The Design of a Pair of Identical Mobile Robots to Investigate Co - Operative Behaviours

Dale Carnegie, Andrew Payne & Praneel Chand

1. Introduction

When moving large or heavy items, the traditional tendency with machinery is to build a large mechanism capable of handling the load. This leads to continuous scaling of the mechanical size and weight of devices being built, with a proportional increase in expense. An alternative to this conventional trend is the use of cooperative behaviour. Humans are limited by their physical size and strength – and yet are capable of moving heavy furniture and other large items outside their individual strength ability. This is accomplished with the help of another person to share the load, or a large number of people for a substantial object.

Requirements of coordination, communication and precision control have restricted this approach being considered for mobile robots in the past. However, more powerful processors and software are removing these barriers, permitting cooperative robots to be seriously considered for a number of applications.

By working in parallel, cooperative behaviour can increase efficiency and reduce the time required to complete a task. Reliability is increased by introducing redundancy when using a team of robots, while cost is reduced due to the use of smaller simplistic machine designs. Application specific design and manufacturing costs can be removed by fabricating the robots semi-generically. Reduced weight means less preparation and upkeep of the working environment, and new complex tasks can be introduced which are too difficult for a single entity to achieve.

We demonstrate a methodology for the construction of a low-cost, versatile, highly manoeuvrable and computationally powerful robot capable of autonomous operation. We demonstrate that these robots can be fabricated at a small fraction of the price of equivalent commercial systems. As our interest is in the development of cooperative robotic behaviour, the software control system, machine interface and an introduction to cooperative robotic systems are also discussed.

1.1 Commercial Mobile Platforms

Most robotic arms are designed to sit on a factory floor, are large, heavy and use a substantial amount of power. A small number of suitably designed mobile platforms with a mounted manipulator arm are made by manufacturers around the world. They are mainly used for research or remote bomb disposal and are sparse and expensive.

The MR-5, shown in Figure 1, is a bomb disposal robot built and sold by ESI (Engineering Service Inc., http://www.esit.com). It is a remote controlled platform, with a manipulator

Source: Cutting Edge Robotics, ISBN 3-86611-038-3, pp. 784, ARS/plV, Germany, July 2005 Edited by: Kordic, V.; Lazinica, A. & Merdan, M.

arm consisting of up to eight joints that can reach 1.7 m supporting a payload of 20 kg. It has dimensions 70 × 127 × 79 cm (w × l × h), weighs 250 kg, a maximum speed of 2.5 km/h and can operate for 3 – 5 hours. The system retails for US$80 K to $140 K depending on the configuration.

The MR-5 is controlled by a human operator via a remote station. The communication is either wireless RF (Radio Frequency) with a 500 m range (line-of-sight), or cable with a range of 200 m. Tracks can be fitted over the wheels for a rougher environment. Optional sensors include a pan-and-tilt zoom camera providing video and sound feedback to the user through the control station, an infrared camera, and an x-ray mounting assembly.

Figure 1. ESI's MR-5

RoboProbe Technologies Inc. (http://www.roboprobe.com) produce a remote controlled bomb disposal robot shown in Figure 2. The robot is similar to the MR-5, but is smaller (43.5 × 91.5 × 40.5 cm), lighter (35 - 45 kg) and with an operating time of 2 – 3 hours. Permanent tracks are used rather than wheels, giving it a similar maximum speed to the MR-5. The arm has a reach of 96.5 cm and can support a 4 kg payload. Incorporated sensors are 3 cameras, an IR (Infrared) camera with pan-and-tilt capability and IR light source (on a high mounting), colour drive camera with halogen lights (at the base of the arm), and a colour camera mounted on the claw.

The MURV-100 is similar to the MR-5, but very lightweight (only 23 kg). Shown in Figure 3 fitted with ten wheels, these can be changed for either six large wheels or tracks. It has dimensions: 43 × 58 × 30 cm, an operating time of 2 – 4 hrs and a modest top speed of 0.39 km/hr. The arm can extend 66 cm and support a 4.5 kg payload. Approximate cost is US$35 K.

Figure 2. RoboProbe Technologies Inc. bomb disposal robot

Figure 3. The lightweight MURV-100

Manufactured by Defenders Network Inc. (http://www.defend-net.com), the MURV-100 offers communication by either a cable or by a standard wireless system, up to 300 m in each case (line of sight required for wireless operation). Optional sensors include a tilt-and-pan camera, a claw camera, and sensors capable of detecting toxic gases, radioactive materials and explosives.

The PowerBot (ActivMedia Robotics, http://www.activrobots.com) fitted with a PowerArm, (Figure 4) is designed for research use. A number of sensors are incorporated into the design to complement this, allowing control to be provided by the robot itself rather than from a remote user. It is designed for indoor use, with a small ground clearance and limited traction on rough terrain. It has dimensions 62.5 × 85 × 43 cm and a maximum speed of 6 km/hr. The arm can reach 80 cm, and support a 2 kg payload

Figure 4. PowerBot with PowerArm from Activemedia Robotics

The PowerBot has a HitachiH8S-based microcontroller onboard, along with shaft encoders and sonar sensors to provide control. An onboard PC, laser scanner, camera and other optional extras are available. Drive is provided to two wheels, with casters added for stability. The robot comes with a price tag of US$25 K to $85 K depending on the options chosen.

Mobile robotic arm kits are available from Lynxmotion (http://www.lynxmotion.com). These are only miniature robots (the base is approximately 20 cm × 20 cm), and are not particularly suited to the applications envisioned for our cooperating robots.

In summary, although mobile platforms supporting an extendable manipulator arm are commercially available, they are generally prohibitively expensive. The affordable options are more hobbyist devices unable to support the sensors and processing power our application requires. The solution then, is to design and construct our own mobile platform and manipulator arm, at a low cost but not at the expense of functionality.
The remainder of this paper details the construction of a pair of mobile robots (christened "Itchy" and "Scratchy") that will be used to investigate autonomous cooperative behaviour. The robots will eventually be required to work together cooperatively to perform a single task, for example carrying a long piece of wood, although the actual cooperative architecture is outside the scope of this paper. A manipulator arm is currently being constructed and will be attached to the front of the robots above the two drive wheels.

2. Specifications

The design must consider the following attributes:

- **Environment** - The robots are required to operate in an outdoor terrain with a relatively smooth and level surface, such as concrete or mown grass.
- **Payload** – Each robot must be able to support all onboard components and an additional external payload of at least 40 kg
- **Manoeuvrability** – Each unit must be sufficiently agile to allow control in limited space. Precise control of the robot's path will be required
- **Self sufficient** – All required power and computation should be onboard the robot to facilitate independent, autonomous operation
- **Operating time** – Each robot must be capable of operating continuously for one hour
- **Communication** – The robots must have a communication link to allow them to transfer data, instructions or intentions
- **Scope** – The design must support future development, providing the computational capabilities, power supply and space to include additional sensors and actuators (particularly the manipulator arm)

3. Locomotion

A number of different drive systems are applicable to this project, and are reviewed in Carnegie et al., (2004). It is anticipated that the units will primarily be used outside on mostly flat surfaces. This avoids the complexity of legged, or self-laying track systems, and permits the implementation of a wheeled bicycle, tricycle or quadcycle locomotion.
A tricycle design was selected as it meets the requirements of:

1) Providing a simple design. Suspension or coordinated steering systems are not required when constructing a simple tricycle.
2) Minimising cost by reducing components required.
3) Being capable of adequately traversing a smooth outdoor terrain.
4) Offering manoeuvrability. It cannot perform point turns; however driving with the steering wheel at the rear will increase manoeuvrability.

