Development of a Control System for a Skid-Steer Amphibious Vehicle

Clarke, Michael; Blanchard, Tom

Publication date:
2010

Citation for published version (APA):
Clarke, M., & Blanchard, T. (2010). *Development of a Control System for a Skid-Steer Amphibious Vehicle*. 41.
http://hdl.handle.net/2160/5312

Development of a Control System for a Skid-Steer Amphibious Vehicle

Michael Clarke
Intelligent Robotics Group
Computer Science Department
Aberystwyth University
Aberystwyth, SY23 3DB
Email: mfc5@aber.ac.uk

Tom Blanchard
Intelligent Robotics Group
Computer Science Department
Aberystwyth University
Aberystwyth, SY23 3DB
Email: ttb7@aber.ac.uk

Abstract— **With the increase in use of autonomous robotic systems for various tasks, including research projects, their ability to operate away from their ideal laboratory environment comes into question. Whilst long term autonomous robotic platforms do exist in various forms, few provide all-terrain manoeuvrability, particularly with skid-steering mechanisms. This paper aims to show the development of an all-terrain robotic platform based on an ARGO 6x6 amphibious petrol-powered skid-steer vehicle. The design and development of a control system, including software and hardware, is discussed. It is shown that the control of such a vehicle is possible with a reasonably high level of precision and with relatively low cost components, suggesting that with more expensive hardware or different control techniques, high-precision control might be possible. Further, it is shown that based on the various characteristics of the robotic platform such as throttle values, wheel rotations and turn rates it is possible to classify, with some level of accuracy, the varying surfaces allowing for future work towards adaptive control systems with the ability to adjust autonomously their tuning parameters, and thus performance, as the robot moves from one unknown environment to another.**

Index Terms— **Autonomous Control, GPS Navigation, Surface Type Detection**

I. INTRODUCTION

WHILST many robotic platforms exist in static stationary environments (such as those performing automated assembly in factories), requiring only limited methods of identifying and adapting to changing operating environments, another breed, field robotics, exist which are increasingly required to cater for dynamic environments. Such robots are often required to operate autonomously for prolonged periods of time in challenging, diverse and dynamic environments. Despite many advances and the plethora of autonomous robots in the world, many are still restricted to limited, idealised, environments. The result is that such robots often fail to be truly versatile outside of their laboratory, lacking the required ability to adapt to environments which change around them.

Numerous attempts have been made to address the problems associated with dynamic environments from mapping techniques to the use of laser scanners and various other sensing equipment [3], [1] and yet it continues to be a restrictive aspect. This paper details attempts to design and develop a control system that provides a solution to this real world problem, using an ARGO 6x6 off-road, skid-steer, amphibious vehicle as the base for this all-terrain robotic platform. Of significant interest is the level of complexity required, both in terms of hardware and software, in order for the resulting control system to deal adequately with the challenges of repeatably and accurately turning by a set number of degrees in varying and unpredictable conditions.

Further to the issues already discussed, the choice of vehicle brings its own challenges, and some advantages. Due to its skid-steer nature the ARGO is relatively imprecise when driven by humans - gyroscope data gathered during initial human driven testing of the vehicle showed significant overshoot in turn rate when attempting to come to a controlled stop at the end of a turn. Further, the use of a petrol engine provides additional challenges - particularly with relation to response times, vibrations and electromagnetic interference. However, the amphibious nature of the vehicle, and the inclusion of 6-wheel drive, provides an excellent platform for the exploration of the performance of the resulting robotic platform on a variety of surfaces.

Ultimately, this paper attempts to answer a number of questions relating to the resulting control system:

1) Can a vehicle such as an ARGO be automated to provide precise and repeatable control and, in particular, what complexity of software and hardware components are required?
2) What effect do different surfaces have on the performance, accuracy and repeatability of the resulting vehicle?
3) Given the observed characteristics of the vehicle, is it possible to identify the surface type and therefore adjust autonomously control and tuning parameters to gain best performance?

