

The Investigation of Mobile Robot Waypoint Navigation Utilising the Global Positioning System and Aerial Imagery

,....

by

Sultan Shair

A Doctoral Thesis submitted in partial fulfilment of the requirements

for the award of Doctor of Philosophy (PhD)

of Loughborough University

(October 2008)

© by (Sultan Shair) (2008)

.

Abstract

Problems faced by UK farmers formed the *raison d'être* for this research. Agriculture plays an important role in the UK economy. However, due to difficulties securing the necessary workforce, problems associated with hiring migrant labourers, and competition from cheap imported produce, the UK's Gross Value Added (GVA) plummeted from £9.8 to £5.2 billion in the decade leading up to 2005. As a result, the automation of certain farming (and horticultural) jobs is becoming a desirable alternative to man-power, in an aim to re-establish the supply and demand for local and exported produce.

The need for low-cost, robust and manoeuvrable robots to attract farmers' interest in agricultural/horticultural automation was met by the introduction of Ransomes Jacobsen's $Spider^{\circledast}$, a grass cutting mower that could be converted into an autonomous robot. With the appropriate transformation in hardware and computer software, it has provided the opportunity for such research.

In recent years, there has been a burgeoning interest in precision farming for applications such as crop monitoring, using aerial images to identify and assess large land areas. This interest has led to the investigation of mobile robot waypoint navigation utilising the Global Positioning System (GPS) and aerial imagery.

Novel explorations were carried out on waypoint selection from aerial images; the study and improvement of the current GPS positional output; the implementation of a two-stage fuzzy guided controller based on GPS accuracy criterions; a unique heading control strategy; the adaptation of the circular stages of closeness model for the waypoint and GPS positional interaction and the integration of these studies into a simulation.

The results have shown that the error of the waypoint selected from the aerial image to its true ground position ranged from 0.087 m - 0.732 m; a 94% improvement in waypoints reached using corrected GPS data, yielding positional accuracy to within less than 0.6 m; and successful waypoint navigation using the fuzzy control strategy and the circular stages of closeness model.

Acknowledgements

First and foremost I would like to dedicate this thesis to **Diana Shillitoe** for having supported me from day one until the completion of this thesis. Without her endless support and emotional commitment throughout these three years I am not sure if I would have finished. I am eternally grateful to her and am looking forward to sharing the success of our achievement.

Also, I would like to dedicate my work to **my Parents** (Usama and Petra) who have made me the person who I am today, and without whom I never would have reached this academic achievement. Thanks to them for all of the support they have given me since my kindergarten graduation.

Thank you to my sister **Mona** for her support and continuous encouragement and for always trying to provide me with tips and tricks on stress reduction.

A huge thank you to my German and Jordanian family who have given me the best of both worlds.

A very big thank you to my supervisor **Professor Rob Parkin**, for guiding me throughout these three years and trusting my judgment during times when I thought I had lost it, and for entrusting me with responsibilities. I have learnt a great deal about myself and my capabilities thanks to him.

A tremendous thank you to my friend and colleague **Dr Victor Javier Gonzalez-Villela**, who has helped me throughout my PhD, and has provided me with great assistance during the time of the robot modelling and in our collaborative work throughout.

Thank you to my co-supervisor and research coordinator **Professor Mike Jackson** for his assistance during this journey and for encouragement and motivation.

A big thank you to **Dr Jim Chandler** from the department of Civil Engineering, who has taken time out of his very busy schedule to help me complete the publication of our paper, and has helped to build my knowledge of the complex world of photogrammetry.

A very big thank you to the technicians **Dave Liddell** and **Terry Harrison**, who have taught me the tricks of the trade which I would not have learnt without their skilful handson approach. Thanks for being so patient and tolerant.

To my friends at the Mechatronics Research Group (MRG):

Vianney Lara Prieto: Thank you for always helping me take the robot and all the equipment out of the lab and into the parking lot, in addition to holding my laptop during the testing. Your assistance has helped me throughout my last two years, and it was always great talking to you about technical/theoretical issues. I have also learned to become a more patient person!

Sedat Elmas: Thank you for always being there to discuss my project and its quirks and for helping me out marking the parking lot with chalk lines, and for helping me with the surveying of the waypoints. Your help has been invaluable.

Brian Hede: Thank you for your technical support with Visual Basic, Mikrobasic and Matlab. I would probably not have finished on time without your experience and practical help in achieving fast and effective solutions. For me you will always be the VB wizard! Also, thank you for your tips, tricks and advice on the completion of my PhD.

Senthan Mathavan: Thank you for helping me out during the times of the modelling and for your constructive viewpoints on some issues. You have been a great guidance and I am grateful for your help.

Pauli Komi: Thank you for your advice and experience in finishing this degree and for always sharing humorous links and stories. Purkula!

Nashtara Islam: Thank you for tolerating me as your desk neighbour. It was always fun having small chit-chat about the latest news, which was a healthy distraction from continuously staring at the monitor.

Khurram Kamal: Thanks for the support and tips you have given me throughout the years, and for showing me around the lab on my first day of arrival.

Najib Salleh: Thanks for always being willing to play a round of foosball when a break from work was very much needed – it is always to fun to play a game against you.

A great thank you to the students who have worked me on different elements of the project: **Dorit, James, Stephane, Peter** and **Mutaz.** It was a great pleasure working with you.

Thank you also to the recent members and friends at the MRG, Tussanai Parthornratt (Tuss), Niranjankumar Ramasubramaniyan (Niranjan), Hafiz, Matthiew and Phillip.

Last but not least, thank you to the **Civil Engineering department** for the lending of the surveying equipment and a big thank you to the staff of the **Wolfson School** and **Research Student Office**.

Preface

- The thesis is broken down into eight chapters:
 - 1) General Background
 - 2) Literature Review
 - 3) Spider Technical Specifications
 - 4) System's Components Preliminary Test Results
 - 5) Spider and Controller Modelling
 - 6) Experimental and Simulated Results
 - 7) Discussion
 - 8) Conclusion
- At the beginning of each chapter, except the first, there is a small introduction stating to the reader what is to be expected. It usually starts off with one to two opening lines followed by a point format description of the content/breakdown of the chapter. At the end of the bullet points the reader is left with two notes: The first which indicates the chapter pre-requisites, referenced from within the thesis, and the second note describes the nature of the chapter, i.e. practical, theoretical etc...
- Throughout the thesis, following each chapter or a major section, a summary is provided that touches upon the major elements covered.
- A list of the publications completed during the PhD is presented after the References section, and the accepted journal paper is attached.
- There are three appendices. The first covers some of the technical work performed on the Spider, the second shows the models used for the simulation and experimental results and the final appendix shows results from experimental work conducted in Chapter 4.

Table of Contents

1	Chapter 1: General Background2
	1.1 Robots
	1.2 Research Objective
2	Chapter 2: Literature Review 12
	2.1 Robot Mechanisms 13
	2.2 Robot Localisation
	2.3 Path Planning and Navigation
	2.4 Map Representation in Mobile Robots
	2.5 Literature Gap
3	Chapter 3: Spider Technical Specifications 54
	3.1 The Spider
	3.2 Similar Robots (Technical Differences)
	3.3 Spider Transformations (Accessorisation)
	3.4 Robot's Hardware
	3.5 Robot's Software
	3.6 Chapter 3 Summary
4	Chapter 4: System Components – Preliminary Test Results
	4.1 Experimental Tools
	4.2 Aerial Image and Waypoint Accuracy75
	4.3 GPS
	4.4 Chapter Summary
5	Chapter 5: Spider and Controller Modelling92
	5.1 Kinematic Model
	5.2 GPS Error Model 102
	5.3 Waypoint Reaching Controller 106
	5.4 Chapter Summary 124
6	Chapter 6: Experimental and Simulated Results 126
	6.1 Spider Tests A 127
	6.2 Spider Tests B

6	.3 Spider Test C	. 172
6	.4 Chapter Summary	. 180
7	Chapter 7: Discussion	. 182
7	.1 Corrected vs. Non-Corrected GPS Data	. 183
7	.2 The GADF effect on controller	. 198
7	.3 Circular Stages of Closeness	. 206
. 7	.4 The heading control strategy	. 21 1
7	.5 Review of Simulation	. 213
7	.6 Future Work	. 215
8	Chapter 8: Conclusion	. 218
9	References	. 224
10	Publications	. 240
11	Appendix A	. 242
12	Appendix B	. 252
13	Appendix C	. 264

List of Figures

Fig. 1.1.2: a) an industrial robot, b) professional surgical service robot, c) domestic grass
cutting service robot and d) domestic assistant service robot
Fig. 1.1.3: ROBOVOLC (right) and its user interface (left)
Fig. 1.2.2: The Spider, a robust grass mower by Ransomes-Jacobsen Ltd
Fig. 2.3.2a: Autonomous path of the car
Fig. 2.3.2b: Autonomous path of the boat
Fig. 2.3.2c: Aerial image for controlling RATLER robots
Fig. 2.4.2.1a: Vegetation map of test area. The green (light coloured) areas represent no
vegetation and the red (dark coloured) represent high vegetation. The black path represents
the robot's path
Fig. 2.4.2.1b: Traversability map of test area. The green (light coloured) areas represent the
traversable area and the red (dark coloured) areas represent the non-traversable area. The
black path represents the robot's path. This image was created after vegetation filtering39
Fig. 2.4.2.2a: Overhead image, traversability map and traversability grid
Fig. 2.4.2.2b: User interface for controlling URBOT
Fig. 2.4.2.2c: ROBOVOLC waypoint command window
Fig. 2.4.2.2d: (Counter-clockwise from top). Aerial Image, discretised grid, 3D model 44
Fig. 2.4.2.2e: (Counter-clockwise from top) User console, precisely surveyed building
corners, MAKLINK
Fig. 3.1: Ransomes Jacobsen's Spider [Ransomes Jacobsen 2006]
Fig. 3.1.1: The chain and belt drive mechanism of the Spider
Fig. 3.2a: A schematic of their modified Spider, the HortiBot [HortiBot 2007]58
Fig. 3.2b: Spider robot modified by the Robotics team of the Warwick Manufacturing
Group, Warwick University [WMG 2007]
Fig. 3.2c: Seekur TM , the indoor/outdoor all-weather robot [Mobile Robots Inc. 2007] 60
Fig. 3.3a: The rotating platform developed for the Spider (in the initial stages)
Fig. 3.4: Hardware layout split between upper and lower area of the platform
Fig. 3.4.1.1: IENSYS General Purpose microcontroller
Fig. 3.4.1.2: RS232 Bluetooth adapter [Initium 2006]64

Fig. 3.4.2.1: The WAAS enabled Garmin 18-5Hz GPS receiver
Fig. 3.4.2.2: The 360° potentiometer mounted on one of the wheels
Fig. 3.4.2.3a: Optical sensor for providing wheel's translational velocity
Fig. 3.4.2.3b: The Spider's relative optical encoder and potentiometer
Fig 3.4.3: General schematic of the data handling equipment and procedure
Fig. 3.5.1: General system layout
Fig. 3.5.2a: Visual Basic application used for the Spider's control system
Fig.3.5.2b: Screenshot of VB software used for the Spider
Fig. 4.2a: 54 accurately surveyed points using static post data processing
Fig. 4.2b. Images a) and b) show the discrepancy between the user-selected point (dark-
coloured) and the surveyed point (light-coloured)
Fig. 4.2c. Shows that the user selected point from the orthorectified image would require a
positional shift in order to match its true location in the global frame, or create a proximity
error circle to encompass the actual point
Fig. 4.3.1.a: Satellite visibility over a 12 hour period (10 am to 10 pm) at the Holywell car
park. Optimal visibility (>12) shown for approximately 1.5 hours
Fig. 4.3.1.b: Position Dilution of Precision over a 12 hour period (10 am to 10 pm) at the
Holywell car park. Optimal precision shown for approximately 1.5 hours around the 12 pm
margin 80
Fig. 4.3.2: Beacon Hill Ordnance Survey pillar
Fig. 4.3.2.1a: Single point comparison at Beacon Hill's pillar on various days
Fig. 4.3.2.1b: Data collected on 18/10/2007
Fig. 4.3.2.1c: Deviation distance of data from actual point over the predefined test period83
Fig. 4.3.2.1d: Single point comparison at test site showing positional data before and after
correction. (4/11/2007 at robot's test site)
Fig. 4.3.2.2a: The Garmin GPS receiver mounted to the trolley for the traversal tests 85
Fig. 4.3.2.2b: The dark lines indicate the GPS results prior to positional correction, and the
light lines indicate the post positional correction. Test was conducted on 25/10/2007 86
Fig. 4.3.2.2c: 19 waypoints used for testing GPS positional correction approach,
repeatability and accuracy

Fig. 4.3.2.2d: The percentage of waypoints hit for 10 runs before and after positional Fig. 4.3.2.2e: The percentage of waypoints hit for 17 runs before and after positional Fig. 4.3.2.2f: The percentage of waypoints hit for 17 runs before and after positional Fig. 5.1.2: The Spider. Kinematic model of the four-wheeled synchronous drive robot 95 Fig. 5.1.4: Wheel's kinematic constraints - with the orthogonal and along wheel frames. 97 Fig. 5.2a: Dark lines represents the positional data before correction (Chapter 4), and the light ones after the correction for the GPS mounted trolley that was driven along a pre defined series of waypoints. A general fluctuation in the positional data can still be seen. The radius of the waypoint is 0.7 m..... 102 Fig. 5.2b: The GPS position after the positional error correction (Chapter 4) is presented by the small circle. This is arranged such that it would always fall within the boundaries of the Fig. 5.2c: 1) the white line shows the fluctuation in the GPS positional data. 2) the white line shows the simulated fluctuation in the GPS error model......105 Fig. 5.2.d: 1) The black lines indicate the centre of the robot with no GPS induced error. 2) In this image, the white lines represent the position of the GPS induced error and the black lines represent the position of the robot. The controller adjusts according to the GPS signal and therefore the robot can be seen to be off-track. The behaviour of the GPS shows similar traits to those seen in Fig. 5.2a, and the robot is also expected to exhibit similar behaviour. Fig.5.3: The control system layout 106 Fig. 5.3.1a: The waypoint reaching model based on work by Gonzalez, V. et al [2004], in which the full functionality of the synchronous drive's manoeuvrability is utilised...... 107 Fig.5.3.1: b) Shows that the Spider is travelling only in the forward translational velocity when reaching a waypoint, therefore not taking full advantage of the synchronous drive capability; b) shows the Spider after implementing the heading control strategy, where both

Fig 5.3.2a: The circular model representations that are used for the waypoint reaching
control strategy. The Spider error circle is pivoted at the P_{GPS} with a radius of R_A . The
waypoint has a radius of R_B
Fig. 5.3.3a: Fuzzy angular and linear velocity controller using a GPS Accuracy Decision
Factor (GADF)113
Fig. 5.3.3b: Fuzzy rules for GADF using a worst case scenario logic system for combining
PDOP accuracy to the number of satellites
Fig 5.3.3c: This figure shows the second stage of the fuzzy controller, where the (d_c) and
$(\theta_{ef} \operatorname{or} \theta_{eb})$ are used in conjunction with the GADF to control the linear and angular
velocities
Fig. 6.1a: Number of satellites (satellite visibility) from 7:00 -19:00 (DST) 128
Fig.6.1b: Position dilution of precision from 7:00 -19:00 (DST) 128
Fig.6.1.1: Before and after GPS data correction – dark line indicate before and the light one
after correction. The large circle shows RA where 95% of the static data falls within its
boundaries129
Fig.6.1.1.1a: Yellow line shows the Spider's path and the white line shows the GPS
positional data – correction – medium GADF
Fig.6.1.1.1b: The linear velocity, angular velocity and GADF respectively. The vertical
lines represent the waypoints (2 - 19) - correction - medium GADF 132
Fig.6.1.1.1e: Percentage of waypoints hit for test with varying radius size - correction -
medium GADF 134
Fig.6.1.1.2a: Simulation of the Spider in Matlab under similar conditions as those shown in
the actual test run – correction – medium GADF 136
Fig.6.1.1.2b: Simulation result showing the GPS data in white and the Spider path in
yellow – correction – medium GADF
Fig.6.1.1.2c: The linear velocity, angular velocity and GADF respectively for the simulated
robot. The vertical lines represent the waypoints $(2 - 19)$ – correction – medium GADF.
Fig.6.1.1.2f: Percentage of waypoints hit for test with varying radius size for simulation -
correction – medium GADF

Fig.6.1.1.3a: Result showing the GPS data in white and the Spider's path in yellow - no
correction – medium GADF 140
Fig.6.1.1.3b: The linear velocity, angular velocity and GADF respectively. The vertical
lines represent the waypoints (2 – 19) – no correction – medium GADF 140
Fig. 6.1.1.3e: Percentage of waypoints hit for test with varying radius size for test- no
correction – medium GADF 142
Fig.6.1.1.4a: Simulation of the Spider in Matlab under similar conditions as those shown in
the actual test run – no correction – medium GADF144
Fig.6.1.1.4b: Simulation result showing the GPS data in white and the Spider's path in
yellow - no correction - medium GADF 144
Fig 6.1.1.4c: The linear velocity, angular velocity and GADF respectively for simulated
robot. The vertical lines represent the waypoints $(2 - 19) - no$ correction – medium GADF.
Fig. 6.1.1.4f: Percentage of waypoints hit for test with varying radius size for simulation -
no correction – medium GADF 146
Fig.6.1.2: Before and after GPS data correction – dark line indicate before and the light one
after correction. The large circle shows RA where 95% of the static data falls within its
boundaries148
Fig.6.1.2.1a: Simulation result showing the GPS data in white and the Spider's path in
yellow – correction – High GADF
Fig.6.1.2.1b: The above figure shows the linear velocity, angular velocity and GADF
respectively. The vertical lines represent the waypoints $(2 - 19)$ –correction – high GADF.
Fig. 6.1.2.1e: Percentage of waypoints hit for test with varying radius size for test -
correction – high GADF
Fig.6.1.2.2a: Simulation of the Spider in Matlab under similar conditions as those shown in
the actual test run – correction – High GADF154
Fig.6.1.2.2b: Simulation result showing the GPS data in white and the odometry in yellow
- correction - High GADF 154
Fig 6.1.2.2c: The linear velocity, angular velocity and GADF respectively for simulated
robot. The vertical lines represent the waypoints $(2 - 19)$ –correction – high GADF 155

Fig. 6.1.2.2f: Percentage of waypoints hit for test with varying radius size for simulation –
correction – high GADF156
Fig.6.1.2.3a: Result showing the GPS data in white and the Spider's path in yellow – no
correction – high GADF158
Fig.6.1.2.3b: The above figure shows the linear velocity, angular velocity and GADF
respectively. The vertical lines represent the waypoints $(2 - 19) - no$ correction – high
GADF
Fig. 6.1.2.3e: Percentage of waypoints hit for test with varying radius size for test - no
correction – high GADF 160
Fig.6.1.2.4a: Simulation result showing the GPS data in white and the Spider's path in
yellow – no correction – high GADF 162
Fig. 6.1.2.4b: Simulation result showing the GPS data in white and the Spider's path in
yellow – no correction – high GADF 162
Fig 6.1.2.4c: The linear velocity, angular velocity and GADF respectively for simulated
robot. The vertical lines represent the waypoints $(2 - 19) - no$ correction – high GADF. 163
Fig. 6.1.2.4f: Percentage of waypoints hit for test with varying radius size for simulation –
no correction – high GADF 164
Fig.6.2.1.1: The path of the robot (yellow) compared to the path of the GPS is shown for a
point to point following approach167
Fig.6.2.1.2: The path of the robot (yellow) compared to the path of the GPS is shown for a
point to point following approach of the simulation168
Fig.6.2.1.2: The path of the robot (yellow) compared to the path of the GPS is shown using
circular stages of closeness at the radius of 0.1 m
Fig.6.2.1.4: The path of the robot (yellow) compared to the path of the GPS is shown using
circular stages of closeness at the undesirable radius of 0.1 m of the simulation
Fig.6.3.1-1: Circular stages of closeness for RA = 2.25 m 173
Fig.6.3.1-1S: Circular stages of closeness for RA = 2.25 m (simulation) 173
Fig.6.3.1-2: Circular stages of closeness for RA = 2.0 m 174
Fig.6.3.1-2S: Circular stages of closeness for RA = 2.0 m (simulation) 174
Fig.6.3.1-3: Circular stages of closeness for RA = 1.25 m 175
Fig.6.3.1-3S: Circular stages of closeness for RA = 1.25 m (simulation) 175

Fig.6.3.1-4: Circular stages of closeness for $RA = 1 m$
Fig.6.3.1-4S: Circular stages of closeness for RA = 1 m (simulation)
Fig.6.3.1-5: Circular stages of closeness for $RA = 0.75$ m
Fig.6.3.1-5S: Circular stages of closeness for $RA = 0.75$ m (simulation)
Fig.6.3.1-6: Circular stages of closeness for RA = 0.5 m
Fig.6.3.1-6S: Circular stages of closeness for $RA = 0.5 \text{ m}$ (simulation)
Fig.6.3.1-7: Circular stages of closeness for $RA = 0.25$ m
Fig.6.3.1-7S: Circular stages of closeness for RA = 0.25 m (simulation)
Fig.7.1.1.1a Number of waypoints hit: - with correction, - no correction
(Medium GADF) 184
Fig. 7.1.1.1b: Corrected GPS data (left image) and non-corrected (right image) at medium
GADF. The yellow lines represent the Spider's path and the white the GPS path
Fig. 7.1.1.1c: Predicted satellite visibility and PDOP during the time of testing for medium
GADF 186
Fig.7.1.1.2a: Number of waypoints hit simulation: - with correction, - with no
correction
(Medium GADF) 188
Fig. 7.1.1.2b: Corrected GPS data (left image) and non-corrected (right image) at medium
GADF of the simulation. The yellow lines represent the Spider's path and the white the
GPS path
Fig.7.1.1.3a Number of waypoints hit: - with correction, - no correction
(High GADF) 190
Fig. 7.1.1.3b: Corrected GPS data (left image) and non-corrected (right image) at high
GADF. The yellow lines represent the Spider's path and the white the GPS path 191
Fig. 7.1.1.3c: Predicted satellite visibility during the time of testing and the PDOP for high
GADF
Fig.7.1.1.4a: Number of waypoints hit simulation: - with no correction, - with
correction
(Medium GADF) 194

Fig. 7.1.1.4b: Corrected GPS data (left image) and non-corrected (right image) at high GADF of the simulation. The yellow lines represent the Spider's path and the white the Fig.7.2.1.1: Number of waypoints hit: - Medium GADF, - High GADF..... 198 Fig.7.2.1.2: Number of waypoints hit: - Medium GADF, - High GADF..... 199 Fig. 7.2.2a: The left hand images show the Spider's path (yellow) and GPS path (white) at medium GADF, with the corresponding fuzzy controller's linear and angular velocities. The right hand images show the same results at a high GADF...... 200 Fig. 7.2.2b: The left hand images show the Spider's path (yellow) and GPS path (white) at medium GADF, with the corresponding fuzzy controller's linear and angular velocities of the simulation. The right hand images show the same results at a high GADF. 201 Fig.7.4.2a: The path of the robot (yellow) compared to the path of the GPS (white). The left image shows the result of the point-to-point approach and the right-hand image that of circular stages of closeness. The results were shown for the same duration of time....... 206 Fig.7.4.2b: Fluctuations of the GPS are accommodated for by the fuzzy controller. The Fig.7.4.2c: The path of the robot (yellow) compared to the path of the GPS (white) of the simulation. The left image shows the result of the point-to-point approach and the right-Fig.7.4a: The left shows the result from the High GADF using the corrected data, and the Fig.7.4b: Close up of the heading control strategy result, showing that the right hand image Fig. A3: Schematic of the hardware used for controlling the position of the rotating Fig. A5: Wireless routers: a) for the lower platform and b) for the upper platform....... 246

Fig. A7: Axis network camera	. 247
Fig. A8: The IRYSIS infrared and AXIS cameras mounted on the Spider	. 248
Fig. A9: The CMPS03 compass by Devantech. [Devantech Ltd. 2007]	. 248
Fig. A10: Dual-Axis MEMS pan and tilt sensor ADXL203 [Analog Devices Inc. 2007]] 249
Fig. A11: A fuel gauge with 10 discrete outputs	. 249
Fig. A12: A screenshot of the interface program. Clockwise from top left: AXIS can	mera,
joystick position, GPS data output and the IRYSIS infrared camera.	. 250
Fig.B1: First stage fuzzy controller	. 253
Fig.B2: The first stage fuzzy controller layout - Matlab	. 253
Fig.B3: The fuzzy rules for the primary fuzzy stage - Matlab	. 254
Fig.B4: The input membership functions for the PDOP - Matlab	. 254
Fig.B5: The input membership functions for the number of satellites - Matlab	. 255
Fig.B6: The output membership functions of the GADF - Matlab	. 255
Fig.B7: Second stage fuzzy controller	. 256
Fig.B8: Second stage fuzzy controller layout - Matlab	. 256
Fig.B9: The fuzzy rules for the secondary fuzzy stage - Matlab	. 257
Fig.B10: The input membership functions for the Theta error - Matlab	. 257
Fig.B11: The input membership functions for the GADF – Matlab	. 258
Fig.B12: The input membership functions for the dc – Matlab	. 258
Fig.B13: The output membership functions for the angular velocity (w) – Matlab	. 259
Fig.B14: The output membership functions for the linear velocity (v) – Matlab	. 259
Fig.B15: Main model of the Spider's simulation – Simulink	. 260
Fig.B16: First sub-model of the Spider's simulation - Simulink	. 261
Fig.B17: Second sub-model of the Spider's simulation - Simulink	. 262
Fig.B18: Third sub-model of the Spider's simulation - Simulink	263

xxiv

,

List of Tables

Table 4.3.2.2: Comparison showing the variation in the shift between the average easting
and northing data pre- and post-testing
Table 5.3.2. Stages of closeness for loading upcoming waypoints. 112
Table 6.1.1: The conditions present during the medium GADF testing
Table 6.1.1.1c: Distance at which the waypoints are reached – correction – medium GADF
Table 6.1.1.1d: Summary of the average velocities of the travelled distance - correction -
medium GADF 133
Table 6.1.1.2d: Distance at which the waypoints are reached - simulation - correction -
medium GADF
Table 6.1.1.2e: Summary of the average velocities of the travelled distance - simulation -
correction – medium GADF 138
Table 6.1.1.3c: Distance at which the waypoints are reached - no correction - medium
GADF
Table 6.1.1.3d: Summary of the average velocities of the travelled distance - no correction
- medium GADF
Table 6.1.1.4d: Distance at which the waypoints are reached - simulation -no correction -
medium GADF
Table 6.1.1.4e: Summary of the average velocities of the travelled distance - simulation -
no correction – medium GADF 146
Table 6.1.2: Conditions present during the high GADF testing 148
Table 6.1.2.1c: Distance at which the waypoints are reached - correction - high GADF. 151
Table 6.1.2.1d: Summary of the average velocities of the travelled distance - correction -
high GADF
Table 6.1.2.2d: Distance at which the waypoints are reached - simulation - correction -
high GADF
Table 6.1.2.2e: Summary of the average velocities of the travelled distance - simulation -
correction – high GADF

Table 6.1.2.3c: Distance at which the waypoints are reached - no correction - high GADF
Table 6.1.2.3d: Summary of the average velocities of the travelled distance - no correction
– high GADF
Table 6.1.2.4d: Distance at which the waypoints are reached - simulation -no correction -
high GADF163
Table 6.1.2.4e: Summary of the average velocities of the travelled distance - simulation -
no correction – high GADF 164
Table 7.2: Summary of number of satellites and PDOP for both GADF test conditions 198
Table 7.2.2c: Summary of the angular velocities from the controller, sensors and simulation
for medium and high GADF – NZ (no zeros)