5) Good stability. By locating heavy components near the bottom of the design and keeping the vehicle's height to a minimum, the centre-of-mass is lowered, reducing the chance of tipping. This is especially important when the manipulator arm is attached.

The methodology used to design these robots is also detailed in Carnegie et al. (2004).

4. Drive System and Motor Selection

With the chosen tricycle design, drive will be applied to the fixed pair of wheels to reduce the amount of slippage that would occur if only the steering wheel was driven. When driving both wheels, an allowance must be made for turning tight corners, since the inner wheel will travel a shorter distance than the outer wheel. Two solutions are available to this problem:

1) Drive each wheel independently using two separate motors.
2) Drive both wheels from a single motor through a differential gear, as used in automobiles.

The first solution requires an additional motor, gearbox and control circuitry. An advantage offered is that by increasing the power to one drive wheel, the robot will experience a rotational force which can assist steering during tight turns in a similar way that a wheelchair configuration steers.

The second solution only uses one motor and one reduction gearbox. This increases efficiency while reducing the required control sophistication and circuitry, though mechanical complexity is slightly increased by adding a differential gearbox. This was the method chosen.

Rather than purchase precision motors and gearboxes for propulsion, two motors (one for each mechatron), differential gears and axles were acquired from mobility scooters. The motors are 24 V dc, 400 W, with an electro-mechanical brake, and can power a 100 kg payload. Experimentally, it was found that these motors draw 3 A unloaded and 10 A when heavily loaded.

A steering motor is required to change the orientation of the single wheel over an expected angle range of ±60°. This motor must have sufficient torque to be able to turn the wheel and hold it in position, estimated to be approximately 10 Nm. A 24 V dc motor is preferred as this could be powered from the same voltage rail as the drive motors. Rather than use stepper motors or servomotors (which for the required torque and voltage values tend to be expensive), the decision was made to use a permanent magnet dc motor and gearing from a truck windscreen wiper. The motor shaft comprises a helical worm gear, which in turn drives a spur gear to give a 99:1 gearing ratio. To control the orientation of the steering wheel, a position sensor (in this case a potentiometer) is attached to the motor to provide feedback to a proportional-integral-derivative (PID) controller. The large reduction ratio provides extremely high torque, and makes position movement easier to control as the output shaft moves rather slowly. The peak current (loaded) drawn by this motor is approximately 6 A, whilst unloaded the current demand is 2 A.

5. Electronics

As these robots will be used to investigate cooperative robotic interaction, substantial processing power needs to be incorporated. Distributed embedded controllers can generally not run high level software packages such as MATLAB or LabVIEW, and a

similar argument rules out Handheld PCs or Palm devices. The robots are reasonably large and can accommodate a full sized PC board. This option was chosen over the incorporation of a Laptop PC due to the cost saving advantages. Additionally, the full-sized PC board allows hardware interface through serial, parallel, USB (universal serial bus), and PCI (peripheral component interconnect) connections.
The specifications of the selected system (chosen as a compromise between power and cost) are:

- 2.6 GHz Celeron
- 512 MB RAM
- 100 GB Hard drive
- Windows XP Professional

Communication between the robots and also to a base station is achieved using Netgear 401 2.4 GHz wireless network cards, using the 802.11b protocol. Transfer speeds are up to 11 Mbit/s with a strong signal.
A National Instruments PCI-6025E card provides data I/O to the CPU from the robot's sensors and actuators. The DAQ card provides 32 digital input/output lines, 16 analogue input lines, 2 analogue output lines and 2 counters/timers.
Global positioning system (GPS) positioning is achieved using Motorola M12 Oncore receivers. The GPS receiver is a 12 parallel channel receiver, capable of tracking 12 satellites at once. A position is reported once per second by a serial data transmission.
Shaft encoders provide position information from the main drive wheels. The encoders are HEDS-5701 panel mount optical encoders, providing a 500 count per revolution quadrature output, and are driven by a pulley and belt arrangement from the drive axle. Noting that the inflated tyre diameter of the robots is 250 mm, each odometer count corresponds to a linear distance travelled of 1.247 mm.
Six Sharp GP2Y0A02YK infrared (IR) object detectors provide a distance measurement to objects within a 20 - 150 cm range, with a stated accuracy of ±15 cm, arranged as indicated in Figure 5. By characterising and filtering each individual detector this accuracy was increased to ±5 cm (for indoor use). The detectors provide an analogue 0.25 – 2.85 V dc output corresponding to the distance measured, updating every 32 ms.

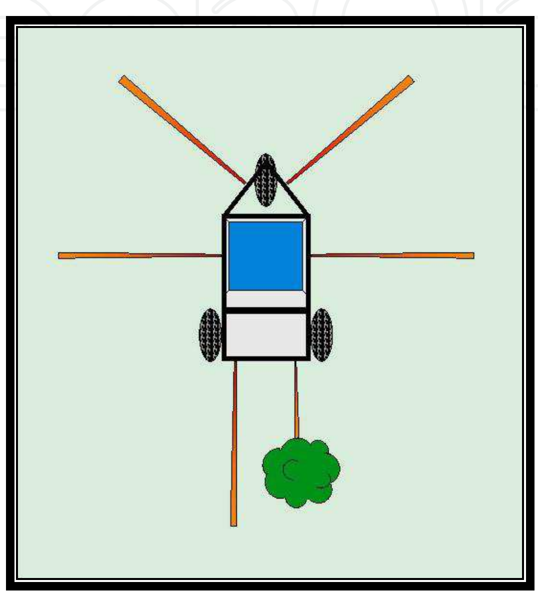


Figure 5. Infrared obstacle detection configuration

The detectors use a triangulation measuring method (rather than reflected intensity) that produces minimal variation in detected distance for objects of different colours or textures. The "forklift" style steering implies that the rear of the robot will swing outwards during a turn while travelling in forward direction. The two rear detectors are angled to monitor the area that the wheel will pass over during steering, and to also monitor behind the robot while reversing. This configuration however, does have a blind spot directly behind the mechatron – large objects will be detected by the angled sensors but smaller items may not be seen.

In most cases the robot will have travelled in a forward direction across the area of concern before any reversing manoeuvre is performed, allowing prior knowledge of the landscape to determine if the intended path intersects with obstacles.

To prevent the infrared light from adjacent emitters beating (due to cross talk resulting from off-axis emission), the two front emitters have each been mounted on a 1° angle away from the centre, making the intensity in any intersecting area negligible.