The remainder of this paper is structured as follows: Section II details the design and implementation of the resulting control system, Section III gives light to the main control algorithms used, Section IV provides a detailed analysis of the results and Section V draws conclusions and provides suggestions for possible future work.

II. SYSTEM DESIGN & IMPLEMENTATION

The base platform for the robotic system is an ARGO 6x6 Frontier 580 (Fig. 1); an amphibious skid-steer vehicle with a 18bhp twin-cylinder Briggs & Stratton engine capable of achieving a top speed of approximately 22mph on land and 3.5mph in water. The vehicle has a manual gear box for selecting high/low range or reverse gears whilst the clutch is automatic. The power from the engine is transmitted to the wheels via drive chains on the left and right sides of the engine. Each set of chains is connected to a simple disc brake which allows for braking or locking of one complete set of wheels. As such, the vehicle is able to skid and therefore turn around the centre wheel; however, it should be noted that the vehicle is not able to turn directly on the spot. The handlebars then connect to these brake discs and as such turning the handlebars left or right applies pressure to the left or right brake discs. Further, the handlebars provide a grip and twist style of throttle control for the operator with a direct mechanical linkage to the governor.

Fig. 1. The ARGO 6x6 Frontier 580 Amphibious Skid-Steer Vehicle

A. Vehicle Modifications

Various modifications were required in order to allow computer control and ultimately automation of the vehicle. In particular, control of the engine speed was required and to achieve this the governor was disconnected from the carburettor in the engine. An high torque electronic servo was then mounted in its place (Fig. 2) allowing for the direct manipulation of the air/fuel mixture in the engine, and as such control of the throttle.

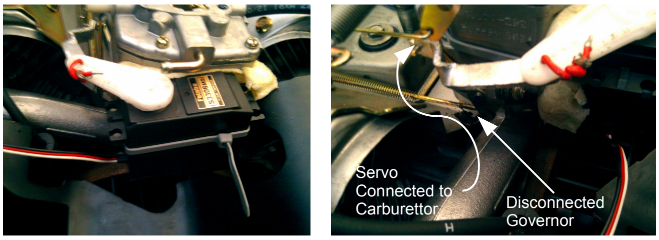

Fig. 2. Simple modifications to the engine to allow control of the throttle via computer.

Further, a method of steering the vehicle was also required. Two options existed, the first a modification to the brake calipers such that a control system would be able to manipulate directly the pressure to each brake disc. However, the complexities of such an operation for a first prototype seemed unnecessary when another much simpler option, the mechanical control of the handlebars, was available. As such a LINAK LA12 linear actuator was attached to the vehicle and then bolted to the handlebars (Fig. 3). This allowed for the manipulation of the handlebars and thus the ability to steer the vehicle.

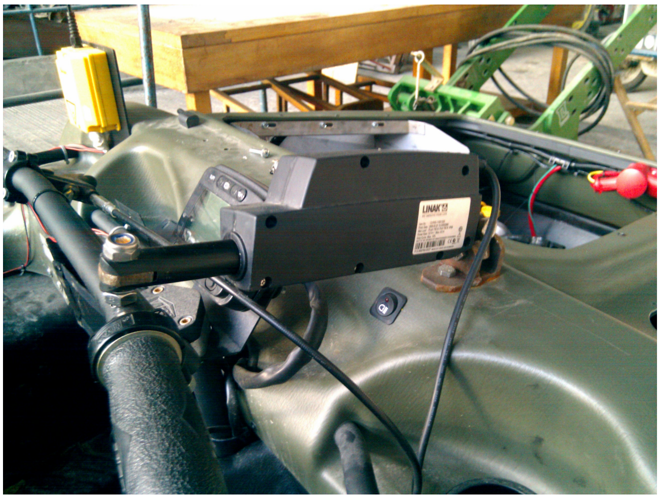

Fig. 3. The LINKAK LA12 linear actuator bolted to the ARGO handlebars allowing for their manipulation via computer.