•

List of Variables

α	Angle of rotation of Spider frame to global frame (units: radians)
b	Distance from axis of rotation of back left wheel (b_1) and of front left wheel
	(f_i) to the centre of the robot (P) along the Y_{Spider} axis (units: metres)
- <i>b</i>	Distance from centre of robot (P) to the axis of rotation of the back right
	wheel (b_r) and to that of the front right wheel (f_r) along the Y_{Spider} axis
	(units: metres)
b_l	Back left wheel of robot
b,	Back right wheel of robot
d	Distance from axis of rotation of front left wheel (f_i) and of front right
	wheel (f_r) to the centre of the robot (P) along the X_{Spider} axis
	(units: metres)
-d	Distance from centre of the robot (P) to the axis of rotation of the back left
	wheel (b_i) and the back right wheel (b_r) along the X_{Spider} axis
	(units: metres)
d_c	Distance between the outside Spider error circle and the outside of the
	waypoint error circle. (units: m)
d_p	Distance between the Spider error circle pivot and the waypoint pivot.
	(units: m)
f_l	Front left wheel of robot
f_r	Front right wheel of robot
Р	The actual positional output of the robot with respect to the global reference
	frame in an ideal environment with no positional error.
P_{GPS}	The GPS positional output of the robot with respect to the global reference
	frame.
ϕ	Rotational angle of wheel around horizontal axis (units: radians)

$\dot{\phi}$	Rotational velocity of wheel around horizontal axis (units: rad/s)
$R(\theta)$	Orthogonal rotation matrix
θ	Angle of rotation (Steering angle) of wheel frame to Spider frame (units: radians)
θ	Angular velocity (Steering velocity) of the wheel frame to the Spider frame (units: rad/s)
$ heta_{b}$	Angle between the negative translational direction of the wheel and the
	Spider's frame. (units: radians)
θ_{eb}	Theta error back: The resultant angle due to the subtraction of θ_b from θ_g .
	This determines the true angle between the reverse translational direction of the wheels and the goal. (units: radians)
$ heta_{e\!f}$	Theta error front: The resultant angle due to the subtraction of θ_f from θ_g .
	This determines the true angle between the forward translational direction of the wheels and the goal. (units: radians)
$\theta_f = \theta$	Angle between the positive translational direction of the wheel and the
	Spider's frame. (units: radians)
$ heta_{g}$	The angle between the centre point of the robot and the centre point of the
	waypoint with respect to the fixed global axis.
ν	Linear velocity of robot frame at point P (units: m/s)
v_f	Forward translational velocity of the Spider. (units: m/s)
v_r	Reverse translational velocity of the Spider. (units: m/s)
X_{Global}	X axis of Global frame
$X_{\it Spider}$	Fixed X axis to robot frame
ξ	Posture vector of robot in global frame
چ	Robot posture velocity vector frame
Y_{Global}	Y axis of Global frame
$Y_{_{Spider}}$	Fixed Y axis to robot frame

xxix

XXX

Glossary

AI	Artificial Intelligence
BOP	Backward Odometry Path
CSEP	Circular Spatially Extended Points
CCW	Counter-clockwise
CW	Clockwise
DARPA	Defense Advanced Research Projects Agency
DEM	Digital Elevation Model
DGPS	Differential Global Positioning System
DOP	Dilution of Precision
DR	Dead Reckoning
EGNOS	European Geostationary Navigation Overlay Service
EKF	Extended Kalman Filter
EO	Earth Observation
EPE	Estimated Positional Error
FLC	Fuzzy Logic Controller
FOP	Forward Odometry Path
FSCAG	Food Supply Chain Automation Group
GADF	GPS Accuracy Decision Factor
GCP	Ground Control Point
GIS	Geographic Information System
GLONASS	Global Navigation Satellite System
GP	General Purpose
GPS	Global Positioning System
GSD	Ground Sample Distance
GUI	Graphical User Interface
HDOP	Horizontal Dilution of Precision
ICC	Instantaneous Centre of Curvature
ICR	Instantaneous Centre of Rotation
IFR	The International Federation of Robotics

IMU	Inertial Measurement Unit
INU	Inertial Navigation Unit
IR	Infrared
IRS	Indian Remote Sensing
LADAR	Laser Detection and Ranging
LIDAR	Light Detection and Ranging
NMEA	National Marine Electronics Association
NOS	Number of Satellites
OS	Ordnance Survey
PAS	Positioning Augmentation Services
PC	Personal Computer
PDOP	Position Dilution of Precision
PID	Proportional-Integral-Derivative (Controller)
PIC	Programmable Intelligent Computer
PPS	Precise Positioning System
QZS	Quasi-Zenith Satellite
RGB	Red Green Blue
RMS	Root Mean Square
RTK	Real-time Kinematics
SA	Selective Availability
SLAM	Simultaneous Localisation and Mapping
SPOT	Satellite Pour l'Observation de la Terre
SPS	Standard Positioning System
ТМ	Transverse Mercator
UAV	Unmanned Air Vehicle
UGV	Unmanned Ground Vehicle
UNECE	United Nations Economic Commission for Europe
VB	Visual Basic
WAAS	Wide Area Augmentation System
WGS	World Geodetic System
WWF	World Wildlife Federation

xxxii

.

xxxiii

Chapter pre-requisite: None

Nature of chapter: Introductory, contextual.

1.1 Robots

1.1.1 History

A *robot* usually conjures images of a man-like machine capable of performing tasks and interacting with its environment in much the same way that a human would. This is not a surprise, considering that the media often represents robots as intelligent, humanoid figures. The word *robot* probably stems from the Czech writer Karel Čapek, whose 1923 play R.U.R. (Rossum's Universal Robots) is about a character who builds humanoid robots for the sole purpose of serving their human masters [NASA 2006].

1.1.2 Industrial vs. Service Robots

However, these ideas only begin to represent the current scope of this field. Robots are found in all shapes and sizes, and can perform various task-oriented functions (see Fig. 1.1.2). The United Nations Economic Commission for Europe (UNECE) and the International Federation of Robotics (IFR) classify robots under the terms *industrial* and *service*. *Industrial* robots perform activities such as handling operations, assembling, and welding with high precision. *Professional service* robots include those deployed in field, medical, logistics, and defence applications (see Fig.1.1.2). Finally, *domestic service* robots are used for tasks such as lawn mowing, vacuuming, entertainment, and even assistance for people with disabilities [IFR 2005].



Fig. 1.1.2: a) an industrial robot, b) professional surgical service robot, c) domestic grass cutting service robot and d) domestic assistant service robot.

1.1.3 Service Robots: Autonomous vs. Semi-autonomous

Robots are often referred to as artificially intelligent (AI) machines. The definition of 'intelligence' is often debatable, although roboticists agree upon situatedness and embodiment as essential characteristics [Chatila, R. 1994]. The former refers to the fact that a robot interacts with an environment, and the latter that a robot does not only consist of a software program, but is a physical machine equipped with effectors and sensors.

Mobile robot *intelligence* can therefore be classified under the following two categories: *autonomous*, wherein the robot is 'intelligent enough' to travel through an environment without any human interference, and *semi-autonomous*, wherein some form of human intervention is required. These terms give a convenient differentiation between robot types.

However, these are presently only useful as relative terms. The majority of autonomous robots cannot operate with complete independence, but can operate for limited periods with variable human intervention. Jenson, B. *et al* [2005] presented 11 autonomous robots (RoboX) as tour guides at the 2002 Swiss National Exhibition; they were able to operate for approximately 12 hours a day, with staff interventions every 3.26 hours (on average), within a structured environment. The most successful autonomous field mobile robot to-date has been Stanley, developed by Stanford University's racing team for the DARPA Grand Challenge. The robot completed the solo 132-mile course through California's Mohave Desert in less than 6 hours, 52 minutes, at an average speed of 19 miles/hour [Thrun, S. *et al* 2006]. Though it managed to complete the course on its own, Stanley required careful monitoring during the course, and significant preparation prior to

deployment. The time has not yet come where a *fully* autonomous robot (i.e. continuously operable without human intervention) has been deployed.

Some robots have the capability of switching between autonomous and semi-autonomous modes. The converted golf cart vehicle ROMEO can operate either autonomously or by teleoperation via a wireless video link [Ollero, A. *et al* 1999]. However, even when the vehicle operates in the autonomous state, it navigates a route that has been predefined by the user. Another example of such a system is that of the volcano explorer ROBOVOLC (see Fig. 1.1.3) [Sim, D. *et al* 2004].

Telerobotics is defined as a form of *teleoperation* wherein a robot combines its own sensory data and intelligence with input from a remote human operator [Sheridan, T. 1989]. In 1945, R. C. Goertz developed the first modern master-slave teleoperators, which manipulated radioactive materials by a human operator outside of a 'hot cell' [Hokayem, P. *et al* 2006]. Šafarič *et al* [2001] demonstrated the usefulness of teleoperation for training purposes using a method wherein users could work on equipment using a simulated representation over the internet – which was then executed on a real device in a remote location. Teleoperation is one form of semi-autonomous performance, and is well established as a technique in the field of mobile robots.



Fig. 1.1.3: ROBOVOLC (right) and its user interface (left).

Autonomous service robots have been implemented in air, land and marine applications. However, land vehicles are the area of interest in this research.

1.1.4 Research in Mobile Robots

Research in mobile robots has often been broken down into components that govern the way the robots 'think', move, and react. Every researcher has a preferred method for describing these actions, but this research will simplify these terms into four categories: *localisation, path planning, navigation, and obstacle avoidance.*

Localisation is the process of determining a robot's position with respect to a global reference frame (i.e. its working environment). This typically involves the use of positioning sensors, such as the Global Positioning System (GPS), odometry with Inertial Navigation Units (INUs) and compasses (in addition to other sensors), as will become clear in the forthcoming Literature Review.

Path planning involves the creation of a path, either autonomously by the robot or with human aid. Preferred methods include the use of Voronoi graphs, visibility edges, potential field methods, bug algorithms, and Simultaneous Localisation and Mapping (SLAM). It is difficult to separate path planning and navigation, because often path planning takes into account motion control and limitations of the robot, which are also crucial aspects of navigation.

Navigation includes the robot's motion control and path tracking aspects. PID, fuzzy systems, and neural networks are only a few of many examples of controllers used for path tracking. Localisation is also an important component to navigation, since traversing the environment requires sensory positioning.

Obstacle avoidance is the process of reacting to obstructions that were unforeseen in the path planning stage. This is accomplished with the use of sensors, such as laser scanners, sonar, and vision systems. This aspect is closely coupled with navigation, since motion controllers must be equipped to contend with obstacles as the robot follows the given path.

In order to have a fully autonomous mobile robot, all of the above-mentioned research areas must be integrated in one form or another. As will become clear, these areas will all play a significant role in the progress of the work. The following research objectives will put these in context and clarify the specific problems that the work will set out to solve.

1.2 Research Objective

1.2.1 Food Supply Chain Automation Group

The contribution of agriculture to the UK's Gross Value Added has decreased from £9.8 to £5.2 billion in the decade leading up to 2005. This is due in part to the growing competition from cheap imported produce. Greater demand for nationally grown agricultural products has enticed UK farmers to compete with these imports [WWF-UK 2006], but they have struggled due to the lack of the necessary workforce. Only 2% of the UK population worked in agriculture, forestry, and fishing in the year 2000 [Lindsay, C. 2003]. As a result, many harvesting jobs have been filled by migrant student workers through the Seasonal Agricultural Workers Scheme, which has introduced problems of its own, including clashes between the workers and the local rural populations [Clarke, J. and Salt, J. 2003].

The harsh working conditions of farm work, along with the growing concern of using cheap migrant labourers, provide a strong motivation for the introduction of autonomous mobile robots for agricultural/horticultural applications. However, because machines are expensive (a combine can cost up to USD \$250,000 (~ GBP £125,000)) and are only used for short harvesting times, farmers are reluctant to invest in newly-developed systems, preferring to use traditional methods and machines [Romans, W. *et al* 2000]. In the past, the lack of funding for robotics in agriculture in the UK has led to slow development, making it difficult to produce low-cost, robust machinery [Hollingum, J. 1997]. (In contrast, the automation of agricultural machines in Japan has been relatively swift since the General Agreements on Tariff and Trade (GATT) in 1993, after which the government

decided to improve their agricultural infrastructure and overall production [Torii, T. 2000].) The Universities of Loughborough, Nottingham and Warwick are therefore working on the deployment of lower-cost autonomous robots through the jointly created Food Supply Chain Automation Group (FSCAG) to pave the way for agricultural/horticultural automation.

1.2.2 Agricultural Robots

There are a variety of autonomous and semi-autonomous robots in agriculture, each with a specific aim and utilising different combinations of sensors and driving mechanisms. Robots have been designed for weeding, crop spraying, rice transplanting, and harvesting, to name just a handful of applications. In addition, some international developments in advanced agricultural machines include a combine harvester that determines the crop yield, a herbicide-spraying machine that ensures that the nozzle is spraying on the correct target, and a liquid fertiliser spreader that determines where fertilisers are most needed [Baerdemaeker, J. *et al* 2001].

Some of these robots are designed with the principle of *precision farming* or the management of variability within field boundaries (for example, collecting environmental information with attention to time, place, and quantity variables to improve efficiency and production) [Earl, R. *et al* 2000]. Precision farming can presently be achieved only with the integration of multiple high-level, high-cost sensors and significant computational power.

Farmers are more likely to be convinced that autonomous machinery is worth the initial investment if the robots are capable of automating a variety of agricultural tasks [Kassler, M. 2001]. Ransomes-Jacobsen's mower, the *Spider*[®] fills the niche for an affordable agricultural robot platform (with appropriate alterations/additions) because of its manoeuvrability and robustness (see Fig. 1.2.2)¹ [Ransomes-Jacobsen 2005]. Currently, the

¹ Note: The Spider[®] will be referred to as the 'Spider' throughout the document.

Spider is capable only of cutting grass; however, researchers at Aarhus University, Denmark, have recently adapted the Spider for weed-spraying, reducing herbicide usage by 75%. Additional alterations, such as using vision-guiding sensors for weed-detection [Komi, P. *et al* 2007], could expand its repertoire of tasks and, therefore, its desirability for the consumer. Mechatronics plays an important role in these new agricultural developments [Sigrimis, N. *et al* 2001].



Fig. 1.2.2: The Spider, a robust grass mower by Ransomes-Jacobsen Ltd.

1.2.3 Research Interest

Considering the above, it is clear that there are still research areas that need to be explored. The following will define the particular areas of interest that this research will explore.

In recent years there has been growing interest in imagery (aerial, satellite, LADAR/LIDAR, DEM and more) for robot path planning. One task that these images can be used for is the collection of waypoints for mobile robot path planning. Conventional deliberative path planning in agricultural applications often requires the user to drive the
vehicle along a predefined path, which will then be used for autonomous navigation. However, the most common procedure is through manual collection (surveying) of waypoints using a high precision differential GPS receiver [Sim, P. *et al* 2003; Sethuramasamyraja, B. 2003], either using real-time kinematics (RTK) or post-processing the data. Even though this is a simple task, it is time consuming and requires thorough knowledge of the robot's working environment. The aerial image will convey all of the information needed for *a priori* waypoint path planning, which would provide a more efficient approach than the current manual field collection process [Sim, P. *et al* 2003].

The concept of using imagery for defining waypoints is not a new idea. Freely available Geographic Information System (GIS) tools such as *Google Earth* are often used by civilians to define their own route of travel. For in-car GPS navigation, the accuracy of these points is not critical because the waypoints are often conveyed relative to a global fixed street network, and are not required for autonomous navigation. Therefore, positional inaccuracies from the GPS receiver and the waypoint positional resolution do not act as a hindrance on the system's overall performance. However, for applications requiring higher navigational precision and autonomy, such as in mobile robots, it is believed that greater significance must be attributed to image settings and coordinate reference systems to improve the waypoint accuracy and GPS settings to ensure that the waypoints are reached.

The freely available differential WAAS/EGNOS signal that works in conjunction with the GPS offers positional accuracy up to 3 m. In addition, the upcoming deployment of Europe's alternative to the US-controlled GPS, the Galileo system, promises positional accuracy to within 1 m with no signal degradation all year round, and will inevitably make this system a desirable stand-alone application for localisation. However, with the current research objectives, the system would not be suitable for accurate guidance of agricultural vehicles for tasks such as fertiliser spraying, since farmers typically expect +/- 15 cm accuracy [Lenain, R. *et al* 2006]. However, utilising the GPS for localisation at its current level of accuracy, in addition to existing high-resolution aerial imagery, it will be determined whether the mobile robot will be sufficiently outfitted to perform an important

step for agricultural/horticultural applications: transport and grass mowing, which is also an exciting area of development.

It is important to note that this system is not intended to replace the need for an Inertial Measurement Unit (IMU) and will not provide sub metre accuracy like those obtained from subscription DGPS; however, the interest in this research is to possibly extend the capabilities of the currently available WAAS/EGNOS signal using a low-cost GPS sensor for localisation.

In order to realise these aims, the investigation of a suitable control strategy would need to be carried out in order to accommodate for all the various elements of the research.

The work on aerial images, currently available GPS technology and an appropriate control strategy would aim to provide researchers with a faster way to test their robots' performance using waypoints. This work will hopefully open up various other research opportunities. It could prove to be valuable in many applications such as security and farming, and with the adaptable nature of the Spider's architecture it is believed that this research will be realisable.

Chapter 2: Literature Review

A thorough review of the literature pertaining to the areas of research mobile robot was conducted. Particular attention was given to the research elements that correlate closely to the novelties explored in this thesis.

These have been partitioned into the following sub-sections:

- a) A review of *mobile robot mechanisms*, which will focus on synchronous drive research.
- b) Robot localisation, which will detail the pertinent sensors used by mobile robots to find their position in space. This section will expound on the GPS and systems that use the GPS.
- *Path planning and navigation*, which will explore both pre-driven and *a priori* approaches.
- Map representations in mobile robots, which describes and criticises conventional maps, aerial imagery, and other imagery considerations.
- e) *Literature ga*p, which highlights the areas of research that have not yet been fully explored by other researchers, and suggests areas that will be topical in this thesis.

In the above sections, the various control systems used by the mobile robots will be briefly described, and the literature gap will touch upon the control system areas that will be developed further.

Each of the sub-sections a, b, c and d is followed by a summary of the review.

Chapter pre-requisite: None. Nature of chapter: Informational, contextual.

2.1 Robot Mechanisms

2.1.1 Synchronous Drive Robot Research

Literature review has revealed few uses of synchronous drive systems in mobile robots. Thus far, synchronous-drive platforms have been utilised in research specific to collision avoidance and the study of odometry error. Past work has also focussed on the study of indoor mobile robots.

In general, the most commonly used general motion equations describing the kinematics for synchronous drive robots are:

$$x(t) = x(t_0) + \int_{t_0}^{t_n} v(t) .\cos(\theta) dt$$

$$y(t) = y(t_0) + \int_{t_0}^{t_n} v(t) .\sin(\theta) dt$$
(1)

$$\theta(t) = \theta(t_0) + \int_{t_0}^{t_n} \dot{\theta}(t) .dt$$

These equations can be found in a variety of publications [Fox, D. *et al* 1997; Siegwart R. and Nourbakhsh, I.R. 2004; Dudek and Jenkin 2000].

In the 1990s, robots from Nomadic Technologies – namely, the Nomad 150, Nomad 200, and the SuperScout – were the most commonly used platforms for mobile robot research [Chopra, A. *et al* 2006]. The Nomad 200 was of particular interest due to its synchronous-drive base, which was attractive for test areas because it was able to manoeuvre in small spaces. However, these robots have become obsolete as their computational components have become severely outdated and incompatible with modern operational systems. Therefore, despite the proven success of these platforms for research, they are nearly inoperative without extensive hardware upgrades.

Tsourveloudis, N. *et al* [2001] combines a synchronous drive mobile robot with a potential field path planner and a fuzzy logic controller, allowing real-time path-planning and collision avoidance. To achieve this, an occupancy map (a map detailing the obstacles) of the environment is first developed and mapped onto a network that applies a potential field to these obstacles (virtual resistor network). The robot (*Nomad 200*) is given a goal, and uses the potential field to find the path of minimum occupancy within the environment. The fuzzy logic engine simultaneously interprets information from 360° omni-directional Infrared positional sensors and achieves collision avoidance from dynamic obstacles from the front, back, left, and right. Because the robot has a synchronous drive platform, it is highly efficient in both its path-planning around the objects in the occupancy map and in its fuzzy logic collision avoidance. Instead of making wide turns around obstacles, the robot can simply shift out of the way, keeping sight of its trajectory with minimum interruption.

Fox, D. *et al* [1997] also studied reactive collision avoidance, taking into account the specific motion dynamics of synchronous drive robots (using the robot *Rhino*). The chosen method, called the Dynamic Window approach, allows robots to reliably avoid collisions at speeds up to 0.95 m/s by considering only the next steering command when searching for obstacles. The authors determine that the motion trajectories of synchronous-drive robots can be accurately approximated by a sequence of circular arcs. This is unique to a synchronous drive system because of its ability to make 'turns' without changing its orientation in space, and this approach will be considered in the proposed work.

Doh, N. *et al* [2006] developed a method for accurate relative localisation for synchronousdrive robots, and described the sources of odometry error which are specific to synchronous-drive vehicles. The authors' Path Comparison (PC) model is achieved by driving the robot (*Nomad 200*) through a known path along a generalised Voronoi graph (is a map that puts the largest possible distance between the robot and obstacles to maintain a collision-free course), while recording odometry information. This is called the forward odometry path (FOP) The robot then travels the path again in reverse (the backward odometry path (BOP)), without correcting the odometry error. Differences between the FOP and BOP are noted by detailing the coordinate transform; from this, error parameters are derived which can be later used to correct odometry. However, limitations do lie in determining those parameters. In order to reduce non-systematic error and obtain the parameters, the robot had to be run at speeds as low as 0.13 m/s for a distance of 94.4 m, revealing an average corrected error of 0.28 m. When tested at twice the speed (0.26 m/s) and half the distance (49 m), an error of 4.23 m was obtained. Given that the robot used for this project, the Spider, has a minimum velocity of 0.83 m/s (which is a factor of six times larger than their recommended speed), this method would not provide a reliable odometry estimator for the proposed work. They proved that major sources of odometry error in synchronous-drive robots do not come from uneven mass distribution, offset distance between the centre of rotation and wheel, or different wheel radii. Instead, errors stem from wheel misalignment, which induces rotational errors from additional moments and forces.

Martinelli, A. [2002] also developed a method for modelling error of a synchronous-drive system (using *Nomad 150*), in addition to a possible method for error evaluation. Unlike Doh, who detailed the coordinate transform of the entire path to determine error, Martinelli measured only the change in position and orientation of the robot's initial and final configurations.

A similarity between the synchronous drive robots used is that they have direct control over the motor velocities using high frequency digital controllers [Alvarez, J. *et al* 1998; Fox, D. *et al* 1997], which is not possible with the robot used in this research. Since translational and angular velocity commands are fixed, no direct control over specific velocities is possible.

To date, no studies have shown synchronous drive robots used for outdoor applications. The majority of the work so far has been confined to indoor environments. Transposing the above-mentioned work to outdoor applications would pose a difficult challenge because terrain consistency and other outdoor features would need to be taken into consideration.

2.1.2 Robot Summary

In review, past work on synchronous drive mobile robot platforms has revealed that:

- a) They have been most frequently used in collision avoidance and odometry error research for indoor mobile robots.
- b) In terms of collision avoidance, synchronous drive platforms have been shown to be highly efficient in manoeuvring.
- c) It has been shown that odometry error in synchronous drive systems usually comes from wheel misalignment, which introduces rotational errors.

A lack of work has been done with synchronous drive robots in outdoor environments.

These findings will be considered as the proposed work is carried out on the Spider, for outdoor navigation.

2.2 Robot Localisation

2.2.1 Sensors

In robotics, localisation involves determining the position of a robot with respect to a global or fixed reference frame. Many types of sensors have been used to conduct work in this widely studied area. For example, research using landmarks for validating a robot's position to an *a priori* known map has been done using vision [Kotani, S. *et al* 1998]. In addition, laser range finders have been used for a technique known as SLAM (Simultaneous Localisation and Mapping), where the robot determines its relative location from mapping an unknown environment [Lee, K. *et al* 2004].

Summarising each of the different techniques used for localisation with their corresponding sensors would be an endless task. This section will therefore only touch upon the types of sensors that have been used for such applications, and they will be classified into the general categories of *active* and *passive*.

Active sensors emit energy into the environment and interpret the energy levels of the returning (reflected) signal. Examples of such sensors include 1D/2D laser scanners, sonar, active infrared cameras and radar. These types of sensors are less energy efficient than passive sensors, but they have proven to be more robust, since they require continuous interaction with the surrounding environment [Dudek, G. *et al* 2000]. This is an important feature for obstacle avoidance or mapping of the environment. Work using laser scanners for obstacle avoidance is frequently used. For example, the series of ROMEO vehicles developed at the University of Seville in Spain are equipped with 2D laser scanners and sonar for obstacle avoidance [Ollero, A. *et al* 1999]. Sonar sensors can calculate distances to obstacles by determining the difference between the time the signal was emitted and the time it was received [Gopalakrishnan, B. *et al* 2004].

Low cost active sensors are also readily available; however, they are often targeted at mobile robot hobbyists or for low cost applications (educational tools), and mainly for applications in indoor environments.

Passive sensors differ from active sensor because they do not emit any energy into the environment, but rather collect energy (data) from the environment. This feature makes them more energy efficient and lower in cost than active ones. Examples of these include potentiometers, passive infrared, cameras, inertial measurement/navigation units (IMU or INU), and compasses. Work relying only on passive sensing is termed dead-reckoning (*odometry*). Using only odometry, however, is a difficult task, because finding sensors that can provide reliable dead-reckoning results over a long distance is nearly impossible. Military or civilian aircraft grade inertial navigation units can provide high accuracy; however they also are associated with very high cost and require periodic calibration [Durrant-Whyte, H. 2005]. The study of odometry (i.e. relying on passive sensing) is important for determining the behaviour of a robot due to any systematic errors [Martinelli, A. 2002; Doh, N. *et al* 2006].

There have also been numerous studies relating to localisation using landmarks, where a robot identifies its location relative to known objects by using vision or active sensors [Kotani, S. *et al* 1998]. In this example, landmark localisation refers to the robot identifying edges of clearly identifiable objects with a vision system and using image transformation techniques to localise itself in space with respect to them. The use of vision for weed detection was conducted at the Mechatronics Research Lab at Loughborough University [Komi, P. *et al* 2007]. This work could eventually be modified for guiding the robot through rows of crops in a farmland.

Cupertino, F. *et al* [2006] used a vision system and IR proximity sensors in a robot to carry out simple 'reach the target' and 'avoid obstacle' behaviours. The information from these passive sensors was fed into two fuzzy controllers, yielding flexible, human-like responses to stimuli.

In robotics, a further classification of sensors exists, termed *proprioceptive* and *exteroceptive*. *Proprioceptive* sensors measure values that are internal to the robot, such as speed and wheel position, and *exteroceptive* sensors measure values from the robot's environment, such as distances to obstacles and landmarks. Even though they differ from active/passive classifications they still share similar traits. For example, a sensor could be both passive *and* extereoceptive, like a vision system, where it measures values from the robot's environment but without having to emit energy into the environment. Optimally, both classifications should be used to describe a sensor. For a detailed summary of sensors and their classifications please refer to the text by Siegwart, R. and Nourbakhsh, I. [2004].

A sensor that is difficult to classify as either active or passive is the GPS. Some researchers classify the GPS as *active* and *exteroceptive* [Siegwart, R. and Nourbakhsh, I. 2004]; however, this classification is debatable. The receiver does not emit any energy into the environment, but rather receives positional information (in this case from satellites). On the other hand, from the satellite's perspective, energy is emitted into the environment enabling the GPS to localise itself within the global frame, therefore terming it active. Regardless of nomenclature, the energy efficiency and practicality of this sensor makes it a desirable unit to be used in mobile robot localisation, as will become clear.