6. Power Requirements

A suitable power supply is required to run the PC. The power supply must offer the following rails (typical current rating shown for each rail):

- +12 V @ 7 A
- +5 V @ 25 A
- +3.3 V @ 8 A
- -5 V @ 0.5 A
- -12 V @ 1 A
- +5 V Standby @ 0.75 A

Previously the Mechatronics Group have used an uninterruptible power supply (UPS) (Cordes & Carnegie, 2002) to convert the 24 V battery supply into 230 V 50 Hz ac. This directly feeds into a standard ATX power supply of a desktop PC. This is a quick, easy solution but has the following major disadvantages:

- Efficiency is low converting 24 V dc to 230 V ac. and then back to 12 V dc and the other required voltages.
- The size of the UPS is large.
- The UPS weighs 10.9 kg.
- To turn the UPS on it must be connected to 230 V mains supply. For outdoor use, this means that the unit must be started in vicinity of a 230 V supply and remain in a standby condition until power up is required.

A better solution is to use an industrial power supply designed for the input voltage available, however these are only available from a small number of manufacturers and suppliers. An ACE-828C industrial ATX power supply was purchased, offering an 18 – 32 V dc input range. It has the same dimensions as a standard 230 V ATX supply.

A comparison of efficiency between using a UPS/230 V ac supply and the 24 V dc supply is shown in Table 1. The current shown is the amount drawn from the 24 V batteries by each power supply option while the CPU performs a defined task. The maximum values listed are the highest peak currents observed, and the "average working" values are the average currents drawn while performing a scandisk operation on the hard drive. The idle setting is the state where the power is on, but no processing or disk access is

operating, and the standby setting corresponds to the state where the CPU is turned off, with only the +5 V standby rail active on the motherboard.
Using this ATX supply rather than a UPS unit results in a 30% improvement in efficiency for the robots under their normal operating conditions (not taking into account the weight savings from utilising the lighter supply).
Current usage (as drawn from the 24 V batteries) is estimated as 2 A continuous, 2.5 A peak for the PC board, 10 A continuous, 17 A peak for the drive motor, 2 A continuous, 6 A peak for the steering motor and an additional 1 A continuous, 2 A peak for the other electronics. The power supply then, has to deliver 15 A continually for one hour, with the ability to supply peak currents up to 27.5 A.
Considering the range of batteries available (Carnegie et al., 2004), two E360C Champion flooded lead acid (FLA) automotive batteries were selected for each robot. These batteries have a cold cranking amps (CCA) rating of 360 A, and a reserve capacity (RC) of 60 minutes. As the RC of a battery is defined as its ability to supply a constant load of 25 amperes at 25 °C without the voltage falling below 10.5 volts (for a 12 volt battery), over 90 minutes of continuous operation should be expected from this source. However, these are not deep cycle batteries, and their life will be greatly extended if run-times are restricted to a maximum of one hour.

	UPS/230 V ac	**24 V dc**	**Increase in Efficiency**
Maximum	3.17A	2.20A	30%
Average Working	2.89A	2.06A	29%
Idle	2.21A	1.41A	36%
Standby	0.79A	0.18A	77%

Table1. Comparison of UPS versus dc power supply current consumption

7. Chassis

The chassis design is constructed from 16 mm box section, 20 mm and 25.4 mm right angle steel to provide a strong frame on which to mount the robot's components.
Designed with the driving wheels at the front, the motor is positioned above the drive axle. This increases tyre traction by applying the weight of the motor onto the two drive wheels. The differential gearbox, which is part of the axle, limits the overall ground clearance of the mechatron to 65 mm. The remainder of the chassis has a ground clearance of 100 mm to prevent the mechatron from bottoming out on uneven terrain.
The batteries provide a significant proportion of the overall weight of the mechatron. Maintaining the centre of gravity within the area defined by the wheels requires that the battery placement be in the centre of the robot. However, the batteries will be used to counterpoise the weight of the manipulator arm (to be installed at a future date) and any load it may carry, and therefore is placed near the rear of the design. This placement, combined with the motor above the front axle, distributes the weight over the entire mechatron, maximising stability.
The PC case has been placed in the centre of the design due to the considerable area it requires. Minimal access to the PC is required as all work can be performed remotely using the wireless network.

The steering wheel is at the rear of the mechatron, with the motor mounted above it. The output from the motor gearbox is used to directly change the steering angle of the wheel.
The electronic printed circuit boards (PCBs) are supported by an acrylic (non-conductive) tray mounted at the top of the mechatron. This permits easy access to the circuits during development, and also allows the addition of new circuits as further sensors and actuators are added. A rectangular steel frame is mounted above the PCB area to protect the circuits from being knocked during transportation. Figure 6 shows the completed chassis frame.
The battery supports are made from 25.4 mm right-angle steel which hold the base of each battery along the front and rear edges. With an area of 125 mm × 370 mm, the batteries (each 125 mm × 180 mm), fit in the supports with little movement.
The PC is mounted in the same manner as the batteries, using right-angle steel across the front and rear of the base. Both the batteries and PC can be removed through the side of the chassis rather than out through the top to allow easier access.

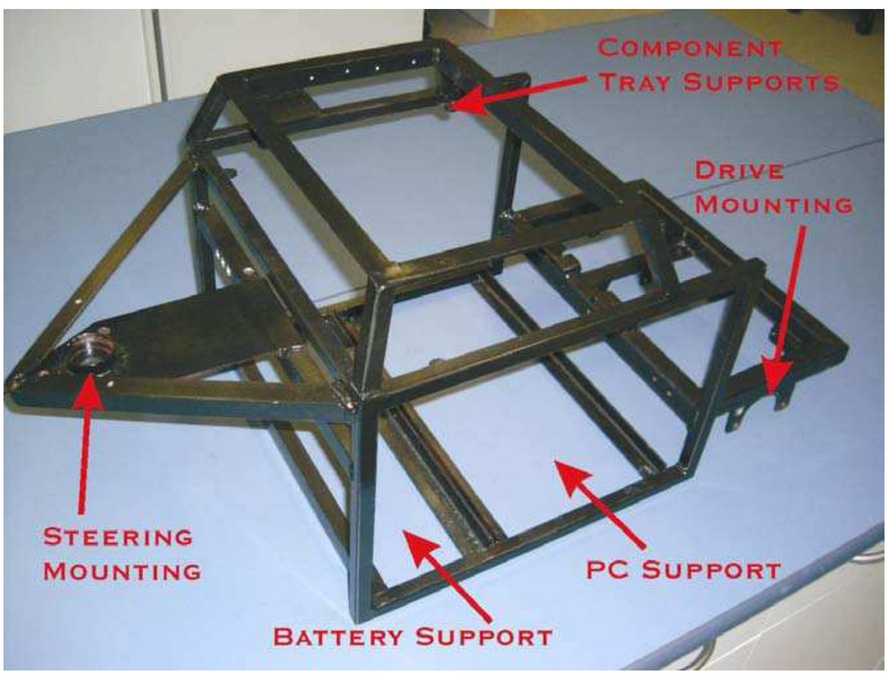


Figure 6. Chassis layout

8. Drive and Steering System

The drive system consists of a motor that drives an internal reduction gearbox (20:1 ratio) to a chain. The chain transfers the power to a differential gearbox (at a 1:1 ratio), which distributes the motion to the two ends of the axle. The assembled drive system is shown in Figure 7.
A 50 mm × 60 mm flat steel plate is welded 20 mm from the top of the mechatron, inside the PCB area to support the GPS antenna. This provides an unobstructed view of the sky for the antenna which uses a magnetic backing to secure it into place. The receiver can be mounted directly underneath the antenna in this configuration, reducing the area required for the complete GPS unit.The completed robots' dimensions are 865 mm × 600 mm × 430 mm (l × w × h), and they each weigh 58 kg (with onboard PC and batteries). The rear steering wheel supports 24 kg and the driving wheels combined support 34 kg, with the location of the centre-of-gravity shown in Figure 8. This is within the triangle generated

between the wheels, and offers good stability. The complete assembled chassis with the drive systems constructed is shown in Figure 9.