Using this approach significantly simplified the hardware requirements of the robotic platform but did introduce the issue of speed. The actuator would take approximately 2 seconds to turn the handlebars from a left-hand full lock to a right-hand full lock. This meant that later, during the development of the software, there were real-time restrictions on the steering that could potentially have been avoided using a more complicated computer controlled hydraulic system capable of directly manipulating brake pressure to each caliper.

It was decided that in order to maintain simplicity the control of the gear stick would be performed by a human operator; given that once the vehicle was in gear the automatic clutch would deal with any further complications. Whilst limiting the autonomous nature of the vehicle slightly (eg. preventing the selection of reverse gear) it was not expected to be detrimental to the performance of the vehicle in any way, and as such it simply became part of the set-up procedure to put the vehicle into gear before allowing it to perform autonomous operations.

In addition to the modifications required for control it was clear from an early stage that having an indication of wheel rotations would be of significant use - both as an additional surface characteristic and to help monitor the rotation speed of the wheels during differing manoeuvres. As such, two small right-angled brackets were produced to hold magnetic proximity sensors (Fig. 4). These brackets were then mounted in such a way that the proximity sensors were close enough to the drive chains between the engine transmission and wheels, allowing for the counting of chain links. It was then calculated that seven pulses (or chain links) were equivalent to a single full wheel rotation.

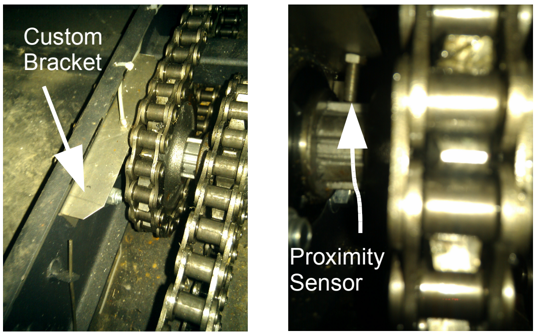

Fig. 4. Custom made brackets, mounted such that the proximity sensors can detect and count wheel rotations.

A final modification to the vehicle included the addition of a radio-control safety shut-off switch. This was wired directly into the ignition, allowing for the remote power-down of the engine.

B. *Hardware Architecture*

A custom board was developed for the robotic platform including a number of sensors and computer hardware components. Specifically, the board was comprised of a non-tilt-compensated compass, a PIC 18F4550, a Gumstix Verdex, a 3-axis accelerometer, a GPS module and a 2-axis gyroscope.

The accelerometer and gyroscope, connected directly to the analogue digital converter (ADC) on the PIC, were primarily expected to be used for surface characteristic identification. Whilst they had the potential to provide data such as the turn rate of the vehicle during manoeuvres the authors felt that this would be an over complication of the control system with other sensor information, such as the compass data, considered to be more valuable for this use. As such, the data from the accelerometer and gyroscope, whilst logged for analysis, was not expected to be used directly for control of the vehicle.

The compass was connected to the Gumstix via the i2c bus. The compass was a vital component of the control system, being the primary method of identifying the heading of the vehicle. Further details regarding the specific uses of the compass with relation to control of the vehicle and the software algorithms used can be found in Section III.

The steering actuator holds a potentiometer in order to measure the extension of the actuator arm at any point. This potentiometer also connects to the ADC on the PIC. Finally, the wheel rotation sensors, previously discussed, connect directly to two I/O pins on the PIC.

In order to help maintain the real-time requirements of the system (such as counting wheel rotations) it was intended that the PIC should act solely as a sensor gathering system whilst the Gumstix would retrieve that formatted and processed sensor data from the PIC via the serial line and then make appropriate control decisions. As such, data from the PIC is sent to the Gumstix via a 19,200 baud rate serial line. Further, the GPS module also communicates with the Gumstix via a 115,200 baud rate serial line. These were both the maximum supported baud rates for the relative components.

To further help achieve the division of labour, and to allow the software to concern itself only with the values that should be used for the throttle and steering positions (rather than how to achieve them), additional hardware components were used in a number of situations where software would have sufficed. Particularly, a Polulu servo controller board was used for the control of the throttle servo and steering actuator (with the addition of a simple motor controller). An overview of the final hardware implementation is shown (Fig. 5).