Research involving the close coupling of different sensors – each often application-specific – requires a number of tasks that require a team of researchers working closely together. In this research no work will be presented on obstacle avoidance, goal tracing or other research requiring the use of active sensors (reactive control), and passive sensors will be used sparingly. Furthermore, since research focussing on dead-reckoning (odometry) or localisation using vision is also not part of this research, they will also not be included.

Recent enhancement of the accuracy of the GPS has also led this research to focus on the study of this sensor as a stand-alone application for robot localisation.

2.2.2 The Global Positioning System

2.2.2.1 General Background

Currently there exist two active satellite positioning systems, the GPS and the GLONASS.

The Global Positioning System (GPS) is a navigation system consisting of a network of 24 solar-powered satellites that provide ground positioning coordinates and precise timing. It was initially developed for military applications, with the first operational satellite launched in 1977, but since the mid-nineties has been freely available for civilian navigation. However, since the system is operated by the U.S. Department of Defence, the full functionality of Precise Positioning System (PPS) is given to the military. As for the Standard Positioning System (SPS), a deliberative timing error via a low-order bit encryption was introduced, which reduced the civilian accuracy to 100 m. This was known as Selective Availability (SA) [Moore, P. and Crossley, P. 1999]. Due to growing concerns about its unreliability, it was finally stopped in 2000, and accuracy was improved to 10-20 m [Ochieng, W. and Sauer, K. 2002]. However, the US military still reserves the right for full signal strength. The standard protocol for the GPS is set by the National Marine Electronics Association (NMEA).

The former USSR launched its own system, GLONASS (Global Navigation Satellite System), in1982. This system has been noted for its high accuracy (nearly as good as the military grade of the GPS) because it was not intended for civil users and therefore did not have built-in inaccuracies. However, the double-system receivers are far too technically complex and expensive for civil use [Michalski, A. and Czajewski, J. 2004] [O'Keefe, K. *et al* 2006].

Inaccuracies stemming from atmospheric conditions, orbit instability, and disturbances in the satellite constellation were first tackled by accurately georeferenced ground stations which acted as beacons and transmitted corrected GPS signals [Satirapod, C. *et al* 2004].

This is known as differential GPS (DGPS). However, the accuracy of the corrected signals degrades as the distance from these stations increases.

2.2.2.2 Differential Positioning (WAAS/EGNOS)

The problem of inaccuracies due to the ground beacon distance was approached by the introduction of geostationary satellites that transmit differentially corrected signals. In the US, this system is known as WAAS (Wide Area Augmented System), and can provide civilians positioning accuracies to within 3 m [Michalski, A. and Czajewski, J. 2004; O'Keefe, K. *et al* 2006].

Europe's answer to the civilian restrictions imposed by the US-controlled GPS is Galileo, a constellation of 30 low-earth orbiting satellites and ground stations. The first satellite was launched in December 2005, but its error-correcting signal service (similar to the WAAS), EGNOS (European Geostationary Navigation Overlay Service), has been correcting US GPS signals since 2003. However, it is believed that the system will not be fully operable until 2010.

The EGNOS/Galileo system will be used for applications requiring high precision such as navigating ships through narrow banks [Bretz, E. 2003]. This setup is expected to provide 3D positional accuracies up to 1 m without degradation all year round. Further cm-level precision will be available for purchase at an additional fee, with accuracies depending on proximity to ground stations.

Currently the EGNOS provides positioning accuracies between 3-5 m, but results had shown to produce positioning accuracies when tested in straight line paths to be offset by a mean of 0.11m as claimed by Witte, T. and Wilson, A. [2005].

It is projected that in 2009 Japan will also introduce an adaptation of the WAAS/EGNOS system known as the Quasi-Zenith Satellite (QZS), which will orbit at a higher altitude than GPS satellites; this feature will allow for high accuracy even in the presence of high-

rise buildings. Currently, cm-level accuracy positioning can be achieved via Mitsubishi's Positioning Augmentation Services (PAS); however, the signal can only be sent through mobile phones, and requires further processing by the user [Higuchi, H. *et al* 2004].

The high accuracy of the EGNOS/Galileo will make it superior to WAAS/GPS or GLONASS for precise robot applications and could make expensive correcting equipment (such as for DGPS beacons) obsolete.

Since the GPS has become a favourable option amongst researchers, a large amount of work has been contributed in this domain. Applications range from navigating a rice-transplanting robot through fields [Nagasaka, Y. 2004], to creating accurate cm-level digital road maps with probe vehicles for car navigation systems [Wang, J. *et al* 2005]. The applications are nearly endless. Therefore, to narrow down the applications and the search criteria, research was only focussed on mobile robots accomplishing localisation with a GPS navigator coupled with an IMU, and those using a GPS navigator alone.

2.2.3 Localisation

2.2.3.1 GPS and Passive Sensor Coupling (GPS & IMU/INS)

Typically, an Inertial Measurement Unit (IMU) is coupled with a GPS in order to provide higher positional accuracies [Durrant-Whyte, H. 2005]. Often this is accompanied by Kalman filtering to compensate for noisy data.

In the invited paper by Abbott E, *et al* [1999], a description of land-vehicle navigation using GPS, rate gyros, a compass, and odometry is studied. The focus of this paper is to show that each sensor is accompanied by error. The authors focus on the coupling of the data from the GPS with those from inertial instruments by means of Kalman filtering. A large proportion of this work discusses the inherent inaccuracy of the GPS system due to the Selective Availability (SA). This leads the research into the territory of sensor coupling,

and the authors state that absolute localisation based on GPS alone is not an option. However, since that paper's publication, SA has been removed and geostationary satellites (WAAS) sending differential positional data has been introduced, dramatically improving positional accuracies.

In work conducted by Panzieri, S. *et al* [2002], the authors used a low-cost GPS receiver, an inertial sensor, and a laser range finder coupled with an extended Kalman filter (EKF). They performed various tests solely with the GPS sensor, to better understand its capabilities, and performed them statically for 15-minute time intervals. The best results were achieved when a minimum of six satellites were available. Any fewer led to a lack of positional resolution in a certain direction. The authors also emphasize that the GPS's dilution of precision (DOP) and estimated positional error (EPE) strongly depend on the number of satellites in view, and hence provide an estimate of the absolute positional accuracy. However, since the authors' interests lay only in relative accuracy, they used the number of satellites as a criterion for calculating their covariance matrix of the GPS needed for the Kalman filter. During the testing stage, they used a simple waypoint path plan and a static time invariant feedback control law for reaching these points. Their concluding remarks state that the GPS can be used for localisation using inexpensive receivers.

It is important to be aware of the fact that this system was tested prior to the launch of the EGNOS geostationary satellite in 2003, when further enhancement of positional accuracy was introduced.

Prior to that time, a vast amount of research was devoted to the coupling of GPS and IMU due to reduced accuracy; however, presently they are still often closely coupled in order to obtain cm-level accuracy for certain applications such as digital road map creation [Wang, J. *et al* 2005].

If cm-level precision is not possible with the WAAS/EGNOS/GPS system on its own, then how far is it possible to stretch the use of the current differentially corrected signal as a stand-alone application? The following summary of work conducted on the use of GPS alone will help to answer this preliminary question.

2.2.3.2 GPS only

In the previous section, it was shown that a large proportion of the work involved the coupling of sensors. However, given that cm-level positional accuracy is not desired, and no GPS/IMU coupling is planned, a search for work involving only the use of the GPS as a stand alone application for robot localisation was undertaken.

Work by Hodo, D. *et al* [2007] has shown that imperfections in sensor measurements can have an important impact on control system performance if they are not taken into account. Their application involved a robotic tractor for pulling a trailer. The authors used a single GPS receiver for providing the robot's positional information. An RTK-GPS was used to provide cm-level accuracy.

As previously mentioned, with the dawn of the Galileo satellite navigation system, higher positional accuracy is expected than that which is currently attained through either the GPS or the GLONASS, and furthermore no signal degradation is expected all year round. Traditional navigation, as mentioned previously, involves relying on the INU to provide dead-reckoning when GPS loss of signal occurs. The integrity, reliability and other important features of the Galileo system would make navigation relying only on satellite positioning a favourable option in mobile robot applications [Ashkenazi,V. *et al* 2000], potentially eliminating the need for INU/GPS coupling. Therefore, this section discusses the aspect of robot localisation relying only on currently available EGNOS positioning.

Work conducted by Holden, M. [2004] has shown that a low-cost educational robot equipped with only a GPS receiver as its sensor has obtained good results for waypoint navigation. This method was also implemented on a boat yielding similar results. More details on this section will be discussed in the Path Planning and Navigation section (2.3).

Vaneck, T. [1997] also used only GPS for localisation with a satisfactory result. With positional fix updates roughly once per second, the author was able to achieve positional accuracy up to 1m on his autonomous boat. It was determined that any positional jumps were due to changes in the GPS satellite constellation used by the receiver, but the author noted that the results were 'more than accurate' for the application. Vaneck's approach will be discussed further in Section 2.3.2.

A single antenna DGPS was used by Cho, A. *et al* [2007] for take-off, landing, and taxiing of an unmanned aerial vehicle (UAV) with successful results. The system contained no inertial sensors, such as gyros or accelerometers, and contained only an airspeed sensor for safety. Though the system was promising, the UAV still resorted to waypoint path control after a specified speed and altitude were reached.

A comprehensive discussion of GPS use in mobile robots was compiled by Zidek, K. *et al* [2006]. The author maintains that GPS units – when used alone – are currently only usable as a 'coarse' fixing aid to within 2 m accuracy. However, this judgement refers to the current DGPS and WAAS systems, and does not take into account new developments that will increase accuracy.

2.2.4 Localisation Summary

In order for a robot to find its position with respect to a global or fixed reference frame, the following methods for localisation have been explored:

- a) Sensors. These can be categorised as either active/passive or proprioceptive/exteroceptive. These can range from potentiometers to infrared proximity sensors to the GPS.
- b) Satellite positioning systems. There are two truly global systems: the US GPS and the USSR's GLONASS. Inaccuracies in the GPS can be corrected by signals from accurately georeferenced ground stations (DGPS), for cm-level accuracy, on a subscription basis. Freely available differentially corrected signals using geostationary satellites (WAAS/EGNOS) can provide civilians with accuracies to within 3 m.
- c) GPS coupled with passive sensors. Research has been done coupling GPS with IMU/INUs to increase accuracy. To a lesser extent (and with variable results), GPS has been used as a stand-alone application for localisation.

The above points and past research will be considered in the proposed work. However, the interest in this research is to extend the capabilities of the currently available WAAS/EGNOS signal using a low-cost GPS sensor for localisation.

2.3 Path Planning and Navigation

In mobile robots, it is quite difficult to distinguish path planning from navigation, since a path is typically planned such that an appropriate navigation (controller) algorithm can be used. There also exist a vast number of path planning techniques. For this research, the focus will be geared towards the deliberative approach, wherein the robot follows a predefined trajectory, or a series of points, during the testing.

Waypoints have been defined differently by various researchers, but with the same principal. The following paragraph touches upon some definitions of waypoints. Cameron [1994] terms pre-defined path planning as the *railway track algorithm* because the vehicle is confined to specific paths or roadways (the 'tracks'). This is usually done when the coordinates of points along a path to be traversed, known as waypoints, are given to the robot. Waypoints have also been recently defined as points within the global frame with specific latitude and longitude coordinates. In a paper published by Durrant-Whyte, H. [2005] the author mentions that classical path planning techniques assume a full knowledge of the robot's environment, which is believed to be correct and complete. Furthermore, he refers to trajectory generation as a series of straight lines, spline curves, or smooth geometric structures between waypoints. This is similar to a method employed by both Ge, S. *et al* [2005] and Ren, W. *et al* [2007], in which trajectories (straight lines) are created between a starting point and each goal (waypoint). Since complete knowledge of the environment for outdoor robots is not possible in the current research, a method employing waypoint-type algorithms is quite suitable.

From this point onwards, the term *waypoints* will refer to outdoor points with given spatial coordinates and are obtained via a GPS or by some other means.

2.3.1 Deliberative (or Global) Approach

2.3.1.1 Pre-driven Path Planning and Navigation

In the research conducted by Ollero, A. *et al* [1999] and Stentz, A. *et al* [2002], a robot autonomously tracks a path around which it was previously driven. The steering commands and vehicle's position are recorded along the course to create this path. This method renders it difficult for the robot to react to changes in the environment, and therefore reactive obstacle avoidance must be implemented. For the tractor by Stentz, a maximum error of 0.28 m at a speed of 8 km/hr was recorded for a travel distance of 7 km.

A similar path-tracking approach was used by Antonelli, G. *et al* [2007]. Lane information gathered by a robot in an initial line-following journey is elaborated by a fuzzification module, which assigns linguistic variables to different characteristics of the path. The robot is then able to re-trace its journey, while adapting to the path following linguistic commands such as 'slow down while approaching a bend.' The implementation of a fuzzy logic controller (FLC) thus makes this pre-driven approach more flexible than that of Ollero or Stentz.

Another interesting method for creating a deliberative route was presented by Kidono, K. *et al* [2002], wherein the user guides the robot along a certain path, and the robot generates a map using stereovision. The robot records its position and the location of features, then adjusts its viewing direction to minimize its localization error. This method is similar to the concept of SLAM. The aim of this method is to develop an autonomous navigating robot that requires minimum user intervention. The results show that the robot autonomously navigates between the objects, and the accumulated error does not exceed an area of 0.2 m x 0.2 m.

Mulvaney, D. *et al* [2006] developed an 'extremely robust' hybrid navigation system combining deliberative and reactive approaches. An initial reactive system explores an unknown environment, identifying waypoints around obstacles; these then determine a

suitable path in later deliberative planning stages, which are performed using a genetic algorithm (GA). By using the parameters obtained from the GA, the computational and memory requirements can be kept to a minimum. Furthermore, the robot can revert to reactive control if operating in a dynamic environment. A similar, but more primitive, approach was employed by Dudek, G. *et al* (1991, 1997) wherein a robot explores an environment, leaving markers on its route; as more markers are added to the environment, the edges between them become a subgraph of an *a priori* undirected graph.

2.3.1.2 A Priori Path Planning and Navigation

Other researchers have found it beneficial to supply robots with *a priori* information about the environment rather than having the robot itself do the data collection for path planning and navigation. Beard, R. [2003] states that the benefit of *a priori* deliberative approaches is that trajectories, and timing, can be explicitly planned; however, one drawback to these methods is that the robot is dependent on the model. In Beard's football-playing robots, *a priori* information includes a series of stored waypoints and waypoint-pointers as part of the global state of the system. The robot is able to track a ball using vision sensors along the waypoint paths using a time-parameterised trajectory generator. A feedback linearisation technique is used to follow this trajectory. Leedy, B. *et al* [2006] used waypoints in a similar approach in a fully autonomous vehicle in the DARPA Grand Challenge.

Path planning and navigation is also aided by *a priori* information in work conducted by Sofman, B. *et al* [2006]. The authors detail how, given traversal cost information and a new overhead image, a robot is able to determine the most appropriate path through an area. This method will be explained further in Section 2.4.2.2.

A wheeled mobile robot developed by Maalouf, E. *et al* [2005] was also given pre-defined waypoints to follow. The authors implemented a fuzzy logic controller to give the robot more human-like responses. For example, if the road between two waypoints was straight,

the robot reacted by moving quickly between them; if the road was curved, it slowed for smoother turning.

Vandapel, N. *et al* [2003] and Silver, D. *et al* [2006] also used prior information to plan traversal costs (see Section 2.4.2.). Using maps from aerial LADAR survey, Vandapel's grid-based path planner is able to consider new trajectories through given waypoints to avoid prohibitive terrain. Silver found that having prior overhead data significantly improved the navigational performance of a mobile robot. Compared to the same robot traversing a course without prior overhead data, the number of required interventions per km while using *a priori* information decreased by more than a factor of three, and the average speed of navigation increased by 22%.

2.3.2 Waypoint Navigation Using GPS

Vaneck, T. [1997] proposed an autonomous boat for acquiring data for creating bathymetric (definition) maps. Using the localisation information obtained by the DGPS, the vehicle was able to navigate between (previously) geographically-defined waypoints. For GPS position estimation between waypoints, a dead reckoning (DR) algorithm was used, which combined GPS reading with compass heading. Steering commands based on fuzzy logic were used with successful results, and accuracies to within 20 cm were achieved.

One application found using GPS as a stand alone application for mobile robots is for an undergraduate mechatronics teaching course conducted at the San Francisco State University [Holden, M. 2004]. No obstacle avoidance functions are implemented. The authors use the WAAS capability of their GPS receiver and obtain positional estimates between 5-10 m due to the sensor's limited bandwidth and accuracy. It is mentioned, however, that in order to successfully control the platform, the robot must be *open-loop stable* (i.e. the steering mechanism would show no loss of control due to the effect of small ground based disturbances, if it were to travel with an open loop controller). The latitude,

longitude, GPS speed and heading estimate are used to control the robot. The robot trajectory is created through a series of waypoints, connected with straight lines. The navigation controller does not aim for the goal, but rather follows the line. There are two main controllers: heading control and speed control. The GPS was mounted on a small boat and a car. The results can be seen in Figs. 2.3.2a and 2.3.2b.



Fig. 2.3.2a: Autonomous path of the car



Fig. 2.3.2b: Autonomous path of the boat

A PID was used for both controllers. The results show good response; however, the jaggedness of the path is due to the sensor discretisation, since it only stores positional data every once every 2.13 m.

Ray, L. *et al* [2007] designed a solar-powered robot that uses GPS as a stand-alone application for waypoint following on the snow-covered, open-space Greenland landscape. The robot is able to reach goals using no additional sensors, with only open-loop motor-speed corrections in response to bearing deviations from the desired waypoint. However, the robot is limited because it is only suitable for use in unobstructed expanses of land [Lever, J. *et al* 2006].

Sandia National Laboratories, a leading centre for mobile robot applications, has developed a system wherein the user controls the positions of multiple robots for strategic military operations (see Fig. 2.3.2c). The system was developed so that a soldier can define each robot's goal via a set of waypoints on an aerial image through a user console, and can define avoidance regions by giving them a repulsive field [Feddema, J. *et al* 1999].



Fig. 2.3.2c: Aerial image for controlling RATLER robots.

In this system, the robots rely on only a compass, a differential GPS, and pan and tilt sensors. The compass is there to provide heading direction, since the robot adopts a drive system similar to a differential style robot.

There exist inherent problems with such an approach. Obstacle avoidance is achieved by a method in which the robot travels along a path, and if it stumbles into an obstacle, the robot attempts to climb it until the tilt sensors reach a threshold. Then, through a series of manoeuvres, the robot reverses, drives around the obstacle and continues along its predefined path. This type of system was built to work in open desert environments without real concern for obstacle avoidance. Their future work will use a potential field algorithm for the path planning stage (see page 35 for a brief description on the potential field algorithm).

The use of waypoint navigation has also been employed in UAVs; one such example is an autonomous kiteplane, which successfully manoeuvred despite wind disturbances using low-cost sensors [Kumon, M. *et al* 2006]. No definition of the systems accuracy were made, but observations from the flight's path were noted. Please refer to paper for more details.

2.3.3 Path Planning and Navigation Summary

There exist numerous methods for path planning and navigation. In this research, the focus will be on deliberative path planning approaches, wherein a robot follows a pre-defined trajectory of waypoints. Previous work has centred on the following methods:

- a) Pre-driven. In pre-driven approaches, a mobile robot stores the path information (creating waypoints) on an initial journey, and then uses this information when it re-traces the track on future journeys.
- b) A priori. In a priori path planning and navigation, the robot is given information about the path-environment (in the form of waypoints) prior to embarking on its first journey - either by using high-grade surveying equipment or waypoint selection from aerial imagery. In other cases, the trajectories and timing are preplanned, but making the robot dependent on the model.

In summary, the use of waypoints and GPS have been shown to be powerful tools for outdoor mobile robot navigating, and these will be explored and utilised in the proposed work.

2.4 Map Representation in Mobile Robots

In mobile robots, a map of the environment is not necessarily a pictorial representation, but is often a collection of sensory data obtained from the robot's immediate surroundings. These are sometimes used in conjunction with *a priori* maps for localisation in methods known as map-matching [Ashkenazi, V. *et al* 2000; Kim, S. and Kim, J.-H. 2001]. *A priori* maps can range from simplistic grids created by the user, to sophisticated Digital Elevation Models (DEMs) of the area. In some research, maps are created and, in the same instant, the robot uses this data to localise itself in the global frame. This method is referred to as SLAM, and is shown in work conducted by Lee, K. *et al* [2004]. Another popular approach is using a potential field method, in which repelling and attracting forces are assigned to obstacles and goals on a 'map' of the environment for a robot to navigate through [Hwang, Y. and Ahuja, N. 1992]. However, for this research there will not be an emphasis on creating sensory based maps or map-matching using landmarks, but instead on the use of aerial images for creating waypoints for an outdoor mobile robot.

This section will provide insight as to why this option was chosen as the preferred method. It will touch upon various types of maps used in mobile robot path planning or for sensorybased navigation. The nomenclature used below has been chosen for clarity and simplicity, and may differ from that chosen by other researchers. The following section will focus on *conventional* maps.

2.4.1 Conventional Maps

Conventional maps are those that have been created by the user in order to mimic the robot's environment.

2.4.1.1 Indoor

Thrun, S. [2002] published a comprehensive discussion of past and present approaches to indoor map creation, which is presented below.

The robotic maps from the mid-1980s to the early 1990s can be loosely divided into two categories: *metric* and *topological* approaches. Metric maps are those that are based upon the geometric properties of the environments. Occupancy grids are one type of metric map, and are composed of grids that show the free and the occupied space of the environment. Chatila and Laumond [1985] developed a second type of metric map using polyhedra to depict an environment. In contrast, topological maps merely describe the connectivity of different areas using arcs, rather than the exact geometric specifications of individual objects. Metric maps thus tend to be more finer-grained. However, it can be difficult to distinguish between metric and topological maps because most topological maps still use some geometric information.

A second way of categorising mapping algorithms, which has also been in use since the mid-1980s, is using a *world-centric* vs. a *robot-centric* approach. World-centric algorithms create maps that are represented within a global reference, whereas robot-centric algorithms create a map based only on sensory data from the robot itself. Even today, robot-centric approaches are unpopular, because it can be difficult for a robot to disambiguate two different areas if they 'look' alike (based on sensory data), without an external reference to orientate itself in space. In addition, robots can have trouble merging the data from two nearby areas, which is not problematic in world-centric approaches.

Since the 1990s, probabilistic techniques and SLAM have dominated the field. Probabilistic approaches all confront the problem of perceptual noise, which can skew mapping data. Each of three approaches – *Kalman filter* approaches, Dempster's *expectation maximisation* algorithms, and *object-identifying* algorithms – model sources of noise in sensory data and show their impact on the measurements.

2.4.1.2 Outdoor

As previously mentioned, there exist several different types of outdoor maps. However, for this review only work revolving around conventional *a priori* created maps will be discussed.

Nagasaka, Y. *et al* [2004] provided such a map to an autonomous rice-transplanting robot. In order for the robot to create its desired trajectory, a representation of paddy field test area was pre-defined as a 50 x 10 m rectangle, with the exact location of each corner accurately measured with an RTK-GPS.

Another such example is work conducted at the University of Cincinnati for a mobile robot competition [Sethuramasamyraja, B. 2003]. The map of the area consisted of the test field's 80 x 90 m boundaries, with specified latitude and longitude coordinates for the robot to traverse. The map was also supplied in Cartesian coordinates on a 2D grid. This map had to be calibrated and supplied to the various teams prior to the competition. The main navigation sensor was the GPS; however, obstacles were placed around the test area by the judges to investigate the obstacle avoidance capabilities of the robots. This was achieved through other active sensors.

Bruch, M. *et al* [2005] used a miniature stereovision system on a small Unmanned Ground Vehicle (UGV) to create a 2D occupancy grid-like map of an outdoor environment without the use of overhead images or maps. The robot achieved sufficiently noise-free results with few false obstacles, but the system had a slow update rate and was limited by low-grade optics.

These methods can be quite disadvantageous, because a significant amount of time is consumed in creating the map of the environment. In addition, the larger the surface area is, the greater the difficulty in calibrating testing areas, hence requiring more points to be accumulated. Using calibrated overhead images for outdoor robot path planning is a promising solution, since it will eliminate the need for acquiring field data, and furthermore will permit speedier testing of various path planning and control strategies.

Wei, Y. *et al* [2004] proposed a building extraction technique from QuickBird satellite images to gather GIS information for mapping applications. The problems of extracting buildings from images range from building-shape complexities, shadows, contrasts between the ground and roof and the spectral characteristics of the roofs' materials. The direction of a shadow in the image is used to determine the presence of a building. Canny edge detection and Hough transformations are used to define building boundaries. A detailed description of Canny edge detection can be found by Ding, L. and Goshtasby, A. [2001].

2.4.2 Aerial Imagery in Mobile Robots

In this section, mobile robots that use various forms of imagery data are presented.

2.4.2.1 Using LADAR/LIDAR and Digital Terrain Models

Research into the use of aerial LADAR images for mobile robots has been conducted at Carnegie Mellon University [Vandapel, N. *et al* 2003]. The images are used for two purposes. One is to determine the robot's localisation using map-matching, and the second is to compute traversable maps for the robot to navigate by filtering out vegetation. LADAR images contain 3D representations of the area, and from these images features such as the vegetation and terrain properties can be determined. These aerial images are obtained from a helicopter equipped with a 3D laser scanner. It is flown several times prior to the robot's deployment for data collection. By combining an aerial LADAR sensor and

a ground LADAR sensor (attached to vehicle), vegetation is filtered. This allows terrain registration to be determined, and thereby reveals traversable routes. The authors' focus is to use these images in conjunction with waypoint navigation.

Fig. 2.4.2.1a shows the vegetation map of the area, and Fig. 2.4.2.1b shows the traversability map of the area after the vegetation has been filtered using their technique.



Fig. 2.4.2.1a: Vegetation map of test area. The green (light coloured) areas represent no vegetation and the red (dark coloured) represent high vegetation. The black path represents the robot's path.



Fig. 2.4.2.1b: Traversability map of test area. The green (light coloured) areas represent the traversable area and the red (dark coloured) areas represent the non-traversable area. The black path represents the robot's path. This image was created after vegetation filtering.

A similar approach was used by Kelly, A. *et al* [2006] for a mobile robot in an off-road environment. The authors utilised a UAV to accumulate date in open-field environments.

Only the LADAR 'hits' from the UAV that penetrated to the ground were recorded, thereby rejecting the vegetation canopy and allowing the mobile robot to follow waypoints beneath trees.

2.4.2.2 Using Aerial and Satellite

Sofman *et al* [2006] developed an algorithm for vehicle traversing in outdoor environments using a self-supervised learning method using pre-obtained overhead images. A linear probabilistic model is used to learn and fuse the data estimates from both the overhead data (including elevation data) and an onboard perception system. The robot is then able to compute terrain costs of different map areas and extrapolate this information to predict traversal costs for new overhead images, thereby extending its local perception system.

Work conducted by Silver, D. *et al* [2006] at Carnegie Mellon University also involved the use of overhead images for robot navigation. This differs from the work of Vandapel, N. *et al* [2003] because the images were obtained from overhead image providers and not via a LADAR equipped robotic helicopter. In this approach, the authors use these overhead images to produce traversal cost maps offline, which are computed from the combination of geometric and semantic data. Semantic data is computed from features extracted from imagery and 3D data through supervised classification.

The authors refer to this method as an *aided exploration scenario*. With the combination of perception data, they were able to create a vehicle capable of traversing rugged terrain for long ranges autonomously.

The authors state that one drawback of using overhead data is that it is often accompanied by heterogeneous resolution, sampling time and sampling pose, and georeferenced data with insufficient accuracy. Similar work involving traversability maps in order to create a global path planner for a mobile robot was achieved by Howard, A. *et al* [2005]. A fuzzy logic set rule is used to determine features from aerial images, which are then passed on to the traversability mapbuilding algorithm to determine traversable indices, which also use a fuzzy-logic rule based system. Followed by this step, a traversability grid is created (see Fig. 2.4.2.2a). The white coloured cells are considered to be highly safe. The light and dark coloured grey areas are considered to be moderately safe and moderately unsafe, respectively. The black cells are considered to be highly unsafe. Then a search algorithm is used to determine a fixed set of waypoints for an optimal route for mobile robot navigation.