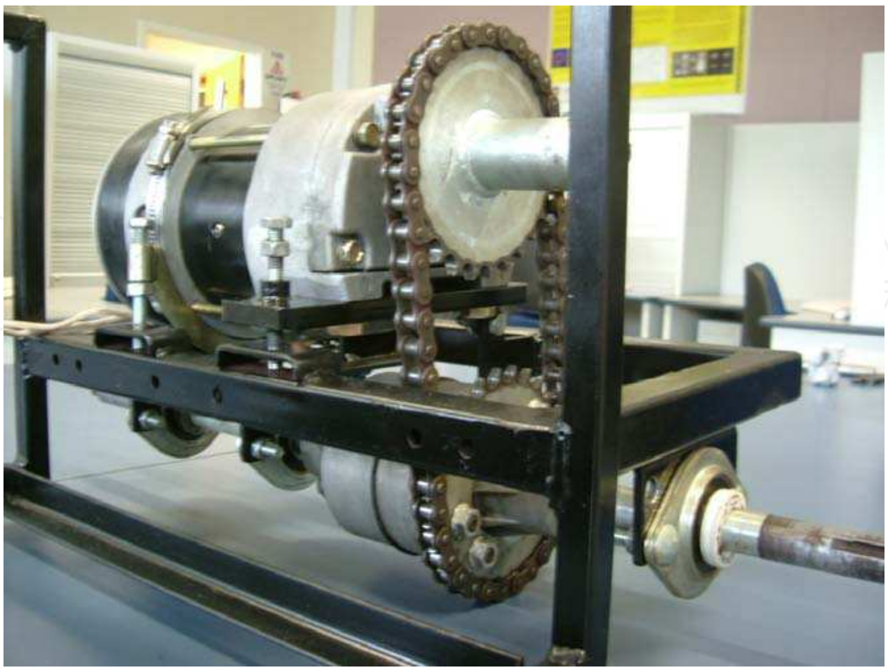


Figure 7. Mounted drive system showing chain tensioning method

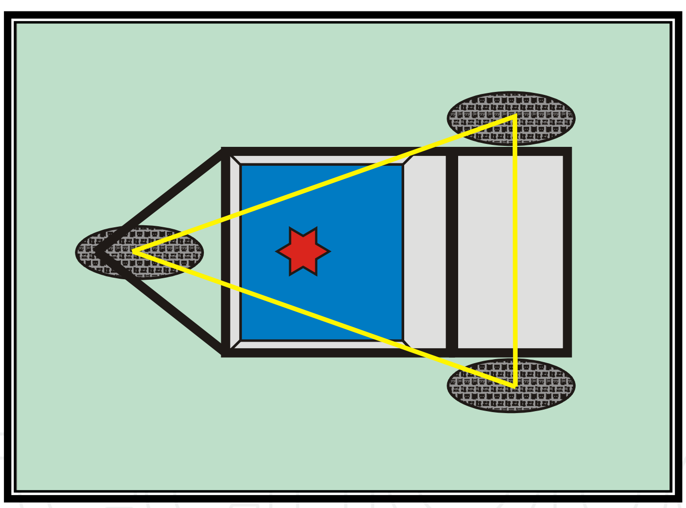


Figure 8. Centre of gravity of the loaded mechatron indicated by the star

Figure 9. Complete chassis and drive components

9. Motor Control

The robots' drive motors are powered via a full H-bridge arrangement using a pulse width modulation (PWM) signal generated by an embedded Philips P89C51RC2 microcontroller in response to a command output from the DAQ card (Payne & Carnegie, 2003).
The modified wiper motor used to orientate the steering wheel, requires both a control system, and a power drive system. The desired steering wheel position (derived from a software PID algorithm) is input to a UC3637 IC using an analogue signal between 0 – 10 V (matching the Lab-PC+ analogue output range). This voltage is compared to the voltage from the potentiometer mounted on the motor drive shaft. The error is amplified, and converted to two 0 – 24 V 30 kHz PWM signals used to control the motor's speed and direction.
The L298 H-Bridge motor driver IC powers the steering motor. The input drive signals require 0 - 5 V logic, so the output from the UC3637 is converted from 24 V to 4.9 V using zener diodes. The completed units (without manipulator arm) are illustrated in Figure 10.

10. Manual Operation

Eventually these robots will be programmed to undertake assigned tasks in a cooperative manner, independent of human intervention. For testing purposes, the devices are controlled from a remote laptop computer, via the wireless LAN connection, using a Joystick. Two different user interfaces have been designed: one for use by the base that is controlling the mechatrons, the other on the mechatron itself.

Figure 10. Completed mobile platforms, "itchy" and "scratchy"

The base provides a client connection to each mechatron. It allows control of the mechatron's speed and direction through the use of two sliders, and an emergency stop button is provided. The mechatron reports its current position, which is displayed on screen. Either a mouse or joystick (USB or analogue) can be used to control the sliders and hence the mechatron's movements. A screenshot of the base station is shown in Figure 11.

To assist development, all data received by the sensors are displayed locally on the mechatron. Any settings or variables can be viewed and altered with minimal effort. The left-hand half of the dialog box of Figure 12 provides control to the motors. If commands are being received through the network, the sliders will respond as the base sliders are moved. Otherwise the sliders can be moved by the user, providing a debugging interface. The progress bar next to each slider shows the mechatron's state, the current steering wheel angle, and the speed of the drive wheels. The stop button provides an emergency halt to both of the motors and can be used at any time.
The right-hand side of the dialog box is broken into a number of tabs, providing information on different parts of the system.