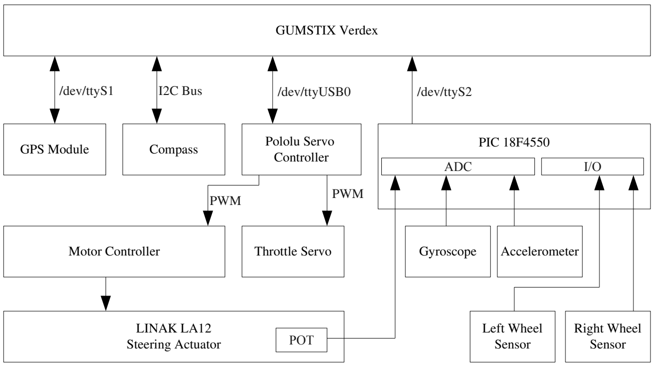

Fig. 5. Block Diagram Overview of the Hardware Components in the System

C. *Software Architecture*

In line with the hardware architecture, the software was divided into two distinct applications. The first, which runs on the PIC, is a relatively simplistic control loop. For every iteration of the loop the frequency of wheel rotations is calculated and all sensor, accelerometer and gyroscope data is gathered. This data is then concatenated into a single string and transmitted down the serial line to the Gumstix. In order to ensure synchronisation of the data being transmitted a start sequence '::' and string termination character '\n' is added to each string before transmission.

The more interesting component of the control system is the software running on the Gumstix. This software is required to perform three jobs. Firstly, it provides a number of interfaces for controlling the servos and actuators and for retrieving the data from the PIC. This, in turn, provides the various control loops with a simple method of setting throttle speeds, handlebar positions, etc. The second task that the software performs is to log continually data such as the sensor data from the PIC, current heading, etc. Finally, and perhaps most important, the software implments two algorithms; one for performing turns (to both specific

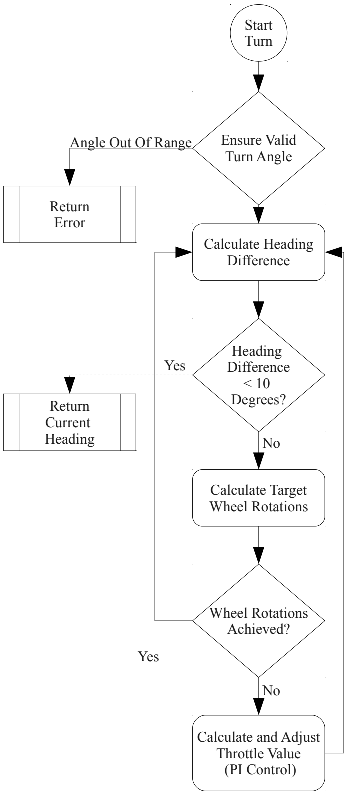

Fig. 6. Flow Chart of the Turn Algorithm

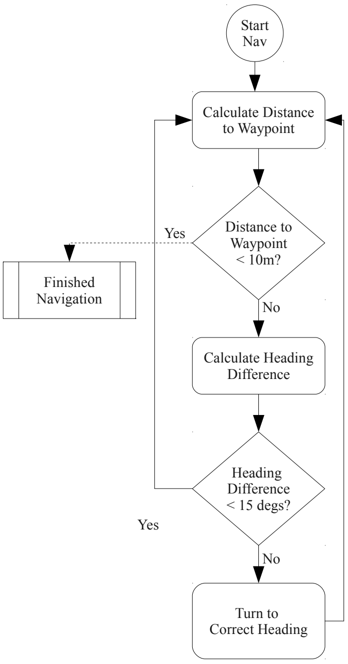

Fig. 7. Flow Chart of the GPS Navigation Algorithm

or relative angles) and another for performing simplistic GPS navigation. The implementation of these algorithms is discussed in further detail in Section III.

III. CONTROL ALGORITHMS

Two control algorithms have been used for the development of the robotic platform, and these are discussed in further detail.