Fig. 2.4.2.2a: Overhead image, traversability map and traversability grid.

Sandia National Laboratories, a leading centre for mobile robot applications, has developed a system wherein the user controls the positions of multiple robots for strategic military operations (see Fig. 2.3.2c in Section 2.3.2). The system was developed so that a soldier can define each robot's goal via a set of waypoints on an aerial image through a user console, and can define avoidance regions by giving them a repulsive field [Feddema, J. *et al* 1999].

A similar approach was adopted by Bruch, M. *et al* [2002]; however, it was clearly stated that the aerial image used was georeferenced and orthorectified, which are two vital processes for achieving accurate waypoint navigation. Once more, the application is intended for military use. The authors claim that accurate results were achieved by coupling the GPS with IMUs, in addition to an odometer with a Kalman filter, without the need of differential corrections (DGPS).

There was no clear description of the duration of these tests, and no quantitative estimate of the accuracy. The authors claim that the accuracy of the positional data depends on the resolution quality of the image, which is debatable since tests have shown that the image ground position does not necessarily match to the true ground position, irrespective of the resolution, as will become clearer in section 2.4.3 and chapter 4. In the investigative work carried out, it is shown that relying solely on the accuracy of the image does not necessarily yield accurate positioning. The architecture of tracked robots does not permit them to travel at high velocities, and therefore a greater number of minor deviations from the path are expected, compared to wheeled robots. See Fig. 2.4.2.2b for a depiction of the user interface used in their work, showing the robot's path in blue.



Fig. 2.4.2.2b: User interface for controlling URBOT.

For non-military applications, Muscato, G. *et al* [2003] used a Graphical User Interface (GUI) for a robotic volcano explorer, ROBOVOLC (see Fig. 2.4.2.2c). Correspondence with Muscato revealed that aerial images have only been used for simple tests, as most of the operations were either carried out by teleoperation or by fixed waypoints. It is clear that the images are not intended for path planning, possibly because terrain features and

landmarks can change rapidly in harsh environments, rendering them useless. Furthermore, if no detailed topographic map of the environment is presented, the user could misjudge steep unforeseen groves, thereby endangering the robot.



Fig. 2.4.2.2c: ROBOVOLC waypoint command window.

An interesting approach is employed by Zein-Sabatto, S. *et al* [2004], wherein a satellite image is digitised to obtain a 2D image (map) for mobile robot path planning. Variations in the landscape are differentiated by colour. The images are then converted to greyscale to avoid having to process the RGB (red, green, blue) variations in each pixel. After they are discretised, a grid-map is created and the robots' paths are generated using the developed genetic algorithm. Once the planned routes are processed, a 3D vector representation of the image is created.

The drawback of such a system is that significant detail of the image is lost due to discretisation (Fig. 2.4.2.2d). This might prove to be a successful approach for path planning over larger distances in which little attention is given to the local field; however, too much detail is lost for finer image areas to be useful for precision applications. Because

no test results were presented about the system's accuracy for practical applications, it shows that the need for precision in imagery is of no relevance to their research.



Fig. 2.4.2.2d: (Counter-clockwise from top). Aerial Image, discretised grid, 3D model.

Finally, the latest work by Meguro, J. *et al* [2005] from Waseda University in Japan adopts such a system for an autonomous robot (Fig. 2.4.2.2e). The user selects a goal on an aerial image and the path planner MAKLINK calculates the fastest route between the start and finish. The method yields vehicle-tracking error of less than 0.25 m in a city environment.

Of interest is the fact that the path planning aspect depends largely on the accuracy of this aerial image. MAKLINK finds the shortest route from start to finish by taking into account the respective locations of the obstacles in space. The obstacles in this case are the buildings, and since manual calibration of the image on its four corners yields accuracies of only 1-4 m, it is discarded as too inaccurate for the robot's applications. To accommodate for image inaccuracies and to make MAKLINK operate reliably, 34 corners of the respective buildings in the image are precisely surveyed with an RTK-GPS to define the obstacle boundaries accurately for route calculation.


Fig. 2.4.2.2e: (Counter-clockwise from top) User console, precisely surveyed building corners, MAKLINK.

This leads to the fact that, until now, no highly-accurate map representations of either rural areas or streets exist despite the fact that the creation of accurate digital road maps is a heavily researched area, and probe vehicles are used to create them. Several techniques are being adopted to meet the growing demand for accurate digital road maps for car navigators, and for potentially using them in autonomous domestic cars. [Rogers 2000; Wang, J. *et al* 2005].

One approach that Meguro could have used for the system is a map-matching algorithm like that found in car navigators. However, if streets are very close to each other and run parallel to one another, inaccurate map matching could occur [Kim *et al* 2001].

Satellite and aerial images have also played a role in navigation for aircraft positioning. Sim, D. *et al* [1999] proposed a method for determining the absolute and relative positions of an aircraft using these images. Absolute position is determined by matching previously stored images to the newly captured images covering the same area. Relative position is determined by taking two images in quick succession, and from the resulting stereo image, the aircraft's displacement using roll, pitch, yaw, and altitude parameters can be calculated.

2.4.3 Essential Imagery Considerations

2.4.3.1 Aerial Images

As mentioned previously, a recent trend in navigation and area representation methods has been the use of various types of imagery. This work will focus on the use of aerial images (photographs) and not 3D Digital Elevation Models (DEM) such as LIDAR/LADAR, since low-cost GPS units do not provide accurate altitude data. Freely available or low-cost imagery (e.g. *Google Earth*) can be several years old and of variable image resolution, rendering it useless for many applications; yet, freely available data remains useful for conveying the landscape for various purposes.

There are many types of orbital satellites which collect images, such as Landsat, SPOT and IRS; however, most have a lower resolution (i.e. less detail) than the recently-launched IKONOS and QuickBird [Mumby, P. *et al* 1997]. The latter two were developed to provide high-resolution imagery for both civil and government use. Many (>30) new remote sensing satellite systems are now operational in addition to 12 further planned launches within the next year, which boast even higher image resolution and positional accuracy [Sensors & Satellites 2007]. IKONOS provides spatial resolution of up to 0.8 m panchromatic (i.e. greyscale) ground sample distance (GSD) and 4 m multispectral (i.e. colour) GSD, whereas QuickBird's resolution is sharper at 0.6 m and 2.4 m [Wu, J. *et al* 1999; Dial, G. *et al* 2003]. Several agencies sell these high-resolution images; however, they are often too expensive for the average user, as a minimum purchase area applies.

Aerial photographs provide a useful alternative to satellite imagery, because they have the advantage of being acquired at closer-range than satellites, and consequently provide higher scale and detail/resolution. These two attributes are necessary to assist enhanced

waypoint identification. For example, an aerial photo taken at 300 m above ground level with the usual 150 mm focal length lens has a resolution on the ground of 0.08 m per pixel [Booth, D. *et al*, 2006], which is more precise than both IKONOS and QuickBird. However, the need for airplane transport (or the commission of individual flights) can make these images expensive [Trisirisatayawong, I. *et al*, 2004].

Another low cost approach for acquiring aerial imagery is a system for remote sensing in times of disaster [Oh, P. *et al*, 2004], which could be used for waypoint-based navigation. In this, a mechatronic kite equipped with a teleoperated camera, video transmitter, battery, remote controlled receiver and two servos (for pan and tilt) have been used for live data capture. It can be deployed rapidly, is lightweight and can quickly obtain images. Finally, another method of capturing aerial images includes using an Unmanned Aerial Vehicle (UAV) that obtains aerial LADAR data [Vandapel, N. *et al*, 2003]. Irrespective of the image used, post-image processing is required for georeferencing.

2.4.3.2 Georeferencing

Georeferencing is the process by which the image is related to a suitable ground coordinate system. Since the earth is not a perfect sphere, setting these factors to a fixed universal mathematical index such as the widely used World Geodetic System 1984 (WGS84), could lead to inaccuracies of several metres, depending on the geographical location of the image in the global frame [Ordnance Survey 2007]. This leads to two concepts: *Map Datum* and *Map Projection*. It is important to set the aerial images to the datum and projection used to represent the country in which the image was taken. In the UK, for example, the map projection used is known as the Transverse Mercator (TM) and the Map Datum as the *Ordnance Survey Great Britain 1936*, which is based on a geographic representation known as the *Airy 1830* ellipsoid. Direct transformations between various map datums (e.g. OSGB36 to WGS84) can be achieved using for example the Helmert Transformation. Unfortunately, such transformations are only approximate at the local scale. In the UK, for example, small scale inaccuracies arising from the 1936 re-triangulation lead to significant

positional errors up to 20 m [Ordnance Survey 2007]; therefore using simple global transformations and published constants is not advisable. It is important to ensure that a consistent underlying coordinate system for the aerial image being used, and that the GPS positional output matches its corresponding location on the image.

2.4.3.3 Photogrammetry

The science developed to relate measurements of imagery to a ground coordinate system is known as photogrammetry [Fryer, J. et al 2007], the impetus for development being primarily the production of the World's National Mapping series [Wolf, P. et al 1983]. There are two types of distortion inherent in any aerial or satellite image, which prevent direct correspondence between the 2D image and 3D ground coordinate system: tilt and relief distortion. Distortions that are created by the light rays leaving the object, passing through the lens centre before creating an image point in the focal plane of the camera, are modelled explicitly using the collinearity equations [Fryer, J. et al 2007; Wolf, P. et al 1983]. These equations model completely distortions due to non-verticality of the sensor. A distortion is also introduced into the image if the terrain is non-planar. Such 'relief displacements' are related to the flying height and focal length of the sensor and can be highly significant for aerial photography. Only a true orthorectification procedure implementing the collinearity equations removes the distortions due to both relief and tilt displacement. Unfortunately, there are various of aerial image products which have not been generated using the required rigorous mathematical procedures. Although such 'map accurate' products are fit for many purposes and applications, they should always be used with caution, particularly when used in conjunction with GPS. Post-image processing is required for both types of images to adjust for camera perspective. Distortions are inherent in satellite images because they 'see' distant objects at an angle; thus, objects directly below it appear larger and upright, whereas further objects can appear at a side-angle [Zhou, G. et al 2005].

The orthorectification procedure can be accomplished by using Ground Control Points (GCP) clearly visible on the aerial images. The 3D coordinates of the GCPs should be

established using a survey grade differential grade GPS and linked to the Ordnance Survey (OS)'s passive network. These coordinates should be subsequently transformed to OSGB36 using the OSTN02 and the OSGM02 models provided by the OS [Ordnance Survey 2007]. Unfortunately, there is no single solution available and different approaches are required in different countries. Advice should be sought from National Mapping Agencies.

The process of orthorectification can introduce discrepancies if the Digital Elevation Model is inaccurate. Therefore, it is important to consider such uncertainties when judging the inaccuracy of the waypoint selected from an aerial image.

2.4.3.4 Mapping Updates

The Geographic Information System (GIS), for the storing and handling of geographically encoded data, is vital for continually updating maps [Nemenyi, M. *et al* 2003]. Digitised satellite/aerial images can be easily added to GIS databases. The emergence of GPS and its accessories over the last decade has allowed researchers to produce more accurate georeferenced images for validation of earth observation (EO) data [Budkewitsch, P. *et al* 2004]. This has been easier in Japan with the cm-level accuracy of Mitsubishi's PAS system for satellite image mapping [Higuchi, H. *et al* 2004]. For the less fortunate countries, Satirapod, C. *et al* [2003] developed a system using a regular dual-frequency GPS to establish accurate GCPs for mapping reference. The performance of this technique was tested over a 15-minute time span, and yielded accuracies within 2.5 m for a satellite image of medium to high resolution.

2.4.4 Map Representation Summary

Broadly stated, a 'map' in a mobile robot is a collection of data that represents the environment that it will explore. The following map considerations have been discussed in the above section:

- a) Conventional maps. These are maps that are created by the robot as it explores the given environment. Conventional maps can be used alone or in conjunction with a priori maps or sensory information.
- b) Aerial imagery. Past researchers have used aerial imagery extensively for a priori map information. LADAR/LIDAR and DEM have been used particularly in complex outdoor areas, in which vegetation filtering is necessary. Satellite and aerial images have also been used for traversing outdoor environments, and aid in the calculation of optimal routes.
- c) Georeferencing and Photogrammetry. The processes and science behind the relation of images to suitable Ground Coordinate Systems has been discussed at length. Orthorectification – the method by which image inaccuracies are corrected – is also described.
- d) Finally, techniques for map updating have been discussed.

It has been shown that, in mobile robots, no significance has been given to criterions behind the selection of imagery for waypoint navigation. Elements such as the choice of appropriate underlying coordinate system, and spatial matching between GPS data and the orthorectified images have not been investigated. This will be explored further in this research.

2.5 Literature Gap

After a thorough review of the relevant literature, it can be seen that there has been a consistent interest in mobile robots, particularly in the fields of localisation, path planning, navigation, and map representations used for the above. In recent years, aerial images have become more topical as accuracy and availability have increased. Despite the promising developments in these areas, there exists a distinct lack of progress in a few key topics.

Though interest in synchronous drive robots was seen in the 1990s, the hardware and software technology available on former synchronous drive platforms has since been outdated [Chopra, A. *et al* 2006]. This has produced limited literature with such a robot mechanism over the past six years. In addition, most of this past research focussed on indoor robots for the study of obstacle avoidance and odometry.

Thus far, research using satellite positioning systems for mobile robot localisation has relied on sensor coupling to improve accuracy. To a lesser extent (and with variable results), the GPS has been used as a stand-alone application for positional information. However, there still lacks a body of research utilising differentially-corrected signals using geostationary satellites (WAAS/EGNOS) alone for mobile robot localisation and navigation. With recent experimentation showing that accuracies to within 3 m can be achieved, there exists an exciting opportunity for development in this area.

Furthermore, there is a lack of literature about novel ways of correcting GPS inaccuracies in mobile robot localisation and navigation. In general, researchers employ additional sensors (such as IMUs) to detect and correct inaccuracies. Preliminary work by Wuersch, M. and Caduff, D. [2005] inspires a new way of achieving refined route instructions based on the proximity of a GPS sensor to an upcoming waypoint. By extending this idea and implementing it in the Spider, the current void in this area of research could be narrowed. The proposed work will therefore involve the use of orthorectified aerial images for waypoint path creation. Errors associated with the waypoints will be investigated. A new navigation technique will then be implemented for the robot, which accommodates for the GPS inaccuracy and waypoint error and would use a modified version of the circular stages of closeness between the sensor and the waypoint [Wuersch, M. and Caduff, D. 2005]. Their proposed method was developed for handheld GPS devices to provide users with refined route instructions, based on the proximity of the sensor to the waypoint. This navigation method will then be accompanied by a novel fuzzy controller strategy to ensure smooth motion between the waypoints.

The assumed working environment will be 2D, and, therefore, relatively flat landscape will be used. The projected use of this system in its current form will be in open-space agricultural environments and uncluttered urban landscapes.

In order to test the validity of the novel system, a simulation of the robot's behaviour is needed to provide a means of comparison to the actual test results. This will introduce a novel Matlab Simulink simulation that accommodates for each of the previously mentioned factors. The introduction of accurate aerial images (in addition to other important considerations not previously mentioned in similar research) will provide better qualitative and quantitative results for validating the robot's true performance in waypoint navigation.

The conversion of the Spider to an autonomous robot for agricultural transport using GPS and aerial images, along with a unique waypoint following fuzzy controller, introduces novelty in multiple aspects and confirms its membership in a multi-disciplinary engineering research topic.

Chapter 3: Spider Technical Specifications

The main purpose of this chapter is to introduce the platform and some of its components used for this research.

The chapter will cover the following:

- a) A brief introduction to the grass cutting mower.
- b) Some design limitations behind its drive and steering mechanism.
- c) Platforms of a similar architecture.
- d) A brief description regarding the additions (transformation) made to the platform.
- e) Finally, reveal the data handling methodology for this research.

Chapter pre-requisite: None. Nature of Chapter: Technical.

3.1 The Spider

In August 2005, Ransomes Jacobsen Ltd. introduced a remote-controlled grass mower with a four-wheel drive, 18 HP four stroke petrol engine (see Fig. 3.1). The manufacturers claim that it is six times more efficient than a string trimmer. The Spider is designed to be able to mow slopes of up to 40° , making it useful in areas too dangerous for humans. The mower has a length and width of approximately 1.3 x 1.3 m, a height of 0.85 m, and weighs 254 kg. It can drive with speeds up to 7 km/h. In addition, the Spider has a synchronous drive mechanism, which means that the wheels can steer 360° continuously and unhindered in both clockwise and counter-clockwise directions. It has an 11-litre tank, giving it an outdoor operation time of up to four hours [Ransomes-Jacobson 2005].



Fig. 3.1: Ransomes Jacobsen's Spider [Ransomes Jacobsen 2006]

3.1.1 Design Limitations

Even though there are some benefits of petrol driven robots over electric ones, there are some design limitations that accompany this mower, which are important to consider during the testing and analysis stages.

The Spider is subject to both internal and external (environmental) constraints that affect the overall operation of the mower. The drawback of a petrol driven engine is the amount of vibration produced, which can inevitably cause loosening of some mechanical couplings and which can prevent proper functioning of sensors. Along with external constraints such as variation in terrain conditions, slack in the chain drive steering system can occur with time, impeding the accuracy of the robot's steering. This flaw in synchronous drive mechanisms has been previously observed [Borenstein, J. *et al* 1996], and it results in wheel misalignment that requires periodic readjustment by the user, which is a relatively time consuming process. Borenstein, J. *et al* propose that a design with a completely enclosed gear-drive would eliminate this problem and furthermore reduce generated noise. Fig. 3.1.1 below reveals the Spider's belt drive (translation) and chain drive (steering) mechanism.



Fig. 3.1.1: The chain and belt drive mechanism of the Spider

Theoretically, it is possible to achieve a specific pose for the robot since the wheels' orientation and translation are decoupled [Fox, D. et al 1997]. In practice, however, the

decoupling concept does not fully apply to the Spider. The steering is achieved via an electric motor and the translation by means of a hydraulic drive. After conducting several test runs with the mower, it was concluded that steering the wheels when the Spider is stationary is not possible – as opposed to indoor synchronous robots. This can be traced to the lack of torque in the electrical steering motor, a high coefficient of friction between the wheel and the terrain (largely due to the size of the wheels and furthermore the weight of the Spider and the addition of a platform), and finally the stage of the battery's life. An attempt to force this motion would create excessive forces on the chain drive system leading to slack and eventually damage.

Given that this is an industrial product, design limitations would have been imposed by the manufacturers. One such limitation is the inability to control the engine's throttle directly. The mower was designed to operate with fixed velocity command outputs for both the electric motor (steering) and the engine (translation). This design restriction acts as a limitation but not a hindrance to the selection of an appropriate controller.

Finally, another drawback is the hydraulic driving system powered by the engine. After the Spider has been in motion and comes to a stop, it is observed that it tends to 'creep'. This is believed to be due to the hydraulic pump not fully retracting to its original state, leading to a small amount of hydraulic fluid still being pumped into the driving system. By trial and error, it was noted that this effect can be cancelled out by briefly throttling the engine (remotely) into the opposite direction of travel, once it comes to a halt. This problem was also observed by the manufacturers and with the replacement of the hydraulic flow valve, supplied by them, the issue can be corrected.

Any further limitations from experimental tests that arise will be elaborated in the discussion.

Because of the known limitations of the Spider's current mechanical configuration, it can be projected that sub-metre level accuracy will be more difficult to attain than conventional battery-powered indoor synchronous drive robots.

3.2 Similar Robots (Technical Differences)

There exist various types of synchronous drive robots that are used in research; some of the most widely used platforms are the B21, and the Nomad 200 as mentioned previously. Some have a three-wheel configuration and others have four.

However, they all share the same feature of having reduced odometry error compared to other mobile robots, since the wheels are typically coupled with chain drives for synchronous steering and heavy duty belt drives for synchronous translation. However, since the majority of synchro-drive robots are chain and belt driven, slack may occur during the course of operation and hence degradation in steering accuracy, leading to wheel misalignment [Borenstein, J. *et al* 1996]. Contrary to the Spider, however, these robots are intended for indoor use.

Researchers coordinated by the Danish Institute of Agricultural Sciences have managed to secure €438,600 in the development of their modified version of the Spider (see Fig. 3.2a). They altered its general architecture to incorporate several steering wheel configurations, such as double Ackermann and crab steering, in addition to its synchronous drive mechanism. The research team's ideal goal is to transform the Spider into a tool carrier for plant nursing. Their website presents various videos demonstrating some of the modified Spider's capabilities [HortiBot 2007].



Fig. 3.2a: A schematic of their modified Spider, the HortiBot [HortiBot 2007].

In addition, the University of Warwick have also added features, such as GPS, vision systems, radars and odometry to the Spider in order to examine ways of automating lawnmowers for large scale landscapes such as golf courses and parks. Its steering architecture has not been modified from the standard synchronous drive mechanism. Please refer to Fig. 3.2.b.



Fig. 3.2b: Spider robot modified by the Robotics team of the Warwick Manufacturing Group, Warwick University [WMG 2007].

Currently the robot on the market that most closely resembles the Spider is the Seekur[™] developed by Mobile Robots Inc. [Mobile Robots Inc. 2007], which has omni-directional steering capabilities (including synchronous), is electrically powered, and can operate continuously for up to seven hours (see Fig. 3.2c). However, one drawback is a charging time equivalent to its operation time. The Spider, on the other hand, is petrol driven, with a continuous operating time of up to four hours, with no power degradation over the course of operation. Furthermore, the 'recharging' time is only the time needed to re-fuel the robot. When the Seekur[™] is in synchronous mode, the robot's orientation does not change. In addition, the absence of a rotating platform or turret does not enable the camera to point in the direction of travel. This leads to difficulties if vision is used for localisation or navigation.



Fig. 3.2c: Seekur[™], the indoor/outdoor all-weather robot [Mobile Robots Inc. 2007]

3.3 Spider Transformations (Accessorisation)

Given that the Spider was not intended for autonomous navigation, a few transformations had to be made to enable this transition.

Since the Spider has a synchronous-drive mechanism, the orientation of the mower never changes – unless wheel slippage occurs. So, in other terms, the frontal side of the robot will always be facing in the same direction irrespective of the wheels' change of orientation. This issue was tackled with the Nomad 200 by introducing an independently rotating turret that housed the sonar sensors, the camera and the main processing unit [Chopra, A. *et al* 2006]. A similar mechanism was adopted for the Spider, in which a rotating platform was built that is synchronised with the wheels' steering direction and angular velocity in its initial stages (see Fig 3.3a). This system ensures that the cameras (vision and infrared) will always be facing in the direction of travel. In order to reduce the load on the Spider, the main frame was built using aluminium bars and the sensor base and rotating platform using machinable hard nylon sheets.



Fig. 3.3a: The rotating platform developed for the Spider (in the initial stages)

Another issue that needed to be solved in order to ensure that the system was transformed into an autonomous robot was the hardware's power requirements. The Spider is equipped with a 12V gel-based battery that can deliver 12 Amps/hour. The main power-consumers in the Spider are: the steering servomotor, the hydraulic pump and its servomotor, the engine's throttle servomotor, the RF transmitter, the ventilator fan, and the main control unit. The battery is charged by means of the 13 Amp alternator that the Kawasaki engine is equipped with.

However, for the tests conducted in this research, an alternative 12V battery is mounted to the aluminium platform to supply the necessary power to the components used for this research.

3.4 Robot's Hardware

A variety of hardware exists that could be suitable for converting the Spider into an autonomous robot. For this research, the focus was made on the particular set of instruments needed to achieve the research objective. However, other equipment that would facilitate future research activities with the Spider had been mounted and carefully set up under supervised projects carried out at the lab. These can be referred to in Appendix A. Fig. 3.4 shows a view of all the components mounted to the Spider's platform.



Fig. 3.4: Hardware layout split between upper and lower area of the platform

3.4.1 Data Handling Equipment (Acquisition)

In this section, the data handling equipment used for the research is presented:

3.4.1.1 IENSYS General Purpose Microcontroller Board

The IENSYS GP board is a general-purpose microcontroller used for sensory data acquisition and for transmitting data through the Bluetooth to the host PC for overall data handling (see Fig. 3.4.1.1). Two of these boards are used on-board the Spider, as is noted in Fig. 3.4.3. The boards used for the Spider have a 7.372 MIP processor. The microcontroller chip used is a PIC 18F458 from Microchip Technology Inc. The board can handle various communication protocols such as CAN, RS232, and SPI. The board can perform A/D conversions, create pulse width modulated signals (PWM), and perform various other features not commonly available on other off-the-shelf microcontrollers [IENSYS Ltd. 2006].

There exist many PIC microcontroller software programs; however, for this project MikroBasic [MikroElekronika 2006], Microchip's C18 compiler and Assembly package [Microchip Technology Inc. 2006] were used.



Fig. 3.4.1.1: IENSYS General Purpose microcontroller

3.4.1.2 Wireless Bluetooth Adapters

The Initium Promi SD202 is an RS-232 adapter that can be seen in Fig. 3.4.1.2. It is capable of supporting baud rates of up to 230400, data transmission rates up to 380kbps, and allows for a range of security settings. The range between two adapters equipped with the default antenna can reach up to 100 m, and with the dipole antenna replacements up to 200 m. Since a Laptop's Bluetooth wireless manager, and two dipole antennas and one standard antenna are used the projected range is expected to be greater than 100 m.



Fig. 3.4.1.2: RS232 Bluetooth adapter [Initium 2006]

3.4.2 Sensors

The Spider is equipped with several sensors necessary to achieve the research objective, which are detailed below.

3.4.2.1 GPS

The function of the GPS is to provide the user with positional data in the world frame.

The Garmin 18-5 Hz is a 12-channel, WAAS-enabled GPS receiver that has a sampling rate of 5 Hz (see Fig 3.4.2.1). This sensor is specifically designed to be used in machine control, agricultural applications and guidance that require velocity and position reports at 5 Hz. The baud rate can be set between 300 and 38400 bits/second. The accuracy of the Standard Positioning System (SPS) is better than 15 m, 95% of the time and velocity accuracy of approximately 0.05 m/s RMS steady state. With WAAS enabled, the position is accurate to within 3 m (or better, as will be shown in *Chapter 4*), 95% of the time, with a velocity accuracy of also 0.05 m/s RMS steady state. Furthermore, it has a real-time clock that is used to time-stamp the remaining sensory data for more precise control and accurate post data analysis. Since WAAS receivers are compatible with EGNOS, the signals will be received from the latter (European coverage) [Garmin Ltd. 2007].



Fig. 3.4.2.1: The WAAS enabled Garmin 18-5Hz GPS receiver

3.4.2.2 Potentiometer

In order to determine the orientation and the angular velocity of the wheels during steering, a cost-effective potentiometer was chosen as an alternative to an encoder. The practical one-channel output makes it a favourable option. The 360° potentiometer by Spectrol[®] was chosen for this application. It boasts stable output and low power consumption - two

favourable options for an outdoor autonomous vehicle. It has a dead band of 50 microseconds. The product is marketed for rotational control systems and angular feedback applications. The potentiometer is mounted to a small purpose-built platform (see Fig. 3.4.2.2).



Fig. 3.4.2.2: The 360° potentiometer mounted on one of the wheels

Furthermore, the potentiometer is used with a secondary potentiometer mounted to the rotating platform for closed loop control of the platform's positions.

3.4.2.3 Relative Optical Encoder

Another sensor for providing *proprioceptive* data of the robot is the relative optical encoder shown in Fig. 3.4.2.3a. It will be used as a back-up for determining the actual translational velocity of the Spider if the GPS signal is temporarily lost. The velocity output will not be used as a feedback in the control strategy but will be used to determine whether the controller velocity output matches the actual translational speed of the robot.



