situation occurs, the human is notified and asked to decide whether the robot is truly blocked or the model has just become inaccurate. If it is the model which is the problem, the user has the option of erasing the model and starting again from scratch.

8 Teleoperation Mode

In this mode the operator uses a joy stick to drive the robot, observing it directly, as displayed on the Indy workstation monitor screen, or via switching glasses as if a two eyed observer on-board the robot itself. Any or all these observation modes are available all of the time. Switching between teleoperation and autonomous mode is just a tracking ball click on the screen.

9 Extras

A simulation mode showing the robot use the current environmental map as a basis to implement the 'visit all' mode of the path planner as if vacuum cleaning the whole free space area is also provided. This mode is not implemented physically. Figure 4 illustrates this mode.

(Recent scientific discovery events involving large molecules found in Martian rocks have caused the museum staff to request that 'scanning the planet' be substituted for 'vacuum the floor' in the descriptive text.)

Also, the robot speaks phrases which ask the operator for instructions and indicate its operational status, thus rendering it more 'friendly'.

Finally, when the robot has completed a specified number of accumulative turns and is in danger of being tangled in the tether (which is lifted aloft by a boom), it announces it will 'unwind' and does so automatically.

10 On Show

When the robot system is installed (anticipated 9 August 1996) in the ScienceWorks Museum in Melbourne, Australia, visitors will be invited to try out the teleoperation mode, select goal points for the autonomous mode and observe the simulated cleaning (scanning) mode (see Figure 4). Occasionally attendants will invite visitors to enter the fenced off area of the robot's activity (perhaps an over-kill in terms of safety requirements) and to rearrange the obstacles whilst it is disabled. This last measure will reassure observers that the environment is not pre-learnt.

11 Conclusions

Whilst a relatively unadventurous design was used in constructing the autonomous mobile robot navigation system described in this paper, the focus on performance, robustness and simplicity has led to the implementation of a reliable and safe system which can operate with minimal human intervention over long periods, perhaps up to 8 hours per day for many months, under the scrutiny of the public gaze at the science museum. It is hoped that, amongst the public participants, an industrialist or two might have an insight as to how a customised version of the system being demonstrated might fulfil some industrial need. The Intelligent Robotics Research Centre at Monash University would then be glad to consider negotiating some research and development contract towards realising such an application.

12 Bibliography

[1] R.A. Jarvis, J.C. Byrne and K. Ajay, "An Intelligent Autonomous Guided Vehicle : Localisation, Environmental Mapping and Collision-Free Path Finding", Proc. International Symposium and Exposition on Robots, Sydney 6-10 Nov., 1988, pp.767-792.

[2] R.A. Jarvis and J.C. Byrne, "An Automated Guided Vehicle with Map Building and Path Finding Capabilities", invited paper, Fourth International

Symposium of Robotics Research, University of California at Santa Cruz, 9-14 Aug., 1987, pp. 155-162.

[3] R.A. Jarvis, "On Distance Transform Based Collision-Free Path Planning for Robot Navigation in Known, Unknown and Time-Varying Environments", invited chapter for a book entitled 'Advanced Mobile Robots' edited by Professor Yuan F. Zang World Scientific Publishing Co. Pty. Ltd. 1994, pp. 3-31.