A. Turning Algorithm

During the course of the research the control algorithm used for performing turns was adjusted and modified. Initially, an unreasonably simple algorithm was tried which set the throttle to an arbitrary value and then waited until the desired heading difference was within the specified dead band (10 degrees). The dead band value was chosen due to the limitations of the compass with vibrations from the ARGO introducing 10 - 15 degrees of error.

As expected, this initial approach performed very poorly and generally resulted in the vehicle skidding around continuously never achieving the desired heading. As such an attempt was made to include wheel rotation sensor data and the algorithm was modified such that if the rotations ever exceeded a specific, but once again arbitrary, value the throttle would be reduced to zero. At this point the throttle would start to increase by a value of one in a tight loop until, once again, the wheel rotations exceeded the specified maximum. Again, this resulted in very similar characteristics compared to the first attempts. The response times of the engine proved to be such that by the time the engine had reduced speed due to the reduced throttle value, the throttle was once again fully open. In effect the throttle therefore remained constantly open.

To combat this it became apparent that a significantly more sophisticated approach would be required. Instead of simply reducing the throttle to zero, it was clear that a proportional mapping between throttle position, wheel rotations and the current heading difference would be required and as a result a simple P controller was implemented. This showed significant improvements, but was prone to steady-state-errors. As such, the P controller was modified to incorporate the integral of the error, resulting in a PI controller. The final algorithm is shown in flow chart form (Fig. 6).

B. GPS Navigation Algorithm

Whilst many sophisticated algorithms exist for the navigation of robotic systems such as D* [4] which uses both global and local path planning techniques to ensure safe and efficient arrival at a target position, avoiding obstacles en route, the use of such would have been out of scope for this paper. All that was required was a simple mechanism to show autonomous operation, particularly including the use of the discussed turning algorithm, to allow for the gathering and analysis of data relating to surface type and operating environment characteristics.

As a result of this, an exceedingly primitive GPS navigation algorithm was developed as shown (Fig. 7). The algorithm continuously checks that the robot is on course. If at any point the ARGO is more than 15 degrees off course the algorithm will use the turning algorithm to adjust course.

As with the turning algorithm the 15 degrees dead band for the heading was based on the compass error.

Having ensured the vehicle is on course the algorithm simply sets the throttle to 20% (approximately walking pace) until the desired GPS way point has been reached or the vehicle is once again off course.

IV. RESULTS

After development of the control system a number of initial automated tests were conducted involving turns to pre-defined angles (0, 90, 180, 270) and (0, 45, 90, 135... 315) on gravel. The resulting data was used to tune the PI controller on gravel (the process of which is out of the scope for this paper). The resulting parameters used were 0.0075 for P and 10.0 for I.

After tuning, identical sets of tests were conducted (to the same pre-defined angles noted above) on three different surfaces (grass, tarmac and gravel) and all accelerometer, gyroscope, compass heading, wheel rotations and throttle values were logged. In addition, target and final compass heading data was recorded for each turn. Initial analysis of the data showed that 75% of turns were within the 10 degrees dead band across all turn distances and all surfaces.

This immediately showed that the first objective relating to the ability to control such a vehicle as the ARGO has been conclusively answered. The control of such a vehicle is not only possible, but is so with primitive, cheap equipment and simplistic software control methods. Further the results show great promise for future enhancements. Whilst there is clear evidence that the resulting control system has the ability to perform with reasonable levels of accuracy on gravel for which it was tuned, it is was interesting to see similar levels of accuracy on the other surfaces with no tuning adjustments. As such it was felt that, if more accurately tuned on the other surfaces the results could prove to be even more accurate. However, in order to achieve any form of automated tuning it was still required that there is a concrete method of identification and classification of surface types. As such, further analysis took place in an attempt to identify specific characteristics of surface types.

Whilst all surfaces seemed to exhibit similar accuracy, the manner in which each desired heading was achieved was considerably different. Data gathered and plotted (Fig. 8, 9 and 10) shows significant variations in the performance across the three separate surfaces, even for smaller 45 degree turns. Further, as the turn distances increased these variations were further exaggerated.