Fig. 3.4.2.3a: Optical sensor for providing wheel's translational velocity

Fig. 3.4.2.3b shows the location of the potentiometer and the relative optical encoder on the purpose-built platform.



Fig. 3.4.2.3b: The Spider's relative optical encoder and potentiometer

3.4.3 Spider's Data Handling Hardware

Fig. 3.4.3 is a general schematic showing the hardware layout for the data handling equipment used for the implementation of the control system in this research.



Fig 3.4.3: General schematic of the data handling equipment and procedure

3.5 Robot's Software

3.5.1 General Software Layout

Fig. 3.5.1 shows the general system architecture used in this research. The layout is shown in a simplified three-step process: in step 1 the image is first acquired. Step 2, the necessary processing is done. Finally, in step 3, the program is implemented for testing.



Fig. 3.5.1: General system layout

3.5.2 Off-board Data Processing

Visual Basic (VB) was chosen as the preferred operating program due to its user-friendly interface development, numerous plug-in tools, and the ability to call other applications in its own environment. The development of the control algorithm was made in Matlab and the variables are passed through the Matlab engine that runs parallel to the VB software. Fig. 3.5.2a is a schematic showing the data handling within the software. Each box within the VB application schematic represents a unique control interface that operates independently, but they are linked internally through the controller application. Fig.3.5.2b shows a screenshot of the VB user-interface created.



Matlab engine



Date Time Time Latitude Longitude Longitude Northing Ratenna Height Antenna Height True Course Magnetic Variation True Course Magnetic Course GPS Fix Statilites HDDP PDOP Speed Km/h Start Com Biticon/Idle Indexterename Biticon/Idle			GPS	•				Wheel Da	ita
Time EVPE m Latitude m Longitude m Nothing m Resting m True Course degree GPS Fix m Satelites m/s HOOP m/s POOP m/s Start Com Start Com Start Com Start Com Stop Stop GPS Stop Start Com start/Stop Stop mrules remang mrules remang	Date			EHPE			- Tai	salacement	degree
Latitude Image: Cause <td>Time</td> <td></td> <td></td> <td>EVPE</td> <td></td> <td>m</td> <td>1.</td> <td></td> <td></td>	Time			EVPE		m	1.		
Laitude Magnetic Variation Longitude m Nothing m Antenna Height m Easting m BPS Fa m Satelike m HDDP m/x PDDP m/x Speed km/h Km/h East Velocity m/x Noth Velocity m/x Start Com Start Com Stop Stop GPS Stop Stop Stop Stop Stop mxde: remaing m/x Notic correction m/x				EPE		10	di	rection	
Longitude In rue course Nothing n Easting n True Course degree GPS Fax n Satellite: n HDOP n PDOP n Speed km/h East Velocity m/s Stant com Stant co	Latitude	_		Magnetic Variatio			10	near velocity	m/s
Nothing in Interformer (or other (or	Longitude		-	Antenna Height		-		auto unto alter	radia
Easting in the course degree EPS Fix Image: Inde course degree Satelliter Image: Inde course degree HDOP Image: Inde course degree PDOP Image: Inde course degree Speed Image: Inde course Image: Inde course Speed Image: Inde course Image: Inde course Speed Image: Inde course Image: Inde course Start Image: Inde course Image: Inde course North Velocity Image: Inde course Image: Inde course Start Image: Inde course Image: Inde course GPS Stop Image: Inde course Image: Inde course Start Image: Inde course Image: Inde course GPS Stop Image: Inde course Image: Inde course Start Image: Inde course Image: Inde course GPS Image: Inde course Image: Inde course Start Image: Inde course Image: Inde course GPS Image: Inde course Image: Inde course Start <td< td=""><td>Northing</td><td></td><td></td><td>Tue Course</td><td></td><td>lama</td><td>1.00</td><td>igular velocity</td><td></td></td<>	Northing			Tue Course		lama	1.00	igular velocity	
GPS Fa agree Satellike: agree HDOP agree PDOP agree Speed km/h East Velocity m/s Noth Velocity m/s Start GPS Com Start GPS Com Start GPS Stop Com Stop GPS Stop Lonomous Notic correction Start/Stop mrute: remaining	Easting			Magnetic Course		legice	Alteres		
Satellite: HDOP PDOP Speed East Velocity Noth Velocity Noth Velocity Start Com Start Com Start Com Stop Stop <t< td=""><td>GPS Fix</td><td></td><td></td><td>Magnetic Course</td><td></td><td>regice</td><td></td><td></td><td></td></t<>	GPS Fix			Magnetic Course		regice			
HDOP PDOP Speed kn/h East Velocity n/s Start GPS Stop GPS Stop GPS nic correction Stop Stop Stop Stop Com mrute: remaining million/Idle	Satellites								4
PDOP Speed East Velocity North Velocity North Velocity Stant GPS Stop GPS Stop Com Stop Stop Lipinition/Idle	HDOP							Com	troner
Speed km/h East Velocity m/x North Velocity m/x Start m/x Start Com Stop Stop GPS Stop Stop Com Stop Engine nrute: remaining Imrute: remaining	PDOP							RB	
East Velocity m/z North Velocity m/z Start GPS Start Com Stop GPS Start Com Stop GPS Stop Com mrute: remaining Imrute: remaining	Speed		km/h			100	82	No of Waypoints	
North Velocity m/s Start GPS Start Com Stop GPS Stop Com Stop GPS Stop Com mrute: remaining Implicition/Idle	East Velocity	2.	m/s					Europe Malasite	
Start GPS Start Com Start Com Start Com Start Autonomous Fuzzy Angular Stop GPS Stop Com Engine Ignition/Idle Start Autonomous dc	North Velocity		m/s			1		ruzzy velocity	
GPS Com Autonomous dc Stop GPS Stop Com Engine Ignition/Idle Stop Autonomous dc	Start	Start			Start			Fuzzy Angular	
Stop GPS Stop Com Image: Common and Common an	GPS	Com	1		aran				
Stop GPS Stop Com Engine Ignition/Idle Stop Autonomous mude: remaining Ignition/Idle Stop Autonomous				0	Autonomo	us II		0C	
GPS Com atic correction Start/Stop Stop Autonomous Emergency STOP S	Stop	Stop	1 110	1	Part along the				
atic correction Start/Stop	GPS	Com	4		Stop				
atic correction Start/Stop				ngine	Autonomo	us		EMERG	ENCY
Startstop	atic correction	minutes remaining	Ign	ition/Idle		10000			OP .
	Stanzstop	i successione		interior de la competition					

Fig.3.5.2b: Screenshot of VB software used for the Spider

There are several benefits to using a human/robot interface. They can make new technology more accessible, which helps to ease the integration of new technology in people's lives [Hollingum, J. 1999]. An interface also allows a human in-the-loop to intervene if a robot encounters problems that are beyond its capabilities.

Furthermore, the collaboration of human/robot skills can yield better results than either one alone. Bechar, A. and Edan, Y. [2003] found that melon-picking robots increased their melon detection by 4% when humans were involved. Shiller, Z. and Gwo, Y. [1991] created an early interactive computer program that was used for optimisation of a local robot path after a global search was performed. Stentz, A. *et al* [2002] developed an 80/20 system, wherein 80% of tasks were classed as being "easy" and were delegated to the robots, and 20% were deemed beyond robot reasoning and were better managed by the human. At present, full robot autonomy cannot be attained, and a functional interface helps to bridge the divide between man and machine.

Thus far, user interfaces have had wide-ranging applications, from teaching purposes [Elnagar, A. and Lulu, L. 2004] to the management of multiple robots by a single human operator [Parasuraman, R. *et al* 2005].

3.6 Chapter 3 Summary

- a) The Spider is a petrol driven synchronous drive grass cutter that is being used as a robot for autonomous navigation.
- b) Other research institutes have used the Spider platform for research into autonomous vehicles.
- c) The Spider's disadvantages are: slack in the drive chain that leads to wheel misalignment; vibrations, which can lead to the loosening of mechanical couplings and the improper functioning of sensors; and the inability of the Spider to exploit its full synchronous drive (decoupled linear and angular velocity).
- d) The robot has been equipped with a rotating platform and numerous hardware components for a distributed network of servers; however, for this research a GPS receiver and potentiometer will be the main sensors in use, that are linked together by a network of Bluetooth transceivers via a host PC.
- e) A custom programmed Visual Basic application that works in conjunction with a Matlab engine, running parallel to the program, has been created.

Chapter 4: System Components – Preliminary Test Results

The purpose of this chapter is to introduce some of the limitations faced with the use of aerial imagery and low-cost GPS for mobile robot waypoint navigation.

A series of tools were used in order to demonstrate:

- a) The waypoint accuracy, which can typically be obtained from an orthorectified aerial image.
- b) The effect of GPS (WAAS/EGNOS) positional variation and the proposed positional correction approach on reaching waypoints.

Chapter pre-requisites: 2.2.2 The Global Positioning System and 2.4.3 Essential Imagery Considerations.

Nature of Chapter: Practical.

Note on axis scale: The axes of the images presented in this chapter may appear difficult to interpret and readjusting the axes scale would defy the purpose of working with a consistent underlying coordinate system. This is because the grid coordinates are in OSGB36, the national framework for the UK. However, to obtain an estimate of the distances in each image, either a scale has been added for the necessary visual guidance or the radius of the waypoint is a clear indication of the image's dimension.

4.1 Experimental Tools

The following items were used:

- 0.18 m/pixel resolution aerial image of the Holywell car park at Loughborough University, orthorectified into Ordnance Survey coordinates (British National Grid)
- Two Leica System 500 receivers for precise differential point positioning using static data post-processing (horizontal accuracy 5 mm + 1 ppm, vertical accuracy 10 mm + 1 ppm)
- 3) Garmin 18 5Hz GPS unit
- Erdas Imagine 9.0 by Leica Geosystems
- 5) Freely-available GPS planning software (Trimble)

4.2 Aerial Image and Waypoint Accuracy

In order to show the nature of disparities between a georeferenced aerial image and waypoints, two tests were performed. In the first one, the Leica System was used to collect 54 points using a survey style 'stop-and-go' approach in an attempt to measure points covering the majority of the car park. These points are superimposed on the aerial image (Fig. 4.2a).



Fig. 4.2a: 54 accurately surveyed points using static post data processing

Clearly recognisable and identifiable landmarks on the image (marked as waypoints) were chosen as points to be surveyed by the high precision GPS on their corresponding points in the car park. It can be clearly seen in Fig. 4.2b that many of the waypoints selected do not match their corresponding surveyed points exactly. It was determined that, for the clearly recognisable points (37 of the 54), the surveyed points had an average 0.37 m NE shift from the user selected waypoints (varying from 0.087 m to 0.732 m) (see Fig. 4.2b (a, b) for a waypoint comparison). On the other hand, for the entire data set (54/54), an average 0.446 m NE shift from the user selected waypoints was obtained (varying from 0.087 m to 2.085 m). Such differences can be accounted by the presence of *bias error* and *variability*. The bias error arises from small inaccuracies in the measurement process; most significant being the slight variation in the parameter settings between the image that is used to establish photo-control points and the parameter settings in the RTK-GPS receiver that are used to measure ground check points. There is also a small and systematic height bias in the extracted DEM, which causes a systematic shift in the position of the pixels comprising the orthorectified image. The *variability* usually relates to natural human induced variation;

waypoints selected from an image by one person may differ from a set collected by another. This is represented by the range, or standard deviation.



Fig. 4.2b. Images a) and b) show the discrepancy between the user-selected point (dark-coloured) and the surveyed point (light-coloured)

Given the variation of the shift throughout the image, it is evident that it is not entirely possible to exactly match an image waypoint to the actual location in the car park. Therefore it is important to define a proximity error around each waypoint. This proximity error, however, is left up to the user to define since it should be based on the image resolution, the image positional inconsistencies due to orthorectification, and human error concerning waypoint selection. It is possible to recalibrate the image to the standard needed; however, this would be a daunting task for the average user, and might be beyond the accuracy needed. The overall shift present in this image is in the NE direction. If the actual position of the user selected waypoint is desired the underlying positional data would need to be shifted. Fig. 4.2c below shows a graphical representation of this concept.



Fig. 4.2c. Shows that the user selected point from the orthorectified image would require a positional shift in order to match its true location in the global frame, or create a proximity error circle to encompass the actual point.

The results therefore show that a shift is present in the orthorectified image. Given that survey-grade equipment is not necessarily available to the average user, a point with a proximity error (circle) could encompass the true ground position of the actual waypoint.

This leads to the next set of experimental results that demonstrate the importance of correcting the GPS receiver's positional output to improve the spatial match between the GPS data and the orthorectified image.

4.3 GPS

WGS84 is the default coordinate system adopted by the GPS receiver. Any other coordinate system selected would be based on a mathematical transformation from the default – which as previously stated yields erroneous results (see *Georeferencing*, *Chapter* 2). Because the GPS showed positional variation for a single spot from one day to the next, irrespective of the coordinate system chosen, it was determined that adopting a *mathematical spatial shift* would inevitably provide significantly improved positional accuracy. This would overcome some of these computational errors obtained due to the receiver's internal Molodensky coordinate system transformation [DePriest, D. 2003]. This would provide 'corrected' (or tuned) positional data, suitable for a certain time period and geographic location. The *mathematical spatial shift* is explained in the proceeding sections, and it is termed *GPS positional correction*.

4.3.1 GPS planning software

In order to ensure the most optimal positional precision freely available GPS planning software was used, known as the Trimble Planning Software [Trimble 2007]. Using this software, the user can define the location of the test, select the satellites of interest, obtain a sky plot for a visual representation of the satellite trajectory over the

a) Satellite visibility: an overview of the number of satellites visible during the time of testing. See Fig. 4.3.1a for a sample output from the program.



Fig. 4.3.1.a: Satellite visibility over a 12 hour period (10 am to 10 pm) at the Holywell car park. Optimal visibility (>12) shown for approximately 1.5 hours

b) Position Dilution of Precision (PDOP): this provides an estimate of the satellite configuration relative to each other. The lower (typically < 2) the Dilution of Precision (DOP), the better the constellation, and therefore the better the positional accuracy. Please refer to Fig. 4.3.1.b for a sample output showing ideal working conditions.



DOP Position

Fig. 4.3.1.b: Position Dilution of Precision over a 12 hour period (10 am to 10 pm) at the Holywell car park. Optimal precision shown for approximately 1.5 hours around the 12 pm margin
4.3.2 GPS positional correction

As previously mentioned, the tests should be conducted at a time when the satellite visibility and PDOP are at an optimum in order to ensure the 'best' results. Given that the GPS positional data varies for a single point from one day to the next, as will soon become clear, setting working conditions based on the Trimble Planning Software creates a form of experimental consistency. Following some tests, this has led to the introduction of the concept of GPS positional correction.

This idea was realised at Beacon Hill, Loughborough, on the Ordnance Survey GPS network pillar (see Fig. 4.3.2).



Fig. 4.3.2: Beacon Hill Ordnance Survey pillar

The test area is ideal for GPS users. The area has: a) clear sky visibility without any signal obstruction (loss), b) no overhead electrical cables, and therefore lacks electrical noise, and c) is the highest point in the area, and therefore multi-path (collision of signals) is ruled out.

4.3.2.1 Static Results

Given the above conditions, a test that would summarise the single point repeatability of the GPS was presented. The tests were carried out for four days (12/10/2007 and from the

16/10/2007-18/10/2007). The Garmin GPS was set to the user-defined² setting which showed greater positional proximity to the actual point than the internal OSGB36 coordinate system. Following this, the acquired points where converted using the OSGB36-UTM (OSGB36-Universal Transverse Mercator) Projection and Transformation Calculations spreadsheet from the Ordnance Survey website [OS 2007]. The equations were reinstated in a custom-made VB application for converting batch data. The results can be seen in Fig. 4.3.2.1a.



Single Point Comparison

Fig. 4.3.2.1a: Single point comparison at Beacon Hill's pillar on various days

It can be clearly seen from the above figure that the single point data varies from one day to the next. On the other hand, each data set appears to have its own cluster and with its mean value at a certain distance away from the actual point. The clusters show positional consistency around its own respective average during the data collection process. To clarify

² Since it was determined that the OSGB36 coordinate system was spatially not as close to the position being measured as anticipated, parameters local to the test area were obtained to improve the overall spatial position. These therefore reduce the amount of mathematical compensation needed. The constants used for the 'User Defined Settings' were: inverse flattening factor (Df): 299.3249646; the semi-major axis, equatorial radius (Da): 6377563.396; positional shift along x axis (dx): 371; positional shift along y axis (dy): –112; and the positional shift along z axis (dz): 434. This is based on the Airy 1830 ellipsoid.

the presence of some of the trails shown in Fig 4.3.2.1a, the data from 18/10/2007 is used as an example. Fig 4.3.2.1b, shows the data cluster for 18/10/2007, with the red circle covering the time needed for the GPS positional data to stabilise as the number of satellites in view and PDOP improve. The black circle, on the other hand, shows the reduction in positional accuracy as the number of satellites and PDOP transition into a reduced accuracy configuration. To demonstrate this effect clearly, Fig 4.3.2.1c shows the absolute deviation of the positional data to the actual (being at zero) over the period the data was collected.



Fig. 4.3.2.1b: Data collected on 18/10/2007



Fig. 4.3.2.1c: Deviation distance of data from actual point over the predefined test period

Given this variation, using the GPS on a mobile platform would yield different results from day-to-day, and hence the controller's performance cannot be properly judged with such significant variations. This led to the principal of correcting (i.e. 'tuning') the GPS positional output for the time of the tests.

The GPS positional correction was conducted using the following method: one point in a relatively open area was precisely surveyed. The Garmin GPS was then placed on the same location, at a height equivalent to the robot's GPS height of 1.5 m, to determine an average value over a proposed 15 minute sample time. The data was then converted to the British National Grid Eastings/Northings, and compared to its corresponding surveyed point. The positional shift was then used to compensate for the positional output from the GPS for the forthcoming test. For each test, the static data collection was repeated and the shift accommodated for.

A sample of such a result before and after positional correction that was taken at the test site can be seen in Fig. 4.3.2.1d. The result yielded an average shift of 1.045 m in the Easting and 1.95 m in the Northing direction.



Pre and Post GPS Data Correction

Fig. 4.3.2.1d: Single point comparison at test site showing positional data before and after correction. (4/11/2007 at robot's test site)

This leads to the question of how long this positional correction will last for. Furthermore, would any positional shift *over time* (after GPS correction) have a more dramatic effect on static testing than when in motion? The following section will test the GPS positional stability by the number of waypoints it would have reached when in motion.

4.3.2.2 Traversal Results (In-motion)

To conduct the in-motion experiments in this section, The GPS unit was attached to a trolley (see Fig. 4.3.2.2a), and was guided around a designated marked line *in the road's centre* of a predefined area. The first test was carried out along the perimeter of a sectional area of the car park and the remainder on a marked line crossing a series of waypoints. This was done with the trolley, rather than the robot, to ensure that it was carefully guided on the designated marked lines. Therefore, any output in the GPS position would not be due to the effect of the robot's controller but rather due to the GPS positional variation. Careful measures were taken to ensure the stability and proper tracing of the marked lines.



Fig. 4.3.2.2a: The Garmin GPS receiver mounted to the trolley for the traversal tests

In order to visualise the effect of the GPS positional correction, a test conducted on the car park premises shows a sectional view of the result of the path data before and after this GPS positional correction approach (see Fig. 4.3.2.2b). The test was conducted for 30 minutes for a total travelled distance of 1.3 km (each turn 420.8 m). The speed was

computed at approximately 0.74 m/s. The results clearly show the improvement in the positional data due to this mathematical compensation.



Fig. 4.3.2.2b: The dark lines indicate the GPS results prior to positional correction, and the light lines indicate the post positional correction. Test was conducted on 25/10/2007.

In the above figure, the light lines fall within the road's centre, showing the positive effect of the correction. The path through which the trolley was driven was marked approximately to fall in the road's centre. The purpose of the example in Fig. 4.3.2.2b is to demonstrate the effect of the GPS correction method qualitatively. The actual path was not surveyed using the RTK-GPS.

However, in order to obtain a quantitative measure of the GPS positional accuracy after correction, a test utilising the number of waypoints hit is set up. The waypoint radius is used to estimate the positional accuracy of the receiver, for this application. For this a series of 19 ground-surveyed waypoints (not image selected), as seen in Fig. 4.3.2.2c, were created and the GPS-mounted trolley was driven through them for 10 runs (0.84 km) on the first day and 17 runs (~1.43 km) on the second and third days. The spacing between each waypoint (along the line) is approximately 4 m.



Fig. 4.3.2.2c: 19 waypoints used for testing GPS positional correction approach, repeatability and accuracy.

An open space area was used to ensure an unobstructed sky view.

The static positional data collection before and after the three days of testing is presented, in order to reveal the magnitude of this shift from start to finish No GPS positional correction is added to the results, since the difference between the averages (pre- and post-testing) would be the same. The static data collection lasted on average for 15 minutes for each static collection stage (see Table 4.3.2.2).

Date	Average Easting (m), pre-testing	Average Easting (m), post-testing	Average Northing (m), pre- testing	Average Northing (m), post- testing	Easting shift (m)	Northing shift (m)	Duration of tests (min)
29/10/2007	450947.1195	450945.9346	318020.5973	318019.941	1.1849	0.6563	60
30/10/2007	450946.2323	450946.806	318020.0696	318021.4076	-0.5737	-1.338	37
4/11/2007	450946.4916	450946.563	318020.8	318021.1	-0.0714	-0.3	55

Table 4.3.2.2: Comparison showing the variation in the shift between the average easting and northing data pre- and post-testing.

The results clearly indicate that a positional shift between the average data before and after the testing is inevitable and accounting for such a shift would be necessary. This however, largely depends on the duration of the testing. As previously stated, the effect of such a time-based positional shift is best analysed within the context of this research, by determining how this will affect the number of waypoints reached/crossed (hit), shown in Fig. 4.3.2.2c, of the GPS data. Present GPS positioning data shows improved positional accuracy for mobile robot navigation compared to results prior to the deployment of the geostationary satellites (EGNOS/WAAS) [Panzieri, S. *et al*, 2002].

The results in this section are unlike the results in the experimental section, where the waypoints hit correspond to the centre of the robot. These results are presented in the form of a clustered column. For more details please refer to Appendix C.

On 29/10/2007

- 10 runs (Fig. 4.3.2.2d)
- Average number of satellites was 8.9 (~ 9)
- Average PDOP of 1.5



Fig. 4.3.2.2d: The percentage of waypoints hit for 10 runs before and after positional correction. - before correction, - after correction.

On 30/10/2007

- 17 runs (Fig.4.3.2.2e)
- Average number of satellites was 9.9 (~ 10)
- Average PDOP of 1.5



Fig. 4.3.2.2e: The percentage of waypoints hit for 17 runs before and after positional correction. - before correction, - after correction.

On 4/11/2007

- 17 runs (Fig.4.3.2.2f)
- Average number of satellites was 10
- Average PDOP of 1.3



Fig. 4.3.2.2f: The percentage of waypoints hit for 17 runs before and after positional correction. - before correction, - after correction

The percentage of waypoints hit may vary from one day to the other.

4.4 Chapter Summary

- a) Due to *bias error* and *variability* present in the selection of waypoints from orthorectified images, it is not possible to match an image point exactly to its true ground location. This positional shift can vary from one image to another.
- b) Working with a GPS planning software in the proposed optimal times provides a form of experimental consistency, especially when working with a low-cost sensor.
- c) The PDOP and number of satellites are two main factors that can significantly affect the positional accuracy.
- d) The GPS static data collection revealed positional variation from one day to the next for the same point.
- e) The proposed GPS positional correction approach has shown that improvements of up to 48% was achieved, for a 0.5 m radius on a certain day, guiding a GPS mounted trolley through a predefined chalk line.
- f) From the results we can conclude that the differential GPS positional accuracy (WAAS) is further improved by using positional correction method.

Chapter 5: Spider and Controller Modelling

This chapter presents the models used for the Matlab simulation of the Spider and the GPS, along with the proposed controller to be used in the experimental tests, which are presented in *Chapter 6: Simulated and Experimental Results*.

This chapter is broken down into the following sections:

- a) The kinematic model derivation, which will present the assumptions made for modelling the Spider, the robot posture and the wheel kinematic constraints used to derive the final model.
- b) The GPS error model, which is used to mimic the effect of the GPS proximity error on the functioning of the proposed control system. This model was derived from experimental results obtained from Chapter 4.
- c) The *waypoint reaching controller* that includes the novel:
 - i. *heading control strategy*, which determines the heading direction of the robot based on the angle between the wheels' forward or reverse translational direction and the goal.
 - ii. *circular stages of closeness strategy*, which is used to determine the topological relationship between the waypoint and the GPS receiver, in order to determine both the appropriate linear velocity and the refined instructions for loading the upcoming waypoint.
- d) The novel two-stage *Fuzzy logic control system* that incorporates the proposed GPS Accuracy Decision Factor (GADF), in addition to the GPS receiver/waypoint distance and the heading angle, to determine the suitable linear and angular velocities, respectively.

Chapter pre-requisite: 4.2 Aerial Image and Waypoint Accuracy, 2.1.1 Synchronous Drive Robot Research Nature of Chapter: Theoretical.

93

5.1 Kinematic Model

5.1.1 Assumptions

Since the research being investigated is very interrelated in nature, several assumptions need to be made in order to validate and incorporate the previous experimental results. The assumptions are that the robot has:

- ideal synchronous wheel rotation
- a symmetric wheel configuration (square wheel configuration)
- homogeneous wheel radii for all wheels
- no lateral or longitudinal wheel slip
- no wheel misalignments
- moves along a 2D (horizontal) plane.

This simplifies the kinematic model to the basic constraints acting on the Spider, which will help in the validation of the accuracy of the waypoint navigation model.

5.1.2 The Model

This section of the chapter is dedicated to providing the reader with a detailed derivation of the kinematic model of a four-wheel synchronous drive mechanism.

As previously mentioned in *Chapter 2* (2.1.3 *Synchronous Research*), most synchronous mobile platforms have been used for indoor applications, where the robots' environments are relatively structured and have a 2D resemblance.

The general kinematic model for a synchronous drive is represented by the following equations a, b and c:

$$x(t) = x(t_0) + \int_{t_0}^{t_n} v(t) \cdot \cos(\theta) dt$$
 (a)

$$y(t) = y(t_0) + \int_{t_0}^{t_n} v(t) \cdot \sin(\theta) dt$$
 (b)

$$\theta(t) = \theta(t_0) + \int_{t_0}^{t_n} \dot{\theta}(t) dt$$
 (c)

To obtain a model which is more convenient to work with in Matlab's Simulink environment, a comprehensive schematic (Fig. 5.1.2) is used for the derivation process. For practical tests currently performed with the Spider please refer to *Chapter 3*. For the description of the variables used, please refer to the *List of Variables*.



Fig. 5.1.2: The Spider. Kinematic model of the four-wheeled synchronous drive robot

The main feature of a robot with a synchronous drive mechanism is that all the wheels can simultaneously rotate 360° continuously and unhindered at the same angular velocity ($\dot{\theta}$), translational velocity (v) and in the same direction ($\pm \theta$) (Fig. 5.1.2). For that reason the instantaneous centre of curvature (rotation) (ICC or ICR) is at infinity. The robot's frame

 (X_{Spider}, Y_{Spider}) will remain constant by an angle (α) to the global reference frame (X_{Global}, Y_{Global}) , unless wheel slipping or other unforeseen external dynamic factors occur.

5.1.3 Robot Posture

The robot posture (ξ_{Global}) can be defined by the following vector representation:

$$\xi_{Global} = \begin{bmatrix} x & y & \alpha \end{bmatrix}^{T}$$
(1)

The position of the point (P) is represented by (x, y), and α represents the orientation angle of the robot frame $\{X_{Spider}, Y_{Spider}\}$ relative to the global frame $\{X_{Global}, Y_{Global}\}$. The orientation angle α is measured from the X_{Global} to the X_{Spider} .