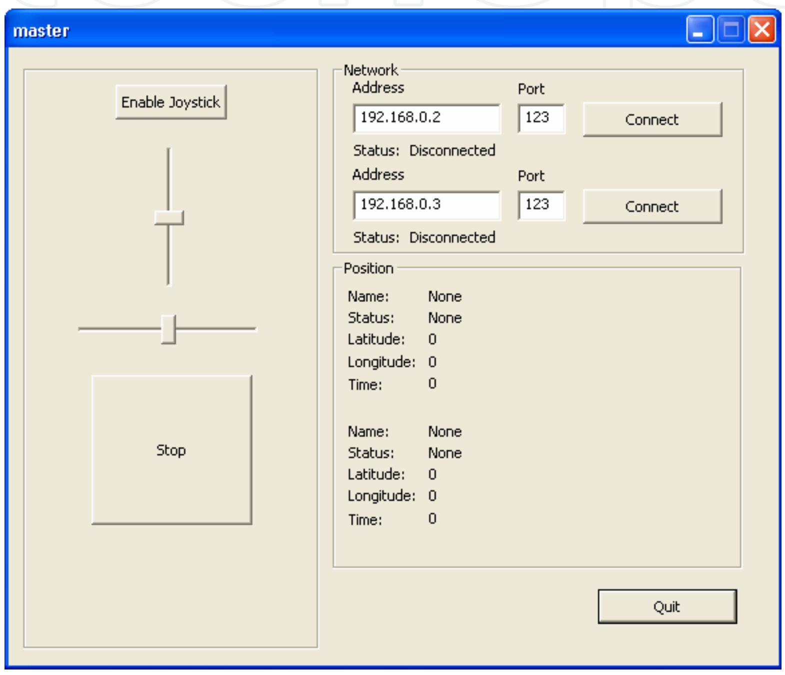


Figure 11. Base station user interface

The *hardware* tab provides network, microprocessor, GPS and IR object detector data. This tab is primarily used to display low-level data from the sensors and to indicate any problems. The status of the network port is shown, along with the number of clients connected. The microprocessor section shows the current PWM signals being sent from the PC, and the response (indicating any errors). The raw encoder values received are shown, again indicating any errors. The GPS shows the status of the receiver, the 3D position, the number of satellites available and the time. The time has been included to allow for logging of data on separate robots for later comparison. Using progress bars, the IR obstacle detectors show a graphical representation of the distance measured to any surrounding obstacles. A numerical representation is also shown with centimetre units.
The *control* tab is used to provide processed information to the user. The distance and speed of each drive wheel is shown, along with a calculated position and heading determined from this data. A large area of this tab has been provided for future development.
The *settings* tab permits on-the-fly alterations to the control system constants. This includes the PID constants for both the drive and steering systems, and physical constants

including the values used to convert encoder counts into a distance. These are provided to allow the user to "tweak" the values for each individual mechatron.
Each mechatron provides GPS data back to the base. This data is relayed so that each unit knows its own position as well as the position of others around it.

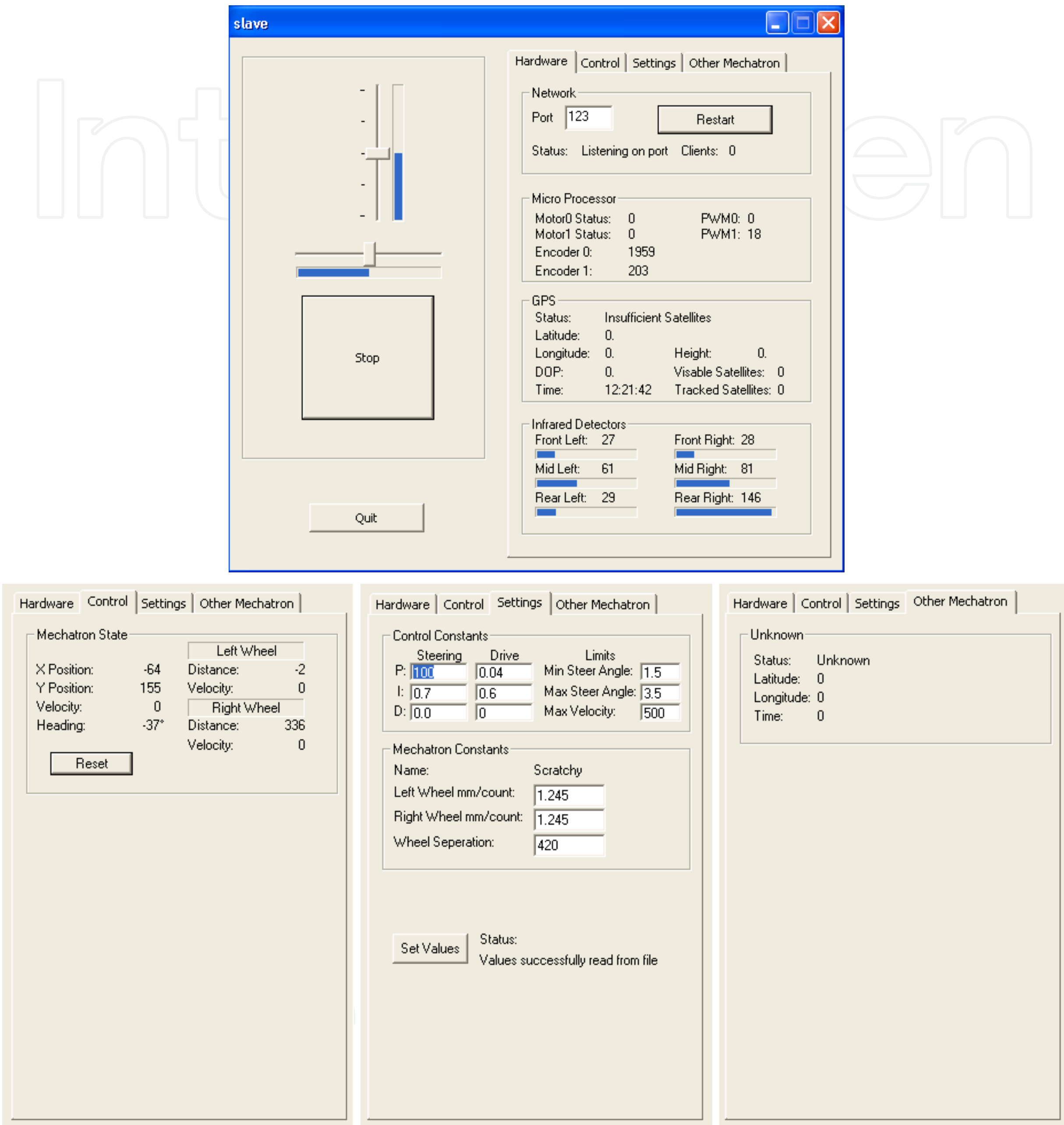


Figure 12. Rover user interface

11. Experimental Results

With the PWM control signal limited in software to a 75% duty cycle, the maximum velocity of the mechatron is 2 m/s. This limit is implemented to prevent damage to the mechatron and motor drive circuits if part of the system fails.
To determine the accuracy of both the GPS and dead-reckoning navigation techniques implemented, a mechatron was driven over a rectangular sports field of dimensions 55 m × 35 m four times.

The mechatron path started and ended at the same location so that any recorded data should form a closed loop when plotted.