Fig. 8 clearly shows that on gravel the wheel rotations (in the direction of travel) had considerable variation. Over the course of the full turn the wheel rotations fluctuated considerably with sudden, apparently random, increases. This was in line with the observed characteristics where the ARGO would often get ‘stuck’ on loose gravel. At this point very little throttle would cause the wheels to spin with great exuberance, followed immediately by a reduction in throttle. With the throttle decreasing the wheel rotation rate would reduce enough such that the wheels would regain grip and the ARGO would continue its turn.

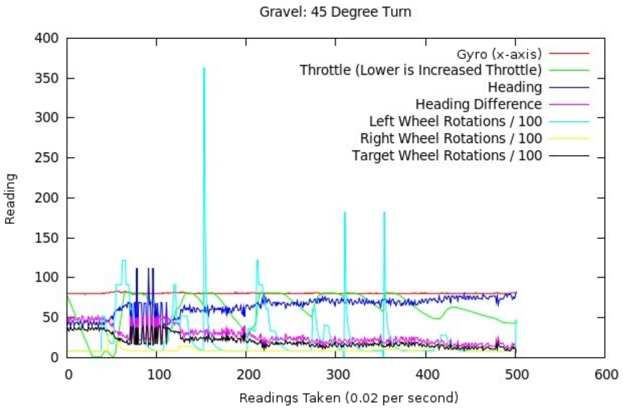

Fig. 8. Data for a 45 degree turn on gravel.

Of particular interest in the data gathered are the approximate readings 50 - 100. These fluctuation in target wheel rotations, compass heading and heading difference were due to significant vibrations from the vehicle. This in turn added to the compass error, resulting in a fluctuating compass value. These invalid values were then used by the turn algorithm to calculate inappropriate target wheel rotations and heading differences. Whilst not evident in the data shown, similar characteristics were seen across many of the turns on all the surfaces tested.

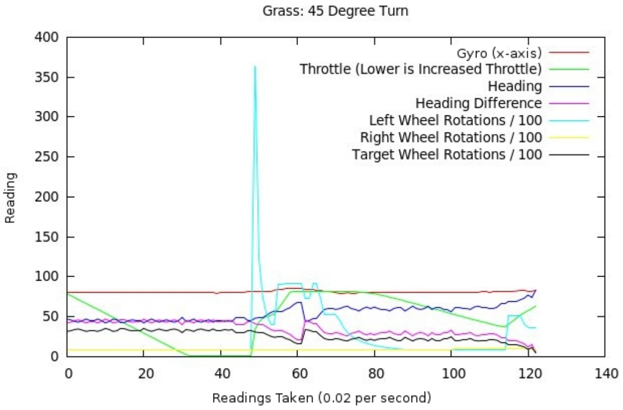

Fig. 9. Data for a 45 degree turn on grass.

Unlike with gravel, data collected for grass (Fig. 9) shows no signs of ‘random’ increases in wheel rotations. The data collected for grass shows a slow and progressive increase in throttle value with no movement of the vehicle. Eventually, at approximately 40 readings, the throttle reaches the maximum value and this is sufficient for the vehicle to overcome the friction provided by grass. At this point, the wheel rotations peak, and the vehicle starts a relatively rapid change in heading. The throttle immediately reduces due to the increased wheel rotations, and by the time it reaches its minimum the vehicle is within the dead band.

As with gravel, the data collected accurately represents the observed characteristics of the vehicle. It was clear during testing that the throttle continually increased before any movement occurred. Once movement did occur, it was rapid and sudden.

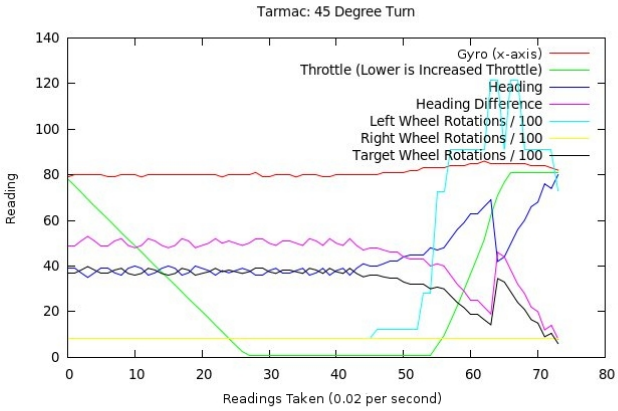

Fig. 10. Data for a 45 degree turn on tarmac.