Since the global reference frame and the robot frame are not aligned, it is necessary to map the motion of the global frame to that of the robot. To achieve this, an *orthogonal rotation matrix* ($R(\alpha)$) is needed:

$$R(\alpha) = \begin{bmatrix} \cos\alpha & \sin\alpha & 0\\ -\sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1 \end{bmatrix}$$
(2)

The calculation is denoted by:

$$\xi_{Spider} = R(\alpha)\xi_{Global} \tag{3}$$

5.1.4 Wheel Kinematic Constraints

The next stage is to calculate the wheels' kinematic constraints. Since this is a synchronous mechanism, the calculation of one wheel is sufficient. For this, both constraints orthogonal to and along the wheel plane need to be determined. Please refer to Fig. 5.1.4.



Fig. 5.1.4: Wheel's kinematic constraints - with the orthogonal and along wheel frames

In order to compute the correct constraints, it is vital to determine the type of wheel being used. The Spider's wheel belongs to the class of *steered standard wheels*. The resolved equations are:

Along the wheel plane:

$$\left[\cos(\theta_i) \quad \sin(\theta_i) \quad d_i \sin(\theta_i) - b_i \cos(\theta_i)\right] R(\alpha) \dot{\xi}_{Global} - r\dot{\phi}_i = 0 \tag{4}$$

Orthogonal to the wheel plane:

$$\left[-\sin(\theta_i) \quad \cos(\theta_i) \quad d_i \cos(\theta_i) + b_i \sin(\theta_i)\right] R(\alpha) \dot{\xi}_{Global} = 0$$
(5)

Where $\dot{\xi}_{Global} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{\alpha} \end{bmatrix}^T$ is the robot's posture velocity vector, θ_i is the steering angle at a certain instant in time, and d_i, b_i are the positions of the wheel's with respect to point *P* along the robot's frame (refer to Fig. 5.1.2). The subscripts stand for:

 b_r : Back right b_l : Back left f_r : Front right f_l : Front left

Given that the Spider has a symmetric four wheel configuration (d = b), then:

$$d_{fl} = d_{fr} = b_{bl} = b_{fl} = d$$

$$d_{br} = d_{bl} = b_{br} = b_{fr} = -d$$
(6)

Therefore, equations 4 and 5 can be modified with the dimensions presented in 6 to obtain the full kinematic wheel constraints acting on the wheel frame, within the assumptions that the wheels are the same size and the same radius r, and that $\dot{\theta}_i = \theta \,\forall i$, $\dot{\phi}_i = \phi \,\forall i$.

Hence:

Along the wheel plane:

$$b_{r}: \quad \left[\cos(\theta_{br}) \quad \sin(\theta_{br}) - d\sin(\theta_{br}) + d\cos(\theta_{br})\right] R(\alpha) \dot{\xi}_{Global} - r\dot{\phi}_{br} = 0$$

$$b_{l}: \quad \left[\cos(\theta_{bl}) \quad \sin(\theta_{bl}) - d\sin(\theta_{bl}) - d\cos(\theta_{bl})\right] R(\alpha) \dot{\xi}_{Global} - r\dot{\phi}_{bl} = 0$$

$$f_{r}: \quad \left[\cos(\theta_{fr}) \quad \sin(\theta_{fr}) - d\sin(\theta_{fr}) + d\cos(\theta_{fr})\right] R(\alpha) \dot{\xi}_{Global} - r\dot{\phi}_{fr} = 0$$

$$f_{l}: \quad \left[\cos(\theta_{fl}) \quad \sin(\theta_{fl}) - d\sin(\theta_{fl}) - d\cos(\theta_{fl})\right] R(\alpha) \dot{\xi}_{Global} - r\dot{\phi}_{fl} = 0$$
(7)

Orthogonal to wheel plane:

$$b_r: [-\sin(\theta_{br}) \cos(\theta_{br}) - d\cos(\theta_{br}) - d\sin(\theta_{br})]R(\alpha)\dot{\xi}_{Global} = 0$$

$$b_l: [-\sin(\theta_{bl}) \cos(\theta_{bl}) - d\cos(\theta_{bl}) + d\sin(\theta_{bl})]R(\alpha)\dot{\xi}_{Global} = 0$$
(8)

$$f_r: \begin{bmatrix} -\sin(\theta_{fr}) & \cos(\theta_{fr}) & d\cos(\theta_{fr}) - d\sin(\theta_{fr}) \end{bmatrix} R(\alpha) \dot{\xi}_{Global} = 0$$

$$f_l: \begin{bmatrix} -\sin(\theta_{fl}) & \cos(\theta_{fl}) & d\cos(\theta_{fl}) + d\sin(\theta_{fl}) \end{bmatrix} R(\alpha) \dot{\xi}_{Global} = 0$$

5.1.5 Resulting Model

Following this, the kinematic constraints need to be expressed in the Matrix form:

$$A(q)\dot{q} = 0 \tag{9}$$

 $\begin{bmatrix} \dot{x} \end{bmatrix}$

This yields:

$$\begin{bmatrix} -\sin(\theta + \alpha) & \cos(\theta + \alpha) & -d(\sin(\theta) + \cos(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin(\theta + \alpha) & \cos(\theta + \alpha) & -d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin(\theta + \alpha) & \cos(\theta + \alpha) & d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin(\theta + \alpha) & \cos(\theta + \alpha) & d(\sin(\theta) + \cos(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & -r & 0 & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\sin(\theta) + \cos(\theta)) & 0 & 0 & 0 & 0 & 0 & -r & 0 & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\sin(\theta) + \cos(\theta)) & 0 & 0 & 0 & 0 & 0 & -r & 0 & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\sin(\theta) + \cos(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & -r & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & -r & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & -r & 0 \\ \cos(\theta + \alpha) & \sin(\theta + \alpha) & -d(\cos(\theta) - \sin(\theta)) & 0 & 0 & 0 & 0 & 0 & 0 & -r & 0 \\ \end{bmatrix} \begin{bmatrix} \dot{y} \\ \dot{x} \\ \dot{\theta}_{br} \\ \dot{\theta}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \end{bmatrix}$$

In order to obtain the *state space representation* of the robot, it is important to determine the *null space* of A(q) for $v = r.\dot{\phi}$, where $\dot{\phi}$ is the rotational velocity of the wheels around their axle, with a radius of r. Please refer to Fig. 5.1.4. Hence the representation in the form of $\dot{q} = S_q u$:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\theta}_{br} \\ \dot{\theta}_{br} \\ \dot{\theta}_{br} \\ \dot{\theta}_{fr} \\ \dot{\theta}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \end{bmatrix} = \begin{bmatrix} \cos(\theta + \alpha) & 0 \\ \sin(\theta + \alpha) & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1/r & 0 \\ 1/r & 0 \\ 1/r & 0 \end{bmatrix}$$
(11)

Since $\theta_i = \theta \,\,\forall i$ and $\dot{\phi}_i = \phi \,\,\forall i$, equation 11 can be reduced to the following:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\theta + \alpha) & 0 \\ \sin(\theta + \alpha) & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1/r & 0 \end{bmatrix} \begin{bmatrix} v \\ \dot{\theta} \end{bmatrix}$$
(12)

Given that $v = r.\dot{\phi}$ equation 12 can be rewritten as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} r \cdot \cos(\theta + \alpha) & 0 \\ r \cdot \sin(\theta + \alpha) & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix}$$
(13)

Referring back to equation (12) it can be seen that five factors are needed to determine the robot's velocity components in the x-y plane (forward kinematics): the steering angle θ , the angle of the robot frame (the pose) with respect to the global frame α , the linear velocity of

the wheel v, the angular steering velocity of the wheel $\dot{\theta}$, and finally the radius of the wheel, r (with r and α being constants and the remainder variables).

Since one of the assumptions states that there is no lateral or longitudinal wheel slip, then the robot's orientation will never change during the course of its motion, which is represented by the rate of change $\dot{\alpha}$ being equivalent to zero. Therefore, geometric relations were used to derive the equations.

In theory (simulation), it is possible to achieve a specific pose if desired (inverse kinematics), since the wheels' angular steering velocity ($\dot{\theta}$) and translation velocity (v) are controlled independently (decoupled). However, the robot has a non-holonomic configuration, which means that you cannot achieve a specific orientation because α cannot be controlled. Therefore, it is possible to go back to the same position but not necessarily in the same starting configuration. In control theory this means that the robot's posture ξ can only be partially stabilised. Moreover, it is not input-output static state feedback linearisable using the method presented by d'Andréa-Novel *et al* [1995], because you cannot use the output equations to linearise the system. However, it is possible to control two variables on the plane, as imposed by the Brockett necessary condition, because it states that if you have two inputs you can only control two outputs [Brockett, R.W. 1983]. Since the plan is to simulate the robot's motion in Matlab, the format of equation 12 will be used.

5.2 GPS Error Model

After the GPS positional correction, a fluctuating shift can still be experienced in the output. This shift is minor compared to the positional correction shift (*Chapter 4*); nevertheless, the presence of this fluctuating shift must be taken into account in the proposed control system. Please refer to Fig. 5.2a for a visual representation of this shift, where it can be seen that the lighter lines are not smooth, indicating a normalised fluctuation in the positional fix of the GPS receiver. Additionally, it can be seen that in two runs the light lines fell within close proximity of one another, but in a third turn the position shifted. Results vary from one test to the other. Both the fluctuation and the shift could have an oscillating effect on the control system output if not taken into account.



Fig. 5.2a: Dark lines represents the positional data before correction (*Chapter 4*), and the light ones after the correction for the GPS mounted trolley that was driven along a pre defined series of waypoints. A general fluctuation in the positional data can still be seen. The radius of the waypoint is 0.7 m.

In order to ensure optimal functioning of the proposed fuzzy controller, it is necessary to introduce a comparable GPS positional error into the simulation.

This proposed GPS error model, although simplistic in nature, will provide an adequate representation of the actual output of the GPS receiver. See Fig. 5.2b for a schematic of the GPS error model.



Fig. 5.2b: The GPS position after the positional error correction (*Chapter 4*) is presented by the small circle. This is arranged such that it would always fall within the boundaries of the GPS error circle.

The shift:

The variable R_{GPS} is derived from the initial positional value of the GPS output at the start of an experiment. Once the robot is launched from a surveyed waypoint, the initial value of the GPS output is used as follows:

$$dq = x_{GPS} - x_{start}$$

$$dm = y_{GPS} - y_{start}$$
(14)

where dq is the error along the y direction of the Spider's axis, and dm is the error along the x direction of the Spider's axis. dq and dm are therefore chosen to represent a circle of radius R_{GPS} , which will be referred to as the GPS error circle:

$$dq^2 + dm^2 = R_{GPS}^2 \tag{14b}$$

(GPS error circle, as seen in Fig. 5.2b).

This leads to the model derivation by means of the transformation matrix:

$$\begin{bmatrix} x'\\ y' \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha\\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} x\\ y \end{bmatrix}$$

$$y_{GPS} = y' + dq$$

$$x_{GPS} = x' + dm$$
(15)

The fluctuation:

Since the GPS positional data shows fluctuations in its path (as also shown by Witte, M. and Wilson, A. [2005]), a similar effect of this behaviour can be reproduced by means of a random number generator that produces a continuous uniform distribution in Matlab.

Fluctuation of data along each axis:

Along the y axis:
$$(dq/80)*unifrnd(-1,1)$$

Along the x axis: $(dm/80)*unifrnd(-1,1)$

$$y_{GPS} = y' - dq + (dq/80) * unifrnd(-1,1)$$

$$x_{GPS} = x' - dm + (dm/80) * unifrnd(-1,1)$$
(16)

dm/80 and dq/80 were based on generating a random sequence of numbers around the 80^{th} of the distance of the GPS from the robot's centre. The mathematical relationship and the chosen factor are assumptions based on experience working with the GPS and the simulation, in which visual comparisons between both outputs were used. Fig 5.2c shows a close resemblance between fluctuations obtained from the GPS receiver and those obtained from the simulated GPS output, using a factor of 80. The fluctuations do not depend on the number of satellites in view or the PDOP. It is important not to forget that this is a basic model to introduce a GPS-like error into the simulation to mimic the actual GPS output as

closely as possible. Fig. 5.2.d shows the position of the robot both before and after the GPS induced error.



Fig. 5.2c: 1) the white line shows the fluctuation in the GPS positional data. 2) the white line shows the simulated fluctuation in the GPS error model.



Fig. 5.2.d: 1) The black lines indicate the centre of the robot with no GPS induced error. 2) In this image, the white lines represent the position of the GPS induced error and the black lines represent the position of the robot. The controller adjusts according to the GPS signal and therefore the robot can be seen to be off-track. The behaviour of the GPS shows similar traits to those seen in Fig. 5.2a, and the robot is also expected to exhibit similar behaviour. Both fluctuation and a shift are present in this GPS induced error.

5.3 Waypoint Reaching Controller

The novel combination control system proposed for the Spider is broken down into two parts: The *heading control strategy* discusses the geometric model for the necessary wheels' orientation, to ensure that the correct heading is used along with the appropriate control system for adjusting the angular velocity within the presence of the GPS error. The *circular stages of closeness* details the model used for controlling the speed followed by the strategy used for deciding whether or not the waypoint has been reached, in order to load the following waypoint. Furthermore, the justification for the choice of fuzzy controller will be discussed in each section. The overall control system layout can be seen in Fig. 5.3, with the details discussed in the proceeding sections.



Fig.5.3: The control system layout

5.3.1 The heading control strategy

The heading controller strategy presented is a modified version of the goal reaching approach proposed by Gonzalez, V. *et al* [2004] combined with a fuzzy control strategy inspired by Vaneck, T.W. [1997], who worked on the development of a fuzzy waypoint following controller. This combined approach takes advantage of the robust manoeuvrability of the synchronous drive robot by minimising the wheels' angular rotation to adjust for its heading, and furthermore avoid robot steering oscillation that can be caused by the GPS inaccuracy, by means of the fuzzy controller:



Fig. 5.3.1a: The waypoint reaching model based on work by Gonzalez, V. et al [2004], in which the full functionality of the synchronous drive's manoeuvrability is utilised.

This is achieved by ensuring that the acute angle between either side of the wheel to the goal becomes the choice for direction of rotation.

$$P_{GPS} = (x_{GPS}, y_{GPS}) \tag{17}$$

$$\theta_g = \tan^{-1} \left(\frac{y_w - y_{GPS}}{x_w - x_{GPS}} \right) \quad 0 \le \theta_g \le 2\pi$$

 $\theta_{ef} = \theta_f - \theta_g$ (error angle to the goal – forward direction)

and

 $\theta_{eb} = \theta_b - \theta_g$ (error angle to the goal- reverse direction)

where $0 \le \theta_f \le 2\pi$ and $0 \le \theta_b \le 2\pi$

If $abs(\theta_{ef}) \leq abs(\theta_{eb})$ then rotate by θ_{ef} in the v_f direction.

If $abs(\theta_{eb}) \le abs(\theta_{ef})$ then rotate by θ_{eb} in the v_r direction.

The result is passed through a fuzzy controller which is presented in Section 5.3.3 Fuzzy Controller. In Fig 5.3.1 b/c, a sample waypoint reaching approach can be seen before and after the proposed heading control strategy. This approach provides an optimised path, since a shorter distance to the waypoint is taken, than the conventional unidirectional motion only. This method (heading control strategy) therefore exploits the full motion range of the synchronous drive robot.



Fig.5.3.1: b) Shows that the Spider is travelling only in the forward translational velocity when reaching a waypoint, therefore not taking full advantage of the synchronous drive capability; b) shows the Spider after implementing the heading control strategy, where both the forward and reverse translational velocities are taken into account.

5.3.2 Circular Stages of Closeness

As previously mentioned in *Chapter 4*, a waypoint selected from an aerial image does not necessarily match its corresponding point on the ground, due to variability and bias error. These errors vary in magnitude throughout the aerial image, and the future goal is to determine their magnitude in order to create an error map. In this work, the error associated with the selection of the correct waypoint is represented as a circle, because the true position of the waypoint in space could be anywhere in an omnidirectional area around the image-selected point. This waypoint proximity error is represented by *RB*. Depending on the magnitude of the error the radius is adjusted, which leads to the creation of waypoints with various radii. Another important result of *Chapter 4* was to show that the error of the GPS positional output can be adjusted, but only to a certain extent. The presence of this additional error, as presented in the *GPS Error Model*, would also need to be accommodated for. As in the waypoints, this is represented by a circle, where the magnitude of the radius represents the GPS error. Therefore, a new circle is created that is pivoted around the P_{GPS} that contains the Spider, which will be known as the *Spider Error Circle* with a radius of *RA*.

The presence of two circles has called for the need for a model that deals with the interaction of two circles. The work in this section has therefore been motivated by the work represented by Wuersch, M. *et al* [2005] on refining route instructions based on topological stages of closeness of a navigator to the waypoint. Hence, the velocity control and the loading of the next waypoint are based on the stages of closeness of the Spider error circle and the waypoint circle (see Fig. 5.3.2 a).



Fig 5.3.2a: The circular model representations that are used for the waypoint reaching control strategy. The Spider error circle is pivoted at the P_{GPS} with a radius of R_A . The waypoint has a radius of R_B .

Where:

$$d_c = d_p - (R_A + R_B) \tag{18}$$

$$d_p = \sqrt{(x_w - x_{GPS})^2 + (y_w - y_{GPS})^2}$$
(19)

The radius of R_A is selected where 95% of the static data collected from the GPS for the positional correction falls within its boundary. The reason for this choice is because industrial GPS standards specify the positioning error of a receiver in terms of the 95th percentile Garmin Ltd. 2007].

The proposal made by Wuersch, M. et al [2005] is that there are 26 combinations of circular intersections that could define the closeness of the navigator to the waypoint. The

combinations are classified in terms of the sizes of the radii of both the navigator and the waypoint. Because the Spider is only capable of limited velocity configurations, it is not fine-tuned enough to slow down for all stages of closeness; therefore, the more impractical configurations have been eliminated (see Table 5.3.2).

Therefore, the variable d_c will be used as the criteria for fuzzy based velocity control and the variable d_p for the decision based on when the following waypoint should be loaded – (presented in 5.3.3 *Fuzzy Controller*) as seen in the following table:

Stages of Closeness								
Radius size		d_c Meet $d_c = 0$	Stage(s) of closeness					
Case 1 $0 < R_B < \frac{1}{2}R_A$			$d_y < \frac{1}{2}R_A \text{ AND } d_y < R_B$					
Case 2 $R_B = \frac{1}{2}R_A$			$d_y < \frac{1}{2}R_A \text{ AND } d_y < R_B$					
Case 3 $\sum R_A < R_B < R_A$			$< R_A \text{ AND} > \frac{1}{2}R_A d_y < \frac{1}{2}R_A$ AND $d_y > R_B d_y < R_B$					
Case 4 $R_{B} = R_{A}$			d, < RA AND d, < RB					
Case 5 $\frac{1}{2}R_B < R_A < R_B$			$ \begin{array}{c} & & \\ & & \\ & \\ & \\ & \\ \\ & \\ \\ & \\$					
Case 6 $R_A = \frac{1}{2}R_B$	0	0	$d_{y} < R_{A} \text{ and } d_{y} < \frac{1}{2}R_{B}$					
Case 7 0 < R _A < ½ R _B	0	0	$d_{y} < R_{A}$ AND $d_{y} < \frac{1}{2}R_{B}$					

Table 5.3.2. Stages of closeness for loading upcoming waypoints.

As the radii of both waypoint and Spider error circle become closer together, less refined stages of closeness need to be used. For instance in the example where Ra = Rb the overlap 3 is simply the safest option since the circle could be approaching it from anywhere.

The particular cases in the fourth column (Table 5.3.2, Stage(s) of Closeness) were selected because the initiation of waypoint loading should not start until those conditions are met. When this initiation begins, the dp will be monitored; if it begins to increase after a steady decrease, then the next waypoint will be loaded.

Since the GPS values fluctuate, as mentioned in the GPS error model, the circle surrounding it (Spider error circle) will also shift. However, the shift may not be sufficient for the wheels to change their orientation

5.3.3 Fuzzy controller





Fig. 5.3.3a: Fuzzy angular and linear velocity controller using a GPS Accuracy Decision Factor (GADF)

The Fuzzy controller is broken down into two stages. The first stage deals with the fuzzification and defuzzification of the two important GPS indicators for positional accuracy - Position Dilution of Precision (PDOP) and Number of Satellites (See *Chapter 4*) - to output the GPS Accuracy Decision Factor (GADF). The second stage uses this decision

factor in combination with the robot's wheel orientation and distance between the Spider error circle and the waypoint to determine the appropriate angular and linear velocity for the robot. Each stage will be dealt with independently. For an introduction on Fuzzy logic and its applications please refer to Bih, J. [2006].

The benefit of using Matlab for the development of the control system was in the ability to set the fuzzy rules in the simulation environment and use them for the experimental tests. In order to validate the performance of the real robot to the simulated robot, the same fuzzy rules and memberships were used. The same quantitative cutoffs that were used in the fuzzy controller in the real robot were also applied to the simulated environment (see Appendix B).

The justification behind these cutoffs was based on experience operating the GPS receiver and the Spider. The main focus is to demonstrate the possibility of using fuzzy logic along with the proposed system and to show the effects of PDOP and number of satellites on the robot's behaviour, which will be demonstrated in detail in Chapter 6.

A potential step forward for adjusting the fuzzy logic cutoffs could be in the adaptation of a neuro-fuzzy controller. This however is also beyond the intended scope of this research.

The GPS Accuracy Decision Factor (GADF)

The general behaviour of the GPS positional output can be predicted, yet it is difficult to anticipate how and by how much the positional data will change with regards to the change in PDOP and number of satellites. It is known that those two factors do affect the positional output; the lower the PDOP and the higher the number of satellites, the better the positional accuracy of the GPS and therefore the lower the positional error. This resultant GPS accuracy is titled the GPS Accuracy Decision Factor, or GADF – where a high GADF denotes a high positional accuracy. Modelling the behaviour of the positional output is beyond the scope of this research; however, accommodating for this behaviour can be achieved by means of a Fuzzy controller. The triangular memberships and the fuzzy rules are presented in Fig. 5.3.3 b. For a more intuitive modelling and interpretation of the GADF fuzzy rules, rather than saying that a low PDOP value denotes high positional

accuracy, *PDOP* triangular memberships now refer to *PDOP accuracy* where in this case a *high PDOP accuracy* would be a positive contributor to the GADF i.e. H_{PDOP} refers to high *PDOP accuracy*, but in real terms means a low PDOP value.

Stage 1 Fuzzy Controller Parameters

Inputs

PDOP: Range: 0.2 to 5. During the course of this research, the GPS receiver only very rarely showed fewer than 7 satellites in view – and these rare occasions only occurred when the receiver was in the vicinity of a building, causing signal blockage. Therefore, the likelihood of having a poor satellite constellation with respect to each other is decreased with a higher number of satellites, because having a higher number of satellites is always the more favourable option, which will become clearer in the discussion chapter. As a result, it is not possible to have a reduction in the positional accuracy that would lead to a PDOP beyond 5. Experimentally, a maximum PDOP of 3.2 was previously noted. Hence, a PDOP of 5 was set as the upper limit for the worst-case scenario, where a building would reduce the number of satellites dramatically, leading to the higher possibility of a high PDOP. In addition, the combination of 10 or 11 satellites was never shown to reduce the PDOP beyond 0.9, and therefore 0.2 had been set as the lower margin to accommodate for possible (rare) occasions when the PDOP would be lower than 0.9 (but greater than 0).

Tuning the PDOP to within a smaller range would require a more detailed study into the effect of such variations, when subject to different environments. This could potentially lead to a reduction or expansion of the triangular membership range.

Number of Satellites: Range: 0 to 12. The membership functions of the number of satellites were set between 0 and 12. Even though the number of satellites would not be expected to be lower than 7 when in an open, unobstructed space, it is possible for the number of satellites to dip below this margin due to the presence of trees, buildings, or other obstructions. Furthermore, during the winter months, the reduction in foliage allows the GPS receiver to have a greater number of satellites in view, compared to the summer

months. Therefore, the medium range from 3 to 7 satellites was created to be flexible enough to accommodate for these variations in obstructions (even within the same test area). For safety precautions, a worst case scenario of zero was set as the minimum for the lower range, to accommodate for the condition if no signal is yet received by the GPS receiver, following a 'cold start' (when the GPS receiver is initially turned on). For the upper range, a view of greater than 12 GPS satellites at one time is not possible, so this was set as the maximum.

Output

GADF: Range: 0 to 1. Given that no such factor had been previously used, the triangular memberships were created in order to ensure that they are evenly spread throughout the range. The *Low GADF* ranges from 0 to 0.4, the *Medium GADF* from 0.1 to 0.9, and the *High GADF* from 0.6 to 1 (see Fig. B11). These ranges can be investigated in any future work, which will require a further study of the effect of the membership functions of the PDOP and the number of satellites on the output of the robot's behaviour.


Fig. 5.3.3b: Fuzzy rules for GADF using a worst case scenario logic system for combining PDOP accuracy to the number of satellites.

The acronyms *L*, *M* and *H* stand for *Low*, *Medium* and *High*, respectively. Since the same nomenclature was used for both the PDOP and the Numbers of Satellites (NOS), the subscripts PDOP and NOS were used to distinguish between them. The resultant output is also presented as *Low*, *Medium* and *High*, with the subscript GADF.

A 'worst case scenario' approach was used to exercise caution in the GADF outputs. For example, if the PDOP accuracy is defined as *high*, but the number of available satellites is only *medium*, then it would be too generous to state that the overall accuracy is high; the best it could be would be *medium-high*. However, to restrict the output choices to only

three (low, medium, and high), it was decided to 'downgrade' to the lower accuracy, or to the average accuracy when possible. This therefore gives a cautious GADF output. A resultant 9 rules (3x3) are yielded.

> If $(PDOPaccuracy \in (L, M, H))$ and $(NOS \in (L, M, H))$ Then $(GADF \in (L, M, H))$

Linear and angular velocities

Because the Spider has fixed velocity and angular velocity command sets, the choice of an appropriate controller is a challenge. Nevertheless, the Fuzzy controller provides inherent benefits that will become more clear below.

1) Since the GPS positional output fluctuates, the defining variables θ_{ef} or θ_{eb} will change accordingly, therefore risking an oscillating output. By using fuzzy rules, these fluctuations can be ignored, leading to smooth waypoint reaching. This method would not be suitable for systems that are not open loop stable, but since the Spider has a robust mechanism this methodology is quite favourable.

2) The use of sophisticated control strategies or algorithms created in ideal simulation environments to validate a concept are difficult to tune [Maalouf, E. *et al* 2005], leading to a subsequent trial-and-error refinement due to the presence of unaccommodated nonlinearities, uncertainties, varying operating conditions, noise and the lack of description of the robot's unstructured environments [Cupertino, F. *et al* 2006].

3) The Spider is designed to mow on rough and sloped terrain, which is a difficult and dangerous environment for a human. This makes it necessary to ensure that a controller is sufficiently robust for these working conditions. Using a controller that is validated (tuned) with human experience, and that would take into account environmental disturbances, would prove to be highly beneficial. See Fig. 5.3.3c for a schematic of the proposed controller.

Stage 2 Fuzzy Logic Parameters

Inputs

 θ_{ef} or θ_{eb} : Range: 0 to 90. The five memberships were created based on the behavior that would be desired as the robot's wheels rotate from 90 degrees down to the *Very Small* margin. When a high GADF is present, and the θ_{ef} or θ_{eb} is large, then it is desired that the angle be reduced as fast as possible in order to ensure a refined control of the small angles. Please refer to Fig. B10.