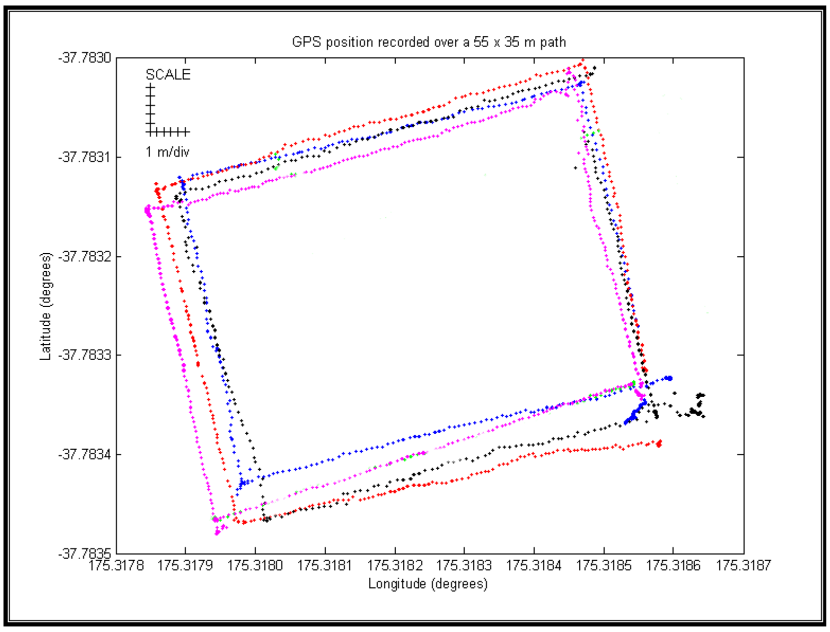


Figure 13. GPS reported position over a rectangular path

The reported position was logged to file every second, and the results of multiple runs illustrated in Figure 13. A scale is shown in the top left corner of the first plot, allowing a degree to metre conversion to be performed. The latitude and longitude axes are scaled independently.

Although each individual run plots a rectangular shape, the rectangle is not always closed as the start and end positions are not aligned (bottom right hand corner of each plot in Figure 13). The start-to-end measured distance (the same physical position) for the four runs was 7.81 m, 8.85 m, 6.33 m, and 0.94 m. This error is too large for the robots to operate in close vicinity of each other (<10 m) using a standard GPS position. A Vector V2X electronic compass has been used in another mobile robot constructed by the Mechatronics Group (Payne & Carnegie, 2003). This sensor has a stated operating resolution of 2°, which would produce a maximum lateral offset of 35 mm over a one metre distance. The use of this sensor approximately halves the start-to-end measurement error obtained from the GPS only trials.

To determine any increase in position accuracy when using differential GPS (DGPS) techniques, two M12 GPS receivers were used to report one position. Achieved by keeping the receivers stationary for a six hour period, the reported position was recorded over time with any variation due to GPS error. The test was repeated using pseudo-range corrections and also a block shift method (Payne & Carnegie, 2003).

Without corrections, standard deviations of 2.38 m and 4.64 m were recorded in the latitude and longitude directions respectively. Using a block-shift correction, the resulting standard deviation results were worse, 3.88 m and 6.90 m. However, the pseudo range correction improved the deviations to 1.68 m and 2.86 m. Whilst an improvement on the stand-alone GPS results, they are still not sufficient to permit accurate interaction between cooperating robots.

It is well known that odometry is subject to accumulating errors (Borenstein & Feng, 1996, Victorino & Borrelly, 2000). Using only odometry information for dead reckoning, the

same test yielded final displacement errors of up to 50 m. There was a noticeable pull to the left due to uneven tyre inflation, and whilst this could be partially offset by the use of an on-board compass, it is obvious that the accumulating errors for this technique render it unsuitable for distances beyond 10 m, especially if the operating surface is uneven.
For reliable cooperative operation where relative distance accuracy needs to be of the order of 10 cm, further improvements must be made to the GPS system. The purchasing of carrier phase capable receivers, and/or dual frequency receivers could provide the accuracy required, but the cost is prohibitive for this system, being in excess of US$10,000. Regardless, some fusing of the GPS information, the odometers (potentially using landmarks to reduce the accumulating errors), the compass and additional sensors such as inertial sensors, or a laser ranger (identifying landmarks), will be necessary to advance the goal of cooperative robotic interaction.
The infrared object detectors did function reliably in the outdoor environment, although their operational range was affected by changes in the ambient sunlight (between 1.0 and 1.5 m).

12. Software Control

With the exception of the poorer than anticipated GPS resolution, the manual testing of the robots was successful. Whilst not the subject of this paper (which intends to concentrate on the mechanical and electronic design of the mobile robot), it is appropriate to provide some details on the anticipated software control of these mechatrons. A hierarchical approach to the design is considered and illustrated in Figure 14, having been successfully employed on three other mobile robots constructed by the Mechatronics Group (Carnegie, 2002).

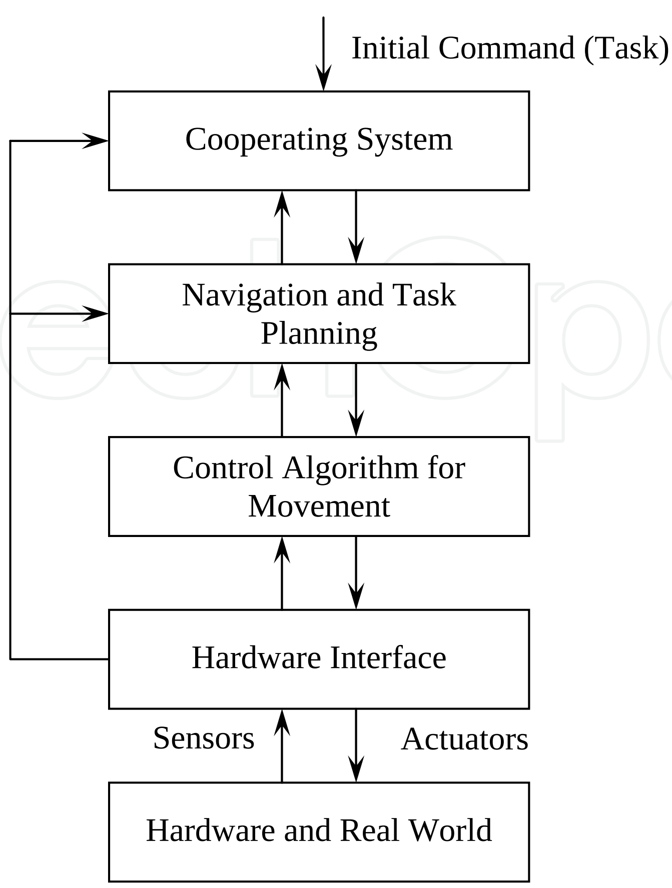


Figure 14. Software hierarchy

The addition to this hierarchy for Itchy and Scratchy is at the top level, the cooperating system which consists of a communications layer and a coordination layer. The communication layer will provide a means of awareness and knowledge sharing amongst the robots and will utilize the wireless LAN. The coordination layer will utilize the shared knowledge so that each robot takes into account the actions of the other robot in order to form a coherent group. This layer also processes the initial command (task) given to the robotic system by the human operator.
The navigation and task planning level involves path planning, i.e. moving in known or unknown environments as well as decomposing the task into subtasks and scheduling them. Below this is the control algorithm for movement which is responsible for controlling the robot's velocity and heading so that the task can be achieved (Lee-Johnson & Carnegie, 2003). The hardware interface provides a link between the hardware and software components and consists of device drivers such as drivers for the steering and drive motors. The lowest level of the architecture consists of the physical robot sensors and actuators, which we have interfaced to the upper level software through the data acquisition card described in section 5.