Finally, Fig. 10 shows data for a 45 degree turn on tarmac. As with grass, the data for tarmac shows a slow and progressive increase in the throttle with no movement. Eventually, the throttle increases to the maximum and the vehicle starts to rotate. Unlike with grass, the wheel rotations seem to increase progressively. This, once again, was in line with the observed characteristics. It was concluded that such characteristics were due to the friction caused by gravel on the tyres. Having achieved sufficient throttle to overcome friction, it had to continue to struggle to complete the turn.

Although not directly shown, data collected for the other turns exhibited similar properties for all three surfaces, indicating there are definitely aspects of the different surfaces that have an impact on the control system parameters. Such data could therefore be used to identify the surface type and as such allow autonomous adjustments to the control system tuning.

At the time of writing there was only limited time for GPS navigation testing, and as such no concrete data was collected. However, current observations suggest that the algorithm developed works well but tends to have a spiral effect inwards towards the target way point. It is predicted that this is due to the significant error from the compass heading.

V. CONCLUSIONS

This paper has shown the development of a fully autonomous robotic platform based on an ARGO 6x6 Frontier vehicle capable of operating with reasonable levels of accuracy on varying surfaces despite issues relating to compass data. We have further shown that the sensor data gathered was more than sufficient for the identification of surface type, and it is evident that such data could be used for the autonomous tuning and adjustment of the control system in order to improve accuracy and repeatability. For example, it is clear that on gravel maintaining a slow wheel rotation rate is important in order to maintain grip, even in situations where the control system (as it currently stands) would increase the throttle to achieve a desired turn rate. By using the surface type data the control system would potentially be able to reduce or even ignore the desire for a slightly faster turn rate in favour of a reduced wheel rotation rate, ensuring continuous and consistent turns.

Despite this, further work is needed to improve the accuracy of the turns, and a new set of algorithms are needed which take into account, and act appropriately, on the surface type detection data. In addition, work is also needed to fine tune the GPS navigation techniques and more testing is required. In particular, it is suggested the use of a Visual Compass [2] would be of considerable advantage on such a vehicle where constant and unavoidable vibrations have such a wide and negative impact. Additional future enhancements might include the direct manipulation of the brake calipers rather than the steering handlebars. Such modifications would allow more rapid changes in heading, and perhaps the opportunity to ‘turn back’ after the overshoot of a desired heading.

Although more testing is also required in order to validate the results in water and on more surfaces than the three detailed in the paper, the results gathered to date show that with minor enhancements to the control algorithms and with autonomous tuning based on the surface type classifications a robotic platform, such as the ARGO, would be more than capable of adjusting its performance to any surface environment.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. Mark Neal, leader of the Intelligent Robotics Group at Aberystwyth University, for his time, patience and supervision. Without his help and guidance this work would not have been possible.

REFERENCES

[1] CM Gifford et al. A novel low-cost, limited-resource approach to autonomous multi-robot exploration and mapping. *Robotics and Autonomous Systems*, 58(2):186–202, 2010.

[2] Frederic Labrosse. Short and long-range visual navigation using warped panoramic images. *Robotics and Autonomous Systems*, 55(9):675–684, 2007.

[3] Feng Lu and Evangelos Milios. Robot pose estimation in unknown environments by matching 2d range scans. *J. Intell. Robotics Syst.*, 18(3):249–275, 1997.

[4] Sanjiv Singh, Reid Simmons, Trey Smith, Anthony (Tony) Stentz, Vandi Verma, Alex Yahja, and Kurt Schwehr. Recent progress in local and global traversability for planetary rovers. In *Proceedings of the IEEE International Conference on Robotics and Automation, 2000*. IEEE, May 2000.