Since the output from the potentiometer would fluctuate, then so would the error angle. In light of this, the *Very Small* cannot be reduced to less than 5 degrees, and therefore accurate adjustment of the wheel angles would not be possible. Reducing it to less than 5 degrees would inevitably cause the robot to continuously respond to the fluctuating output of the potentiometer and of the GPS receiver, therefore leading to unstable control of the Spider. As for the other triangular memberships, they are defined in order to ensure a smooth rotation of the wheels to the desired angle.

 d_e : Range 0 to 4: The maximum range was set at 4 since the majority of the spacing between the waypoints were at approximately equivalent to 4 m. Even though a few of the other waypoints are spaced further apart, the fuzzy controller accommodates for this, and considers it to be a *Large* distance. In the case where the distance is *Large* in conjunction with a high GADF, it would approach the waypoint at its maximum velocity. The even spacing between the triangular memberships *Medium* and *Small* is to ensure that a smooth velocity transition occurs as the robot gets to within a close proximity of the waypoint. Once within the close proximity (*Very Small*), the velocity is reduced dramatically to ensure a more refined control, if θ_{ef} or θ_{eb} is greater than 5 degrees; otherwise, it will approach the point at a reduced speed. This approach would work well with any future plans that would use localised reactive control. *GADF:* The cutoffs are identical to the output from stage one, as it is the common link between both stages. The combinatorial effect with θ_{ef} or θ_{eb} and d_e can be seen in Fig. 5.3.3c.

Outputs

 θ : The singleton outputs for the angular velocity were set at 0, 0.2, 0.4, 0.6 and 0.8 rads/s. These values were obtained by suspending the robot off the ground and measuring the angular velocities corresponding to each command set from the joystick. Even though these values do not coincide with the true angular velocities when the wheels are in contact with the ground (due to friction and dynamic effects), they are nevertheless linked to the correct angular velocity command. Obtaining an accurate estimate of the angular velocity would prove to be a difficult task due to multiple reasons, such as slack in the chain of the steering mechanism and the state of the battery, as will become clearer in the Discussion chapter.

 ν : The singleton outputs for the linear velocities were set at 0.1, 0.3, 0.5 and 0.7 m/s, which are based on the low velocity command setting of the Spider. The robot has two settings: the low velocity setting which goes up in four steps to around 2.5 km/h (0.7m/s), and the high speed setting that goes up to around 7 km/h (also in four steps). For safety reasons, only the low velocity setting was used. The linear velocities of the Spider throughout the various velocity increments (steps) were not consistent (see Discussion chapter, section 7.2.2); however, these values were nevertheless used in order to compare the simulation and experimental results, since the simulation also uses these values. A linear velocity of zero was never set for the robot, as it was not desired that the wheels rotate without the robot being in motion, as was previously discussed in Chapter 3. Therefore, unless the WAAS/EGNOS signal was lost, the robot would always be in motion, even if only at its lowest speed. Future work could investigate the effect of using this fuzzy logic controller with a high speed setting.



Fig 5.3.3c: This figure shows the second stage of the fuzzy controller, where the (d_c) and $(\theta_{ef} \text{ or } \theta_{eb})$ are used in conjunction with the GADF to control the linear and angular velocities.

It should be noted that, as in the first-stage fuzzy controller, a worst-case scenario approach is used to exercise caution and to minimise errors related to overly-large velocities.

The acronyms VS, S, M, L and VL, stand for Very Small, Small, Medium, Large and Very Large respectively, for both the $(\theta_{ef} \text{ or } \theta_{eb})$ and (d_c) ; where the subscript (A) stands for Angle and (d) for distance. The outputs are presented with the acronyms S, M, F and VF which stand for Slow, Medium, Fast and Very Fast respectively. The subscript $(\dot{\theta})$ is used to represent the angular velocity and (v) for linear velocity. In the fuzzy rules for the angular velocity output, a further option of N is added which stands for No Change. This refers to the fact that if the angle falls within the Very Small category, the angular velocity will be zero.

For both fuzzy rules, only the positive output is presented since the aim is to reduce the number of fuzzy rules. Given that the fuzzy model is symmetric, the negative of the output for both would define either the clockwise or reverse direction for the angular and linear velocities respectively, and therefore the addition of further rules would be unnecessary.

Furthermore, in the second set of membership functions the magnitude of the velocity is dependent on the distance between the circles as seen in Fig. 5.3.2. The magnitude output of the distance is always positive, due to $d_p = \sqrt{(x_w - x_{GPS})^2 + (y_w - y_{GPS})^2}$, and therefore the velocity output will also be positive from the fuzzy controller. The direction of the velocity (i.e. forward or reverse) depends on the condition $abs(\theta_{ef}) \le abs(\theta_{eb})$ as seen in section 5.3.1, which is used to adjust the output sign.

A combination of 5x3 + 5x3 = (30) rules is the resultant.

If
$$(d_c \in (S, M, L, VL))$$
 and $(GADF \in (L, M, H))$ Then $(v \in (S, M, F, VF))$

If $(\theta_{ef} \text{ or } \theta_{eb} \in (S, M, L, VL))$ and $(GADF \in (L, M, H))$ Then $(w \in (N, S, M, F, VF))$

The fuzzy logic controller shows even further advantages for use in this project in addition to the ones previously stated. The choice of singleton outputs is exactly what is needed, since the Spider is limited with angular velocity and linear velocity commands. Furthermore, the presence of a Matlab toolbox for compiling the tuned fuzzy controller in Simulink, is a favourable option, since the file can then be called from a Matlab instance running in Visual Basic. This allows for a speedy deployment and adjustment of parameters.

5.4 Chapter Summary

Several novel contributions are described in this chapter, and to re-state them, are outlined below:

- a) In this chapter, several models were presented. The kinematic and the GPS error model were both developed strictly for the Matlab simulation. The former was needed for the geometric modelling of the Spider, and showed that only three inputs (r,θ,α) are needed to determine the robot's position in the simulation space. The GPS error model, on the other hand, was created in order to mimic a more realistic simulation of the positional error that would accompany a GPS receiver.
- b) The waypoint reaching controller is a novel development for both the simulation and the Spider prototype. This was broken into several sections:
 - The novel heading control strategy, which determines whether the reverse or forward translational direction (heading direction) of the wheel has a more acute angle to the goal in order to economise travel distance and exploit its synchronous drive capability.
 - The novel use of the circular stages of closeness has two functions: The first of which is used for the loading of the upcoming waypoint, depending on the relationship between the two intersecting circles of the GPS receiver and the waypoint. The second, uses the circles as a gauge for controlling the velocity of the Spider as the circles intersect, in order to ensure that the robot approaches the waypoint at an optimal velocity.
- c) The novel fuzzy controller strategy is broken down into two stages. The first of which uses the Position Dilution of Precision (PDOP) and the Number of Satellites (NOS) to determine an accuracy criteria for the GPS receiver. This is named the GADF (GPS Accuracy Decision Factor). In the second stage this output is then used to determine, in conjunction with the distance between the circles (d_c) and the angle between the wheels' heading direction and the goal (θ_{ef} or θ_{eb}), the

appropriate linear (ν) and angular ($\dot{\theta}$) velocities. The fuzzy rules were created on the basis of a worst case scenario.

Chapter 6: Experimental and Simulated Results

In this chapter the combination of the previously obtained results and the proposed models had been implemented, and both the experimental and simulated results are presented in the following order:

- a) Spider Tests A: The results of the fuzzy controller, number of waypoints hit and the sensory data are shown for both the corrected and non-corrected GPS data under both medium and high GADF conditions.
- b) Spider Tests B: The comparison between the use of circular stages of closeness and using a point-to-point approach.
- c) Spider Tests C: The qualitative results of the different cases of circular stages of closeness, achieved through varying the Spider error circle (RA), are shown.

The results of this chapter are discussed in Chapter 7.

Chapter pre-requisite: 4.0 System's Components – Preliminary Test Results, 5.0 Spider and Controller Modelling. Nature of Chapter: Practical and Simulation.

Note: Prior to each test, the robot and wheel orientation were readjusted to the home position to ensure consistency.

6.1 Spider Tests A

In this section of the Spider tests several aspects of the research will be presented. After a series of tests, a selected few have been compiled³. The Spider tests A are broken down into two main parts: Medium GADF and High GADF. It was intended to test the robot at low GADF; however, these conditions (as presently defined) would only have existed under a combined effect of fewer than six satellites and a PDOP greater than two, which did not occur during the given testing periods.

In each section the following will be shown:

- 1) A summary of the test conditions.
- 2) The static data collection used for the GPS correction.
- 3) Spider's positional data (P) and the GPS positional data (P_{GPS}) superimposed onto the orthorectified image.
- The results of the Fuzzy control system's response (See Appendix B for Matlab's fuzzy controller).
- The percentage of waypoints hit, with respect to the robot's centre, before and after correction.
- Simulation results for all of the above tests. (See Appendix B for Matlab's Simulink model).

On the given day of the tests Figs. 6.1a and 6.1b show the variation in the number of satellites and position dilution of precision, respectively, over the course of a 12 hour period starting from 7:00 am.

³ The tests in Chapter 4 demonstrated the repeatability of the GPS data both with and without positional correction, over three different days and a varied number of satellites and PDOP. Therefore in Chapter 6, repeatability is not a criteria for the investigation because repeated results for the robot's test run would demonstrate consistency with those obtained with the GPS mounted trolley, however the offset in the GPS positional output due to the variation of the number of satellites and PDOP is a criteria.

These graphs (as shown previously in *Chapter 4*) have been used in order to reveal the predicted positional accuracy during a certain time interval, based on the projected paths of the satellites and their relative alignment to each other.

The radius of RA, the Spider error circle, was defined on the basis that 95% of the static data collection would fall within its radius.



Visibility

Fig. 6.1a: Number of satellites (satellite visibility) from 7:00 -19:00 (DST).



DOP Position

Fig.6.1b: Position dilution of precision from 7:00 -19:00 (DST).

6.1.1 Medium GADF

The conditions present during medium GADF are presented in Table 6.1.1 below:

Number of waypoints	19 waypoints total, 18 waypoints to be reached since robot starts on waypoint 1
Circular stages of closeness used	Yes
Radius of waypoint (RB)	0.7 m
Static data collection	15 minutes
Correction along the Easting	-1.325 m
Correction along the Northing	-2.641 m
Spider error circle (RA)	1.924 m
Average number of satellites (NOS)	7
Average PDOP	1.5

Table 6.1.1: The conditions	present duri	ng the medium	GADF testing	g
-----------------------------	--------------	---------------	--------------	---

Fig. 6.1.1 shows the static data collection of the GPS during the 15 minute interval before (default GPS output) and after positional correction.



Fig.6.1.1: Before and after GPS data correction – dark line indicate before and the light one after correction. The large circle shows RA where 95% of the static data falls within its boundaries.

In this work, it was never intended to use odometry. Nevertheless, numerous attempts were made to establish its potential suitability as a passive means for determining the robot's centre, in order to calculate the percentage of waypoints reached, but with little success. This was largely attributed to the presence of wheel misalignment that inevitably would create an imbalance of forces across the wheels, leading to eventual slip and inaccurate odometry. In addition, the presence of slack in the chain driven steering mechanism would create a 'wobble' effect in the sensors reading that would amplify this error. Therefore, to overcome this problem and obtain an estimate of the robot's centre, it was found that determining the offset between the robot's starting point and the GPS's initial starting position and maintaining that offset throughout would yield a realistic depiction of the robot's path. This method is to be used with caution since the test area is relatively small in size and the testing period was short. A longer test could change the offset used from the GPS and result in an inaccurate robot path, and larger distances would require that the topology be taken into account.

This offset (shift) is also used in the computation of the GPS error model, shown in Section 5.2, for simulating the relative position between the GPS positional output and the true position of the receiver.



6.1.1.1 Medium GADF – corrected GPS

In Fig. 6.1.1.1a the path of the robot and that of the GPS are shown through all 19 waypoints. The robot is shown to be travelling parallel to the GPS path. Fig. 6.1.1.1b shows the corresponding command velocities.



Fig.6.1.1.1a: Yellow line shows the Spider's path and the white line shows the GPS positional data – correction – medium GADF.



Fig.6.1.1.1b: The linear velocity, angular velocity and GADF respectively. The vertical lines represent the waypoints (2-19) - correction - medium GADF.

Table 6.1.1.1c shows the total distance travelled by the robot was 87.47 m from its starting waypoint. The values show the cumulative distance travelled by the robot and the distance at which the next waypoint was loaded. The values shown in Table 6.1.1.1c correspond to the positions of the vertical dashed lines in Fig. 6.1.1.1b.

Waypoints	1	2	3	4	5	6	7	8	9
Distance travelled	0	4 45	0.24	12.26	16.21	20.55	24.60	29.97	25 25
(m)	U	4.40	0.34	12.30	10.21	20.55	24.09	20.07	55.55

87.47

39.07 43.69 47.60 51.77 55.62 60.89 74.70 79.47 83.41

Table 6.1.1.1c: Distance at which the waypoints are reached - correction - medium GADF

The summary of the average velocity obtained from the GPS and the fuzzy controller are shown in Table 6.1.1.1d, in addition to the angular velocities from the fuzzy controller and the potentiometer. Given that some changes in the GADF can be seen in Fig. 6.1.1.1b the average of those are also presented. The average angular velocity results shown have been calculated with and without zeros, in order to compare the overall angular velocity during the trip to the average of the time the angular velocity commands were on.

Average absolute velocity from GPS	$0.295\ m/s\sim 0.3m/s$
Average absolute velocity from the fuzzy controller	$0.1887\ m/s\sim 0.2m/s$
Average angular velocity obtained from the potentiometer	with zeros 0.156 rad/s ~ 0.2 rad/s without zeros 0.371 rad/s ~ 0.4 rad/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.1663 rad/s ~ 0.2 rad/s without zeros 0.288 rad/s ~ 0.3 rad/s
Average GADF	$0.540 \text{ (max} = 0.54, \min = 0.51)$

Table 6.1.1.1d: Summary of the average velocities of the travelled distance - correction - medium GADF

Finally, the number of waypoints hit by the robot at the current waypoint setting is 6% as can be seen in Fig. 6.1.1.1e. However, if the waypoint radius is increased to 1.7, then 100% of the waypoints would have been reached. Since the robot travels parallel to the waypoints, this means that the robot is 1.7 m away from the GPS signal.



Fig.6.1.1.1e: Percentage of waypoints hit for test with varying radius size - correction - medium GADF

6.1.1.2 Medium GADF - corrected GPS - Simulation

The simulation of the robot under the same conditions present during the actual testing is shown below (Figure 6.1.1.2a).



Fig.6.1.1.2a: Simulation of the Spider in Matlab under similar conditions as those shown in the actual test run - correction - medium GADF

Fig. 6.1.1.2b shows the path taken by the robot and that taken by the simulated GPS.



Fig.6.1.1.2b: Simulation result showing the GPS data in white and the Spider path in yellow – correction – medium GADF.



Fig.6.1.1.2c: The linear velocity, angular velocity and GADF respectively for the simulated robot. The vertical lines represent the waypoints (2 – 19) – correction – medium GADF.

Table 6.1.1.2d shows the total travelled distance by the robot was 84.91 m from its starting waypoint. The values show the cumulative distance travelled by the robot and the distance at which the next waypoint was loaded. The values shown in Table 6.1.1.2d correspond to the positions of the vertical dashed lines in Fig. 6.1.1.2c.

Wa	5	1	2	3	4	5	6	7	8	
istance	travelle	ed (m)	0 4	5.32	9.22	13.28	17.14	21.32	25.16	29.01
10	11	12	13	3	14	15	16	17	18	19
39.01	43.52	47.31	51.4	46	55.27	59.29	72.85	76.87	80.91	84.91

The summary of the average velocity obtained from the GPS and the fuzzy controller are shown in Table 6.1.1.2e, in addition to the angular velocities from the fuzzy controller and the potentiometer. The GADF was held constant during the test since the number of

satellites and PDOP cannot be changed during the simulation. The average angular velocity results shown have been calculated with and without zeros, in order to compare the overall angular velocity during the trip to the average of the time the angular velocity commands were on. The additional offset present was equivalent to 0.6 m in the East direction and - 1.24 in the North direction.

Table 6.1.1.2e: Summary of the average velocities of the travelled distance - simulation - correction - medium GADF

Average absolute velocity from the fuzzy controller - Simulation	0.145 m/s ~ 0.15 m/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.054 rad/s ~ 0.05 rad/s without zeros 0.228 rad/s ~ 0.23 rad/s
GADF	0.543

Finally, the number of waypoints hit by the robot at the current waypoint setting is 0% as can be seen in Fig. 6.1.1.2f. If the waypoint radius had been increased to 1.7, then only an 11% improvement can be seen.



Fig.6.1.1.2f: Percentage of waypoints hit for test with varying radius size for simulation – correction – medium GADF

Note: Ideally, the simulation results would be superimposed onto the experimental for the figures displaying the velocity, angular velocity and GADF; however, since the total travelled distances do not match (compare 6.1.1.2d to 6.1.11c), the positions of the vertical lines would differ and also the velocity outputs, which would make the graph very crowded. The differences in the results will be elaborated on in the discussion.

6.1.1.3 Medium GADF - non-corrected GPS

From the time the medium GADF tests for the corrected GPS had been done, the number of satellites had increased on average to 7.8. That shows that 8 satellites were present during the non-corrected results for the majority of the time. The dilution of precision remained at 1.5.



Fig.6.1.1.3a: Result showing the GPS data in white and the Spider's path in yellow – no correction – medium GADF



Fig.6.1.1.3b: The linear velocity, angular velocity and GADF respectively. The vertical lines represent the waypoints (2 – 19) – no correction – medium GADF.

Fig. 6.1.1.3a shows the path of the robot and GPS, and Fig. 6.1.1.3b shows the corresponding velocity outputs. Table 6.1.1.3c shows the total travelled distance by the robot was 81.78 m, and Table 6.1.1.3d shows the average velocities during the testing.

	Table 6.1.1.3c: Distance at which the waypoints are reached - no correction - medium GAD												
Waypoints			s	1	2	3	4	5	6	7	8	9	
Distance travelled (m)				0	1.58	5.51	9.55	13.49	17.71	21.68	25.56	31.80	
	10	11	12		13	14	15	16	17	18	19		
	35.58	40.22	44.08	48	8.22	52.05	56.00	69.70	73.74	77.72	81.78		

Table 6.1.1.3d: Summary of the average velocities of the travelled distance - no correction - medium GADF

Average absolute velocity from GPS	0.295 m/s ~ 0.3m/s
Average absolute velocity from the fuzzy controller	0.186 m/s ~ 0.2 m/s
Average angular velocity obtained from the potentiometer	with zeros 0.178 rad/s ~ 0.18 rad/s without zeros 0.391 rad/s ~ 0.4 rad/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.1863 rad/s ~ 0.2 rad/s without zeros 0.295 rad/s ~ 0.3rad/s
Average GADF	0.547 (max = 0.61, min = 0.51)



Fig. 6.1.1.3e: Percentage of waypoints hit for test with varying radius size for test- no correction - medium GADF

6.1.1.4 Medium GADF - non-corrected GPS - Simulation

The simulation of the robot under the same conditions present during the actual testing is shown below (Fig. 6.1.1.4a).



Fig.6.1.1.4a: Simulation of the Spider in Matlab under similar conditions as those shown in the actual test run - no correction - medium GADF

Fig. 6.1.1.4b shows the path taken by the robot and that taken by the simulated GPS.



Fig.6.1.1.4b: Simulation result showing the GPS data in white and the Spider's path in yellow – no correction – medium GADF



Fig 6.1.1.4c: The linear velocity, angular velocity and GADF respectively for simulated robot. The vertical lines represent the waypoints (2 – 19) – no correction – medium GADF.

Table 6.1.1.4d shows the total travelled distance by the robot was 82.17 m from its starting waypoint. The values show the cumulative distance travelled by the robot and the distance at which the next waypoint was loaded. The values shown in Table 6.1.1.4d correspond to the positions of the vertical dashed lines in Fig. 6.1.1.4c.

Wa	ypoint	s	1 2	3	4	5	6	7	8	9
Distance	e travelle	ed (m)	0 2.46	6.35	10.48	14.43	18.69	22.54	26.46	32.70
10	11	12	13	14	15	16	17	18	19	

Table 6.1.1.4d: Distance at which the waypoints are reached - simulation -no correction - medium GADF

Average absolute velocity from the fuzzy controller - Simulation	$0.147 \text{ m/s} \sim 0.15 \text{ m/s}$
Average angular velocity obtained from the fuzzy controller	with zeros 0.06 rad/s without zeros 0.235 rad/s ~ 0.24 rad/s
GADF	0.54

Table 6.1.1.4e: Summary of the average velocities of the travelled distance - simulation - no correction - medium GADF

The summary of the average velocity obtained from the GPS and the fuzzy controller are shown in Table 6.1.1.4e, in addition to the angular velocities from the fuzzy controller and the potentiometer. The GADF was held constant during the test since the number of satellites and PDOP cannot be changed during the simulation. The average angular velocity results shown have been calculated with and without zeros, in order to compare the overall angular velocity during the trip to the average of the time the angular velocity commands were on. Finally, the number of waypoints hit by the robot at the current waypoint setting is 0% as can be seen in Fig. 6.1.1.4f. If the waypoint radius is increased to 1.3, an 89% improvement would be noted.



Fig. 6.1.1.4f: Percentage of waypoints hit for test with varying radius size for simulation – no correction – medium GADF

6.1.2 High GADF

Table 6.1.2 shows the conditions present during the following series of tests.

Number of waypoints	19 waypoints total, 18 waypoints to be reached since robot starts on waypoint 1
Circular stages of closeness used	Yes
Radius of waypoint (RB)	0.7 m
Static data collection	15 minutes
Correction along the Easting	-2.082 m
Correction along the Northing	-0.512 m
Spider error circle (RA)	3.374 m
Average number of satellites (NOS)	10
Average PDOP	1.3

Table 6.1.2: Conditions present during the high GADF testing

Fig. 6.1.2 shows the static data collection of the GPS during the 15 minute interval before (default GPS output) and after positional correction.







6.1.2.1 High GADF - corrected GPS

In Fig 6.1.2.1a the robot's and GPS path can be seen passing through the waypoints and are just slightly separated. Fig 6.1.2.1b shows the corresponding velocity command outputs.



Fig.6.1.2.1a: Simulation result showing the GPS data in white and the Spider's path in yellow – correction – High GADF



Fig.6.1.2.1b: The above figure shows the linear velocity, angular velocity and GADF respectively. The vertical lines represent the waypoints (2 – 19) –correction – high GADF.

Table 6.1.2.1c shows the total distance travelled by the robot was 85.87 m from the starting waypoint. These values show the cumulative distance travelled by the robot up to that point and the distance at which the next waypoint was loaded. The distances in the table correspond to the vertical lines in Fig 6.1.2.1b.

	Table 6.1.2.1c: Distance at which the waypoints are reached - correction - high GADF											
	Wa	ypoints	5	1	2	3	4	5	6	7	8	9
C	istance	travelle	ed (m)	0	4.34	8.31	12.69	16.59	20.88	24.94	28.98	35.38
	10	11	12	1	13	14	15	16	17	18	19	
	39.10	43.74	47.53	51	1.85	55.71	59.81	73.62	77.64	81.71	85.87	

The summary of the average velocities obtained from the GPS, the fuzzy controller and the potentiometer, during the test run, are shown in Table 6.1.2.1d. In addition, the average GADF is also presented.

Average absolute velocity from GPS	$0.354\ m/s\sim 0.4m/s$
Average absolute velocity from the fuzzy controller	$0.45\ m/s\sim 0.5m/s$
Average angular velocity obtained from the potentiometer	with zeros 0.261 rad/s ~ 0.3rad/s without zeros 0.43 rad/s ~ 0.4 rad/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.236 rad/s ~ 0.2 rad/s without zeros 0.350 rad/s ~ 0.4 rad/s
Average GADF	0.756 (min = 0.61, max = 0.767)

Table 6.1.2.1d: Summary of the average velocities of the travelled distance - correction - high GADF

The percentages of waypoints hit are presented in Fig. 6.1.2.1e, where from a radius of 0.6 m onwards a 100% hit is recorded. The result also reveals that the robot is less than 0.6 m away from the GPS receiver.



Fig. 6.1.2.1e: Percentage of waypoints hit for test with varying radius size for test - correction - high GADF
6.1.2.2 High GADF - corrected GPS - Simulation

The simulation of the robot under the same conditions present during the actual testing is shown below (Fig. 6.1.1.2a).



Fig.6.1.2.2a: Simulation of the Spider in Matlab under similar conditions as those shown in the actual test run - correction - High GADF.

Fig. 6.1.2.2b shows the path taken by the robot and that taken by the simulated GPS.







Fig 6.1.2.2c: The linear velocity, angular velocity and GADF respectively for simulated robot. The vertical lines represent the waypoints (2 – 19) –correction – high GADF.

Table 6.1.2.2d shows the total travelled distance by the robot was 84.73 m from its starting waypoint. The values show the cumulative distance travelled by the robot and the distance at which the next waypoint was loaded. The values shown in Table 6.1.2.2d correspond to the positions of the vertical dashed lines in Fig 6.1.2.2c.

Т	able 6.1.	2.2d: Dist	tance at v	vhi	ch the	waypoir	nts are rea	ched - si	nulation	- correct	ion – higl	h GADF
	Wa	ypoint	5	1	2	3	4	5	6	7	8	9
C	istance	travelle	ed (m)	0	5.03	8 8.92	13.01	16.90	21.13	24.97	28.88	35.09
	10	11	12		13	14	15	16	17	18	19	
	38.81	43.32	47.14	5	1.32	55.13	59.09	72.64	76.66	80.70	84.73	

Average absolute velocity from the fuzzy controller - Simulation	0.2554 m/s ~ 0.26 m/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.053 rad/s ~ 0.05 rad/s without zeros 0.227 rad/s ~ 0.23 rad/s
GADF	0.7665

Table 6.1.2.2e: Summary of the average velocities of the travelled distance - simulation - correction - high GADF

The summary of the average velocity obtained from the GPS and the fuzzy controller are shown in Table 6.1.2.2e, in addition to the angular velocities from the fuzzy controller and the potentiometer. The GADF was held constant during the test since the number of satellites and PDOP cannot be changed during the simulation. The average angular velocity results shown have been calculated with and without zeros, in order to compare the overall angular velocity during the trip to the average of the time the angular velocity commands were on. Finally, the number of waypoints hit by the robot at 0.7 m is 89% as can be seen in Fig. 6.1.2.2f. If the waypoint radius had been increased to 1.1, the results would have been improved to 100%.



Fig. 6.1.2.2f: Percentage of waypoints hit for test with varying radius size for simulation – correction – high GADF

6.1.2.3 High GADF – non-corrected GPS

In Fig 6.1.2.3a the Spider's and GPS path can be seen passing through the waypoints and are just slightly separated. Fig 6.1.2.3b shows the corresponding velocity command outputs.



Fig.6.1.2.3a: Result showing the GPS data in white and the Spider's path in yellow – no correction – high GADF



Fig.6.1.2.3b: The above figure shows the linear velocity, angular velocity and GADF respectively. The vertical lines represent the waypoints (2 – 19) – no correction – high GADF

Table 6.1.2.3c shows the total distance travelled by the robot was 81.81 m from the starting waypoint. These values show the cumulative distance travelled by the robot up to that point and the distance at which the next waypoint was loaded. The distances in the table correspond to the vertical lines in Fig 6.1.2.3b.

	Table	6.1.2.3c:	Distanc	e at whic	h the way	points an	e reached	d – no co	rrection -	- high GA	ADF
	Wa	ypoints	5	1 2	3	4	5	6	7	8	9
[Distance	e travelle	d (m)	0 3.14	6.88	10.68	14.64	18.86	22.76	26.62	32.93
	10	11	12	13	14	15	16	17	18	19	
	36.61	40.87	44.62	48.65	52.48	56.47	70.05	73.90	77.86	81.81	

The summary of the average velocities obtained from the GPS, the fuzzy controller and the potentiometer during the test run are shown in Table 6.1.2.3d. In addition, the average GADF is also presented.