12.1 Co-Operative Robotic Control

Co-operative robotic behaviour independent of human intervention is a very active area of mobile robotic research. Ideally, a human should only provide the initial command to the robots that then decide for themselves how to execute the task. For the purposes of completeness, a brief review of cooperative robotic research is included here.
Three major applications of co-operative robotic behaviour are transportation, sensing and foraging. Cooperative transportation involves multiple robots transporting objects from one location to another and has been exhibited in robot soccer teams (Shim et al., 1997, and de la Rosa et al., 1997) as well as in box pushing (Mataric et al., 1995) and object lifting and carrying (Huntsberger et al., 2003). Cooperative sensing develops a group robotic system for map building and localization for navigation and exploration (Sossai et al., 1999 and Yamauchi, 1999). In foraging, groups of robots must pick up objects scattered in the environment (Parker, 1998).
The control or group architecture provides the infrastructure upon which collective behaviours are implemented and determines the capabilities and limitations of the system (Cao et al., 1997). Research in this area addresses issues such as action selection, delegation of authority and control, the communication structure, and heterogeneity versus homogeneity of robots (Arai et al., 2002). One of the key architectural features of a group architecture for mobile robots is whether the system is centralized or decentralized. Centralized architectures are characterized by a single control agent whereas decentralized architectures allow multiple control agents. The decentralized architecture has been the dominant group architecture since it has several inherent advantages over centralized structures. Two types of decentralized architectures include hierarchical architectures and distributed architectures (Cao et al., 1997).
Hierarchical architectures are locally centralized. The agents are independent in carrying out tasks to achieve certain goals but they communicate with a master or host that has a global view of operations and assigns goals to the agents. CEBOT, a hierarchical architecture which consists of a group robotic system that is dynamically reconfigurable, has been simulated (Fukuda & Iritani, 1995). The GOFER architecture which uses a central task planning and scheduling system was used to study distributed problem solving by multiple robots in an indoor environment using traditional AI techniques and was

successfully used with three physical robots (Cao et al., 1997). Cooperative behaviour based on fuzzy logic optimized by micro genetic algorithms for fixed obstacle and multiple robot avoidance in a centrally managed robot system has been simulated in Glorennec (1997). Hierarchical architectures have also been implemented in co-operative robot soccer teams where soccer robots are linked to a host computer system but have distributed control. The host computer may be used as a coach for the team (de la Rosa et al., 1997) or in a more difficult implementation, as a sensor for processing vision data only (Shim et al., 1997). The use of two cooperative robots operating in a master/slave configuration to facilitate localization and mapping has also been studied (Sossai et al., 1999). Hierarchical architecture implementations, in general, require an additional computer system to act as the master or host. This means additional costs and also leads to the implementation not being robust in the event of this primary device failing.

Distributed architecture implementations remove the need for a master or host. In distributed architectures, all agents have equal control and hence are completely autonomous in the decisional process with respect to each other. A practical application based on a distributed architecture is map building for exploration in an unknown environment using two co-operative robots (Yamauchi, 1999). In this application the robots share perceptual information but maintain separate maps and make independent decisions which leads to the system being robust to the loss of communication between them as well as to the loss of a robot. A distributed system carrying out a box pushing task using explicit communication for coordination has been shown to perform more effectively than a single robot or two non-communicating robots (Mataric et al., 1995). A cooperative box pushing mission by two heterogeneous robots has been achieved using a fully distributed system at both the individual robot and team levels based on the ALLIANCE architecture. This architecture has also been implemented on a physical robot team performing a laboratory version of a hazardous waste cleanup (Parker, 1998). The ALLIANCE architecture has the advantage of using adaptive actions to achieve fault tolerant control within small to medium sized teams of heterogeneous robots. The ABBA architecture, which is designed for distributed cooperative planning, has been utilized in the implementation of a cooperative cleaning task with two autonomous mobile robots (Jung & Zelinsky, 1999). This implementation has also shown the advantage of robustness in the face of failures. CAMPOUT, a distributed control architecture for tightly coupled coordination of multiple robot systems is being developed at the Jet Propulsion Laboratory (Huntsberger et al., 2003). It has been applied to ongoing physical experiments involving the exploration of cliff faces and the deployment of extended payloads.

An approach to distributed coordination in a heterogeneous multiple robot system based on dynamic role assignment is described in Iocchi et al., (2003). The robots are heterogeneous in individual robot control architectures as well as in physical nature. A programming environment called ETHNOS is used to implement the different architectures as well as the communication layer. The distributed coordination protocol has shown robustness to communication failures.

Itchy and Scratchy will implement a decentralized distributed group architecture due to the advantages outlined above. Their initial task will be one of cooperative mapping of an unknown environment. At present the basic control of the individual robot movements are based on PID control laws. A behaviour-based architecture (reactive system) (Brooks, 1986) for the individual robots is being constructed for the coordination layer using neural networks and fuzzy logic. (Mataric et al., 1995, Huntsberger et al., 2003, Parker, 1998 and Jung & Zelinsky, 1999) have also used behaviour-based approaches.

13. Future Work

The reduction in the range of the infrared sensors in bright sunlight indicates that it is advisable to include additional obstacle avoidance sensors. Laser range finders are an attractive solution; however they tend to be expensive, requiring either high speed electronics or a dedicated digital camera system (often with an associated frame grabber card). A simple alternative is the incorporation of ultrasonic sensors. These can reliably project out to a distance of several meters, and detect obstacles that the robot is likely to encounter. By narrowing the transmitted beam, multipath reflections can be reduced, and simple time-of-flight calculations can easily yield the robot-to-obstacle distance.

Accurate localisation of the robots will need further development. The several meter accuracy of the GPS units is not adequate for fine positioning of the robots. Whilst infrareds and/or ultrasonics can provide accurate *relative* positioning once the robots are closer than 5 meters apart, it would be desirable to have sub-meter absolute positioning. This could be achieved with the purchasing of more expensive GPS modules (US$10K), this would negate the low-cost emphasis of this construction. Odometry and inertial sensing are accurate over short distances, so it is anticipated that use will be made of landmarks to reset the accumulated odometry error, and provide for more accurate localisation. Unfortunately, this constrains the robots to operating in a known environment, and is not an optimal solution.

Finally, the manipulator arm obviously needs to be designed and mounted on the robots. This is a significant task, and is not the focus of this article.

14. Conclusion

Two identical robots, “Itchy” and “Scratchy” have been constructed at low cost (below US$2000). They are mechanically and electronically complete, though awaiting the future implementation of a manipulator arm. Weighing 58 kg, they are capable of supporting an external payload of 80 kg, and can operate on their contained power supply for times in excess of 1 hour. The devices are very manoeuvrable, capable of turning within a 1.5 m enclosure.

The data from the shaft encoders provides a resolution of 1.245 mm, and accuracy within 1% on outdoor terrain. The heading calculated from the shaft encoder data alone is insufficient for localisation, but combined with the recently installed compass provides a more accurate dead reckoning navigation system. The GPS receiver is capable of providing an absolute position with standard deviations of 1.68 m and 2.86 m in the longitude and latitude directions when using pseudo-range corrections. For applications where the mechatrons are distant from one another, this can be used for initial localisation. The motors are controlled by a microcontroller interfaced to the PC.

The PC permits the use of advanced software tools, and communication amongst themselves or with a base station via a wireless LAN card. The operator can directly control the mechatrons from the onboard computer through a virtual desktop using Windows XP terminal server, or using the remote base software written. The remote software allows a joystick to be used on the local machine to ease control of moving the mechatron.

The hierarchical software system used successfully by the authors for other mobile robots is being extended to facilitate human independent co-operative interaction between Itchy and Scratchy.

Acknowledgements

The authors would like to express their gratitude to the Automation Systems Division of Industrial Research Ltd. (Parnell), and to Bruce Rhodes and Scott Forbes of the Department of Physics and Electronic Engineering at the University of Waikato.

15. References

Arai, T., Pagello, E., and Parker, L.E., "Editorial: Advances in multi-robot systems", *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 665-661, 2002.

Borenstein, J., and Feng, L., "Measurement and correction of systematic odometry errors in mobile Robots", *IEEE Transactions on Robotics and Automation*, 1996, **12**(6).

Brooks, R., "A robust layered control system for a mobile robot", *IEEE Journal of Robotics and Automation*, vol. 2, no. 1, pp. 14-23, 1986.

Cao, Y.U., Fukunaga, A.S., and Kahng, A.B., "Cooperative mobile robotics: antecedents and directions", *Autonomous Robots*, vol. 4, no. 1, pp. 7-27, 1997.

Carnegie, D.A., Towards a fleet of autonomous mobile mechatrons. In *Proceedings of the International Federation of Automatic Control Second Conference on Mechatronics Systems*, Berkeley, 9-11 December 2002, pp 673-678.

Carnegie, D., Loughnane, D.L., and Hurd S.A., "The design of a mobile autonomous robot for indoor security applications", *Proc. Instn Mech. Engrs Part B: J. Engineering Manufacture*, 2004, **218**(B5), 533 – 543.

Cordes, J.C., and Carnegie, D.A., The Creation of an autonomous multi-terrain mechatron. In *Proceedings of the Ninth New Zealand Electronics Conference*, Dunedin, New Zealand, 14 – 15 November 2002, pp 49 – 54

de la Rosa, J.L., Oller, A., Vehi, J., and Puyol, J., "Soccer team based on agent-oriented programming", *Robotics and Autonomous Systems*, vol. 21, no. 2, pp. 167-176, 1997.

Fukuda, T., and Iritani, G., "Construction mechanism of group behaviour with cooperation", in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, USA, 1995, pp. 535-542.

Glorennec, P., "Coordination between autonomous robots", *International Journal of Approximate Reasoning*, vol. 17, no. 4, pp. 433-446, 1997.

Huntsberger, T., Pirjanian, P., Trebi-Ollennu, A., Nayar, H.D., Aghazarian, H., Ganino, A.J., Garrett, M., Joshi, S.S., and Schenker, P., "CAMPOUT: A control architecture for rightly coupled coordination of multirobot systems for planetary surface exploration", *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 33, no. 5, pp. 550-559, 2003.

Iocchi, L., Nardi, D., Piaggio, M., and Sgorbissa, A., "Distributed coordination in heterogeneous multi-robot systems", *Autonomous Robots*, vol. 15, no. 2, pp. 155-168, 2003.

Jung, D., and Zelinsky, A., "An architecture for distributed cooperative planning in a behaviour-based multi-robot system", *Robotics and Autonomous Systems*, vol. 26, no. 2-3, pp. 149-174, 1999.

Lee-Johnson, C., and Carnegie, D., "The development of a control system for an autonomous mobile robot". In *Proceedings of the Tenth New Zealand Electronics Conference*, Hamilton, New Zealand, 1 -2 September, 2003, pp 77 – 82.

Payne, A.D., and Carnegie, D.A., "Design and construction of a pair of tricycle based robots to investigate cooperative robotic interaction", In *Proceedings of the Tenth*

Electronics, New Zealand Conference, Hamilton, New Zealand, 1 – 2 September 2003, pp 53 – 58

Mataric, M.J., Nilsson, M., and Simsarian, K.T., "Cooperative multi-robot box-pushing", in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, USA, 1995, pp. 556-561.

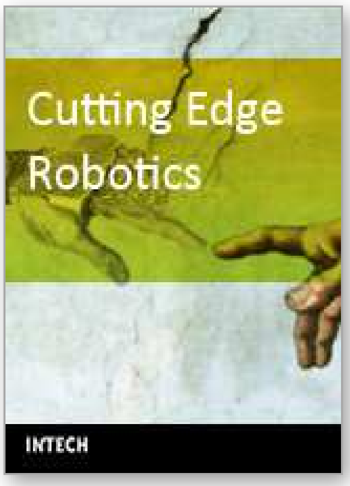
Parker, L.E., "ALLIANCE: An architecture for fault tolerant multirobot cooperation", *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, pp. 220-240, 1998.

Shim, H.S., Kim, H.S., Jung, M.J., Choi, I.H., Kim, J.H., and Kim, J.O., "Designing distributed control architecture for cooperative multi-agent system and its real-time application to soccer robot", *Robotics and Autonomous Systems*, vol. 21, no. 2, pp. 149-165, 1997.

Sossai, C., Bison, P., Chemello, G., and Trainito, G., "Sensor fusion for localization using possibility theory", *Control Engineering Practice*, vol. 7, no. 6, pp. 773-782, 1999.

Victorino, A.C., Rivers, P., and Borrelly, J.J., Localization and map building using a sensor-based control strategy. In *Proceedings of the 2000 IEEE International Conference on Intelligent Robots and Systems*, Takamatsu, Japan, October 2000.

Yamauchi, B., "Decentralized coordination for multirobot exploration", *Robotics and Autonomous Systems*, vol. 25, no. 2-3, pp. 111-118, 1999.



Cutting Edge Robotics

Edited by Vedran Kordic, Aleksandar Lazinica and Munir Merdan

ISBN 3-86611-038-3

Hard cover, 784 pages

Publisher Pro Literatur Verlag, Germany

Published online 01, July, 2005

Published in print edition July, 2005

This book is the result of inspirations and contributions from many researchers worldwide. It presents a collection of wide range research results of robotics scientific community. Various aspects of current research in robotics area are explored and discussed. The book begins with researches in robot modelling & design, in which different approaches in kinematical, dynamical and other design issues of mobile robots are discussed. Second chapter deals with various sensor systems, but the major part of the chapter is devoted to robotic vision systems. Chapter III is devoted to robot navigation and presents different navigation architectures. The chapter IV is devoted to research on adaptive and learning systems in mobile robots area. The chapter V speaks about different application areas of multi-robot systems. Other emerging field is discussed in chapter VI - the human- robot interaction. Chapter VII gives a great tutorial on legged robot systems and one research overview on design of a humanoid robot.The different examples of service robots are showed in chapter VIII. Chapter IX is oriented to industrial robots, i.e. robot manipulators. Different mechatronic systems oriented on robotics are explored in the last chapter of the book.