Average absolute velocity from GPS	0.369 m/s ~ 0.4 m/s
Average absolute velocity from the fuzzy controller	0.37 m/s
Average angular velocity obtained from the potentiometer	with zeros 0.184 rad/s without zeros 0.402 rad/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.194 rad/s without zeros 0.326 rad/s
Average GADF	0.756 (min = 0.61, max = 0.767)

Table 6.1.2.3d: Summary of the average velocities of the travelled distance - no correction - high GADF

The percentages of waypoints hit are presented in Fig. 6.1.2.3e. It can be seen that, from a radius of 1.3 m and greater, a 100% hit is recorded. The result also reveals that the robot is less than 1.3 m away from the GPS receiver.



Fig. 6.1.2.3e: Percentage of waypoints hit for test with varying radius size for test – no correction – high GADF



6.1.2.4 High GADF - non-corrected GPS - Simulation

The simulation of the robot under the same conditions present during the actual testing is shown below (Fig. 6.1.1.4a).



Fig.6.1.2.4a: Simulation result showing the GPS data in white and the Spider's path in yellow – no correction – high GADF

Fig. 6.1.2.4b shows the path taken by the robot and that taken by the simulated GPS.



Fig. 6.1.2.4b: Simulation result showing the GPS data in white and the Spider's path in yellow – no correction – high GADF



Fig 6.1.2.4c: The linear velocity, angular velocity and GADF respectively for simulated robot. The vertical lines represent the waypoints (2 – 19) – no correction – high GADF.

Table 6.1.2.4d shows the total travelled distance by the robot was 83.89 m from its starting waypoint. The values show the cumulative distance travelled by the robot and the distance at which the next waypoint was loaded. The values shown in Table 6.1.2.4d correspond to the positions of the vertical dashed lines in Fig. 6.1.2.4c.

Wayp	oints	1	2	3	4	5	5	6	7	8	9
Distance tra	avelled (m) 0	4.07	7.95	12.00	15.	95 20	0.17	24.06	27.99	34.24
10	11	12	13	14	15	5	16	17	18	3	19
27.09	12 18	46 30	50 47	54.3	1 58	27	71.85	75.8	1 70	01 8	3 80

ingi 02	ADI
Average absolute velocity from the fuzzy controller - Simulation	0.249 m/s ~ 0.25 m/s
Average angular velocity obtained from the fuzzy controller	with zeros 0.070 rad/s ~ 0.07 rad/s without zeros 0.233 rad/s ~ 0.2 rad/s
GADF	0.7665

Table 6.1.2.4e: Summary of the average velocities of the travelled distance - simulation - no correction - high GADF

The summary of the average velocity obtained from the GPS and the fuzzy controller are shown in Table 6.1.2.4e, in addition to the angular velocities from the fuzzy controller and the potentiometer. The GADF was held constant during the test since the number of satellites and PDOP cannot be changed during the simulation. The average angular velocity results shown have been calculated with and without zeros, in order to compare the overall angular velocity during the trip to the average of the time the angular velocity commands were on. Finally, the number of waypoints hit by the robot at the current waypoint setting of 0.7 m is 6%, as can be seen in Fig. 6.1.2.4f. If the waypoint radius had been increased to 1.5, the waypoint hit would be at 100%.



Fig. 6.1.2.4f: Percentage of waypoints hit for test with varying radius size for simulation – no correction – high GADF

6.2 Spider Tests B

As previously shown, a waypoint hit is considered when the robot passes through a waypoint. Therefore, a 100% hit at a radius of (for example) 0.7 m would indicate that the robot is within 0.7 m of the GPS receiver. However, the GPS positional output fluctuations can have an effect on reaching the waypoint and loading the next waypoint. It is true that a 100% waypoint hit can be achieved, but does that indicate that it was reached on its first attempt, or as a result of the robot having to drive back and forth to reach it?

Given that the GPS positional fluctuations are known, and that there is a limit to how small a waypoint radius can be used to ensure spatial matching (due to the variability and bias errors of waypoints selected from an orthorectified image – see *Chapter 4*), the comparison of the implementation of circular stages of closeness versus the approach of reaching a single point in space is presented in this section.

6.2.1 Circular Stages of Closeness vs. None

The fuzzy control system and the GPS corrected-data are used in both the point-to-point *and* the circular stages of closeness approaches to effectively demonstrate the two under constant test conditions.

6.2.1.1 Point to Point



Fig.6.2.1.1: The path of the robot (yellow) compared to the path of the GPS is shown for a point to point following approach.

Fig 6.2.1.1 shows the paths of the robot and the GPS in an attempt to reach the first waypoint. The circles (radius 0.1 m) shown in the image have been drawn to help identify the scale of the image. The result in the figure shows the data gathered over the course of 30 seconds without the GPS reaching the point. Following this failed attempt the autonomous navigation of the robot was stopped.

6.2.1.2 Point to Point - Simulation



Fig.6.2.1.2: The path of the robot (yellow) compared to the path of the GPS is shown for a point to point following approach of the simulation.

Fig. 6.2.1.2 shows the simulated path taken by the GPS and the Spider under the same conditions as in the actual testing. Once again the GPS failed to reach the waypoint and therefore load the following one. The differences between the experimental and simulated results will be clarified in *Chapter 7*.

6.2.1.3 Circular Stages of Closeness

It is necessary to state that even with the use of circular stages of closeness a waypoint radius as small as 0.1 m would most likely not yield satisfactory results, which would include reaching the waypoint at first attempt, due to the presence of fluctuations in the GPS positional data, as has been previously discussed.

Fig 6.2.1.2 shows the path of the robot and that of the GPS using the circular stages of closeness approach at a radius of 0.1 m. Waypoints 4, 5, 6 and 7 are those of interest. Waypoints 2 and 3 (the first two) are not shown since they were passed successfully on first trial. Waypoint 4, at the bottom of the image, was also reached; however, as the robot approached waypoints 5, 6 and 7, the robot required several attempts, but nevertheless were successful. The test was terminated at waypoint 10.



Fig.6.2.1.2: The path of the robot (yellow) compared to the path of the GPS is shown using circular stages of closeness at the radius of 0.1 m.

6.2.1.4 Circular stages of Closeness - Simulation

Fig 6.2.1.4 shows the path of the simulated GPS and robot passing through waypoints of a radius of 0.1 m. The figure clearly shows that that the waypoints were successfully reached even with the presence of an induced GPS error model.



Fig.6.2.1.4: The path of the robot (yellow) compared to the path of the GPS is shown using circular stages of closeness at the undesirable radius of 0.1 m of the simulation.

6.3 Spider Test C

In Spider Test B the effect of the circular stages of closeness on the performance of the robot in reaching waypoints has been shown. In Spider Test C, the seven cases of circular model interactions are presented by varying the Spider error circle, RA.

6.3.1 Varying RA

In this section the results are presented qualitatively, since the images clearly show the circular interactions between the waypoint and the Spider error circle.

The tests were all performed under similar conditions with 10 visible satellites for the majority of the time. However, the dilution of precision varied from 1.3 to 1.5. This change, even though it might appear to be slight, can have an effect on the number of waypoints hit, but does not affect the circular stages of closeness. RB was chosen at 1.0 m, and RA was chosen *not* on the basis of the 95th percentile from the static data collection, but instead on the basis of ensuring that all seven circular stages of closeness can be demonstrated: 0.25 m, 0.5 m, 0.75 m, 1.0 m, 1.25 m, 2.0 m and 2.25 m. Not all of the data points are plotted in order to avoid overcrowding of the images. Both the experimental and simulated results are presented. The third waypoint is used to show the various stages of closeness for each of the seven cases:

Case 1:

Fig. 6.3.1-1 shows the circular stages of closeness for the first case of the actual test results and Fig. 6.3.1-1S shows the equivalent test under simulation.



Fig.6.3.1-1: Circular stages of closeness for RA = 2.25 m.





Case 2:

Fig. 6.3.1-2 shows the circular stages of closeness for the second case of the actual test results and Fig. 6.3.1-2S shows the equivalent test under simulation.



Fig.6.3.1-2: Circular stages of closeness for RA = 2.0 m.



Fig.6.3.1-2S: Circular stages of closeness for RA = 2.0 m (simulation)

Case 3:

Fig. 6.3.1-3 shows the circular stages of closeness for the third case of the actual test results and Fig. 6.3.1-3S shows the equivalent test under simulation.









175

Case 4:

Fig. 6.3.1-4 shows the circular stages of closeness for the fourth case of the actual test results and Fig. 6.3.1-4S shows the equivalent test under simulation.



Fig.6.3.1-4: Circular stages of closeness for RA = 1 m.





Case 5:

Fig. 6.3.1-5 shows the circular stages of closeness for the fifth case of the actual test results and Fig. 6.3.1-58 shows the equivalent test under simulation.



Fig.6.3.1-5: Circular stages of closeness for RA = 0.75 m.





Case 6:

Fig. 6.3.1-6 shows the circular stages of closeness for the sixth case of the actual test results and Fig. 6.3.1-6S shows the equivalent test under simulation.







Fig.6.3.1-6S: Circular stages of closeness for RA = 0.5 m (simulation)

Case 7:

Fig. 6.3.1-7 shows the circular stages of closeness for the sixth case of the actual test results and Fig. 6.3.1-7S shows the equivalent test under simulation.



Fig.6.3.1-7: Circular stages of closeness for RA = 0.25 m.



Fig.6.3.1-7S: Circular stages of closeness for RA = 0.25 m (simulation)

6.4 Chapter Summary

a) Spider Tests A:

Even though the static data collection for the first set of results (medium GADF) shows a better scatter than those of the (high GADF), the significance of the PDOP and the number of satellites in view prevails.

For both the experimental and simulated results:

- The tests were performed for both corrected and non-corrected GPS data and under differing GADF conditions.
- The paths of the robot and that of the GPS are superimposed onto the orthorectified image, showing the process of waypoint navigation.
- The fuzzy controller velocity outputs are shown under the GADF condition present.
- The distance at which the waypoints are reached and when the following ones are loaded is presented.
- A summary of the average angular and linear velocities from the sensors and the controllers are compared.
- The percentage of waypoints hit for each particular case is presented in histograms.
- b) Spider Tests B:
 - Navigating the robot to a single point compared to a circular waypoint is shown using GPS corrected data and the same fuzzy controller. This has been done experimentally and in the simulation.
- c) Spider Tests C:
 - The seven cases of circular stages of closeness are presented by varying the Spider error circle (RA) and showing the circular interactions.

Chapter 7: Discussion

In this section of the work, the results from the previous chapter (*Experimental and Simulated Results*) will be discussed and scrutinised in relation to the proposed theoretical framework.

It is broken down into the following sections:

- a. The critique between using corrected and non-corrected GPS data in terms of the waypoints hit and the path adherence of the robot under medium and high GADF - for both the actual and the simulated results.
- b. The critique of testing the robot under a medium and high GADF, for GPS corrected data, and how it affects the number of waypoints hit and the response of the fuzzy controller for both the actual and the simulated results.
- c. The critique on using the circular stages of closeness model compared to using a single point approach, and a discussion on the effect of varying the radius RA on the performance of the control system – for both the actual and simulated results.
- d. The heading control strategy outcome, that governs the shortest distance and that requires the least amount of movement, will be discussed.
- e. A review of the simulation
- f. Finally, a discussion of future work.

Chapter pre-requisite: 6.0 Experimental and Simulated Results Nature of Chapter: Discussion/Theoretical.

7.1 Corrected vs. Non-Corrected GPS Data

In section 4.3 (GPS), it was shown how the GPS positional correction would improve the accuracy of the GPS output if it is accommodated for (Fig. 4.3.2e). These tests were performed on a GPS-mounted trolley and driven along a marked chalk line on the ground. The purpose of those tests was to show the significance of this correction procedure. However, given that the contraption was manually guided through the points it is important to judge how the proposed system would perform under autonomous navigation. It is necessary to note that in the previous tests the number of waypoints hit was defined in terms of the GPS data passing through the waypoints. This GPS correction had shown to reduce the positional offset between the positional output and the GPS receiver. Therefore, for the test results presented, only if the robot passes through a waypoint is it considered a hit. This creates a more critical margin of analysis for the system's effectiveness, since the robot's spatial position is of higher significance. Judging only by waypoints hit does not give the big picture; consequently the element of path adherence is added, as shall become clearer.

Holden, M. [2004] acknowledges that GPS positional data can be inconsistent from one day to the next using a low-cost GPS receiver; however, this work shows the effect of correcting this data in order to produce usable, more accurate readings.

A significant component of this research is the concept that even slight variations in waypoint radius have an effect on the number of waypoints hit. Snider, J. *et al* [2004] set their waypoint radius (seemingly arbitrarily) to 0.6 m, for their autonomous robot using DGPS. This research shows that using corrected WAAS/EGNOS enabled GPS, it is possible to ensure that the robot could pass through radii as small as 0.4 m. Therefore, what was their criterion for having chosen a waypoint of that dimension, given that they are using DGPS?

Prior to the testing, the Trimble Planning software was used to determine the expected time period for improved positional accuracy, which is shown in section 7.1.1.

7.1.1 Waypoints Hit and Path Adherence

The *Spider Tests A* were performed in order to show the number of waypoints hit by the Spider under different PDOP and number of satellites, for both the actual and the simulated results.

7.1.1.1 Medium GADF

From Fig 7.1.1.1a it can be seen that despite the use of the GPS corrected system, the number of waypoints hit is lower.



Fig.7.1.1.1a Number of waypoints hit:■ - with correction - no correction (Medium GADF)

But it is important to note that during the testing where *no GPS correction* was used the number of satellites had increased from 7 to 8 while the PDOP remained unchanged. This shows the huge influence the presence of one additional satellite has on the overall performance of this method. The reason why the corrected GPS reached 100% beyond the 1.6 m radius but remained at 83% for the non-corrected can be seen in Fig 7.1.1.1b.



Fig. 7.1.1.1b: Corrected GPS data (left image) and non-corrected (right image) at medium GADF. The yellow lines represent the Spider's path and the white the GPS path.

From Fig. 7.1.1.1b it can be seen that the path the Spider takes (yellow lines), using the corrected data, is in fact in parallel alignment to the waypoints, even though the Spider does not hit all the waypoints. On the other hand, in the figure to the right (non-corrected), the robot appears to pass through the majority of the points but never reaches the furthest waypoints (3 in total), showing the presence of a shift between the GPS receiver and the Spider. It is this shift that accounts for the non-corrected waypoints hit to stabilise at 83% and not improve as the radius is increased up to 1.9 m. Since the corrected GPS data accommodates for this effect, by increasing the radius the waypoints hit reaches the 100% margin. In both cases the GPS data (white lines) pass through all the waypoints, showing that the controller is functioning appropriately. It is expected that if the number of satellites had not changed during the course of the testing that the path the Spider would have taken, during the non-corrected testing, would be more likely to be running parallel to the waypoints.

The test was performed from 14:35 to 14:40 for the corrected and 14:46 to 14:52 for the non-corrected data (refer to the Trimble planning software, Fig. 7.1.1.1c). During the first period of testing the number of satellites that can be expected is 9, however the actual number of satellites present was 7. Even though the number of satellites for the second period was expected to decrease from 10 to 8, the number of satellites actually went up from 7 to 8. For the Dilution of Precision, a margin of 1.5 was measured, but the figure shows a much higher expectation (2.20 to 2.80). These results indicate that the Trimble planning software should not be used as a definitive measure of the expected positional accuracy, but rather as a tool for guidance.



Fig. 7.1.1.1c: Predicted satellite visibility and PDOP during the time of testing for medium GADF.

186

7.1.1.2 Medium GADF - Simulation

The conditions from the results of the medium GADF were used in the simulation in order to expose the virtual robot to similar working conditions.

In Fig. 7.1.1.2a, the result of the simulated waypoints hit for the corrected and the noncorrected is shown.



Fig.7.1.1.2a: Number of waypoints hit simulation: ■ - with correction, - with no correction (Medium GADF)

From the graph it is clearly visible that the non-corrected data shows a significant advantage over the corrected. At a waypoint radius of 1.3 a 78% reduction in the simulation waypoints hit occurs compared to 33% in the actual tests. But by comparing the simulation results to the actual it can be seen that they do show some discrepancies and similarities. In terms of the non-corrected there is approximately 5% difference from 1.3 m radius onwards, which corresponds to one additional waypoint (1/18 x 100). The values displayed have been rounded to whole numbers; nevertheless, both corrected results lag behind the non-corrected results. But it is crucial to remember the effect the PDOP and
one additional satellite have on the outcome of the waypoints hit, as previously discussed. In the simulation, the PDOP and number of satellites are fixed, therefore the effect of temporary change in both those factors is not taken into account. From this it can be concluded that the simulation is in fact an underestimate of the actual performance. This does raise the issue that the simulation perhaps needs to be revisited. As previously mentioned, it is necessary to observe the path adherence results shown in Fig 7.1.1.2b to get the overall picture.



Fig. 7.1.1.2b: Corrected GPS data (left image) and non-corrected (right image) at medium GADF of the simulation. The yellow lines represent the Spider's path and the white the GPS path

Fig. 7.1.1.2b shows the simulation results of the Spider's path and the GPS path. By comparing the path adherence to those of the actual results, a similarity can be seen. Once again, it is clearly visible that in the image with the non-corrected GPS data several waypoints were never reached by the robot, similar to in the actual results. This is due to the presence of this shift along the length of the waypoints. As for the left hand image, the robot appears to be travelling for the majority of the time parallel to the waypoints and the change in direction of travel occurs opposite the waypoint. Compared to the actual results, the robot path in the simulation is smoother. However, some small oscillations in the path can still be noted.

The effect of this oscillating behaviour that we can see in the actual results will be discussed in section 7.2.2.

7.1.1.3 High GADF



In Fig. 7.1.1.3a, a slightly different result to the medium GADF can be seen.

Fig.7.1.1.3a Number of waypoints hit: ■ - with correction, ■ - no correction (High GADF)

Fig 7.1.1.3a shows a clear improvement in the number of waypoints hit with GPS correction. The number of satellites and the PDOP were consistent during the testing, at 10 and 1.5, respectively. The results show that the Spider has a positional accuracy of within 0.6 m. By referring to Fig 7.1.1.3b the results of the path adherence can be seen.



Fig. 7.1.1.3b: Corrected GPS data (left image) and non-corrected (right image) at high GADF. The yellow lines represent the Spider's path and the white the GPS path.

Given that the overall number of satellites was higher than in the previous test results (medium GADF), the positional offset between the non-corrected and corrected data should be smaller than before. In the left-hand image of Fig 7.1.1.3b, it is quite clear that the robot passes through each waypoint and that the GPS and robot paths follow each other closely, compared to the non-corrected. The results reveal the effect the changes in the number of satellites and PDOP have on the number of waypoints hit and positional offset between the GPS receiver output and the robot.

The tests were performed from 16:12 - 16:16 for the corrected, and from 16:26 - 16:31 for the non-corrected data. By referring to Fig. 7.1.1.3c it can be seen that the expected number of satellites is 9, but the GPS had been showing 10. For the PDOP a much higher range is shown (~ 3.0 to 3.10) compared to the measured 1.3. Nevertheless, the number of satellites had been consistent (even) for the duration of the two tests, which matches with the consistency (evenness) shown in the figure, but that does not apply for the PDOP.



Fig. 7.1.1.3c: Predicted satellite visibility during the time of testing and the PDOP for high GADF.

Unfortunately, a low GADF working condition did not occur during long periods of testing. But, the results are expected to show an even further reduction in waypoints hit compared to the medium.



7.1.1.4 High GADF – Simulation

Once again the same working conditions for the actual test results were used in the simulation. The percentage of waypoints hit for both scenarios can be seen in Fig. 7.1.1.4a.



Fig.7.1.1.4a: Number of waypoints hit simulation:
- with no correction,
- with correction
(Medium GADF)

The PDOP and the number of satellites were constant during the testing. Given these factors, a higher resemblance between the actual results and the simulation can be seen compared to the case of medium GADF, where changes in the PDOP and number of satellites had occurred. In both the actual and simulation results the number of waypoints for the corrected data was at 100% from 1.1 m onwards. Prior to that the simulation shows a decrease compared to the actual. In future work, it will be necessary to establish the reason behind these large lags in the non-corrected simulated results compared to the actual, which would potentially lead to a reinvestigation of the GPS error model. It must be reminded that the GPS error model is only an approximate model based on experience. Given that the simulation reveals underestimated results it can be argued as being a positive characteristic, since the actual tests would always yield better results.



Fig. 7.1.1.4b: Corrected GPS data (left image) and non-corrected (right image) at high GADF of the simulation. The yellow lines represent the Spider's path and the white the GPS path.

In Fig. 7.1.1.4b the simulation results of the high GADF are shown. By close comparison to the actual results obtained they can be seen to follow a close trend. The robot's path in the left-hand image, Fig. 7.1.1.4b, appears to pass through the majority of the waypoints. Compared to the medium GADF in the simulated and experimental results, the robot's path for the non-corrected is in fact travelling parallel to the waypoints and changing its direction of travel opposite to the waypoints. This shows the effect of the PDOP and the number of satellites has on the path adherence and waypoint hit. Similarly, the robot's path is not very smooth but does not show the extent of the oscillatory motion of the actual results.

Work carried out by other researchers can be compared to some of the results obtained here. Witte, T. and Wilson, A. [2005] have shown that for a series of 25 tests on a marked line, using WAAS/EGNOS enabled GPS, yielded an offset of 0.11 m (from a range of 0.05 to 0.2 m). The number of satellites they observed ranged from 4 -10 for the duration of testing, with an average of 8 satellites in view. (In the work described in this section, the effect of the number of satellites and PDOP is quite visible between the medium and high GADF, under the influence of 7 satellites for *medium* and 10 satellites for *high*.) Therefore, such an offset could only be accomplished under a fixed number of satellites. The authors do show an improvement between WAAS/EGNOS-enabled GPS versus none (as do Shair, S. *et al* [2006]). However, the authors' offset therefore refers to the fluctuation in the GPS. No spatial matching had been made between the GPS positional data and the true ground

positional data; therefore, the authors' presented results were plotted on an arbitrary origin. What benefit would such an offset have, if its true ground position can not be related to the GPS coordinate system?

7.2 The GADF effect on controller

Previously, both effects of GPS correction and variation of GADF were shown in relation to the waypoints hit and path adherence. For the following work the difference between the waypoints hit of the corrected medium GADF to the corrected high GADF will be analysed followed by the fuzzy controller's response to this variation. The test conditions for the PDOP and number of satellites are shown in Table 7.2.

[able]	7.2: Summary	of num	per of satellites	and PDOP 1	for both	GADF te	est conditions.
					and the lot of the lot		

	Number of Satellites	PDOP
Medium GADF	7	1.5
High GADF	10	1.3

7.2.1 Waypoint reaching

7.2.1.1 Medium GADF vs. High GADF

From Fig. 7.2.1.1 it can be clearly seen that the number of waypoints hit for the tests taken during high GADF show significant improvement to its medium counterpart. At a waypoint radius of 0.6 a 94% improvement can be seen. From 1.7 m onwards they both measure a 100% waypoint hit.



Fig.7.2.1.1: Number of waypoints hit: - Medium GADF, - High GADF

The issue therefore arises as to what effect this would have on the performance of the controller. Prior to that, the corrected simulated medium GADF and High GADF results are shown.

7.2.1.2 Medium GADF vs. High GADF - Simulation

Fig. 7.2.1.2 compares the medium GADF to the high GADF results of the simulation. The trend between the actual results and the simulated is quite apparent. Both show an improvement in the waypoints reached at a higher GADF. At a radius of 1.1 m the simulated results show a 92% difference compared to the actual of 89%, and at 1.9 m a 44% difference between the simulated medium GADF and actual can be seen. This difference shows that the simulation is an underestimate to the true performance.



Fig.7.2.1.2: Number of waypoints hit□ - Medium GADF,□ - High GADF

7.2.2 Fuzzy controller

The control strategy approach used in this research is a novel contribution, as has been previously shown. However, the effect of this strategy in real life testing has not been critiqued yet. Looking at the Spider's paths displayed in Fig 7.2.2a, it is apparent that there is room for improvement in terms of the output smoothness.



Fig. 7.2.2a: The left hand images show the Spider's path (yellow) and GPS path (white) at medium GADF, with the corresponding fuzzy controller's linear and angular velocities. The right hand images show the same results at a high GADF.

The results obtained are slightly different than anticipated, as both medium and high GADF results show oscillations in the output. It appears that the principle of having the angular velocity commands and the velocity commands increase with an increase in GADF might not be an optimal approach after all, even though the simulated output Fig 7.2.2b shows the success of this approach. See Fig 5.3.3c (*Chapter 5*) for the fuzzy rules.



Fig. 7.2.2b: The left hand images show the Spider's path (yellow) and GPS path (white) at medium GADF, with the corresponding fuzzy controller's linear and angular velocities of the simulation. The right hand images show the same results at a high GADF.

By referring to the angular velocity commands in Fig 7.2.2a and Fig 7.2.2b it can be seen that the outputs from the actual tests are larger and more frequent than the simulation. This oscillating effect is an unfavourable characteristic that can be loosely attributed to overshoot. For quantitative measures of the results, Table 7.2.2c is presented. This table prompts the need for perhaps having to revisit the control strategy for the angular velocity output.

The role of the fuzzy logic controller was to have a means for combining the multiple elements of the research, mainly to demonstrate the effect of the variation of the number of satellites, the PDOP, and the circular stages of closeness on the robot's behaviour. In addition, this research has been concerned with the relationship between the real and the simulated environment. The parameters used for these experiments have been justified for these purposes and can be seen in Chapter 5 (section 5.3.3); however, for the purpose of

optimising the fuzzy logic parameters to enhance/improve robot control, further work is recommended.

Re-adjusting the fuzzy parameters in the simulation environment did not show any changes in the output of the robot, even with the presence of the introduced GPS error model. Any adjustments of these parameters in the simulated environment, however, would most likely affect the response of the robot experimentally, due to the presence of other sensor errors, dynamic and mechanical constraints. Adjusting the fuzzy parameters for the real robot, however, would be counterproductive to the purpose of creating the simulation environment to tune the controller, because it is expected that the experimental robot would behave comparably to the simulation. In order to obtain comparable robot behaviour under similar fuzzy parameters, it is necessary for any future work to focus on enhancing the simulation.

	Med GADF (rad/s)	High GADF (rad/s)	Med GADF (m/s)	High GADF (m/s)
Fuzzy controller	0.1663/0.288(NZ)	0.236/0.35(NZ)	0.1887	0.45
Sensors	0.156/0.371(NZ)	0.261/0.43(NZ)	0.295	0.354
Simulation	0.054/0.23(NZ)	0.053/0.23(NZ)	0.145	0.26

Table 7.2.2c: Summary of the angular velocities from the controller, sensors and simulation for medium and high $GADF - NZ^4$ (no zeros)

From Fig 7.2.2a it can be seen that even as the robot approaches the waypoint, the angular velocity is vividly attempting to adjust the robot's position. As soon as the robot reaches the waypoint and following is loaded, a peak in the angular velocity can be seen. In comparison to the simulated results in Fig 7.2.2b, this is not present.

It would be difficult to point out the sole flaw to the control system, given that the simulated results have shown a satisfactory output. Even though the fuzzy controller

⁴ NZ (No zero) refers to the average of the angular velocity only during the times where it was activated. This provides an average for the magnitude of the steering velocity command.

eliminates the need for complex dynamic models, dynamic factors (in addition to mechanical and/or electrical characteristics) are believed to be a cause of the robot's behaviour.

Referring back to Table 7.2.2c, it can be seen that during the high GADF the overall velocity outputs were higher than the medium GADF results. The same applies to the actual output from the sensors. On the other hand, it can be noted that the sensor outputs do not match the fuzzy controller outputs under the same working conditions.

The Spider is equipped with several dc electric motors and a central control unit that all rely on battery power. For the tests, a brand-new battery was purchased to ensure that the robot was performing under optimal and consistent conditions. Given that the steering is controlled by a dc motor, any reduction in the battery's performance will change the angular velocity output. Therefore, matching the angular velocity outputs from the fuzzy controller to the actual robot would prove to be difficult.

Accommodating for the battery state in the fuzzy logic controller would be possible; however, in order to achieve this, continuous monitoring of the battery's charge would need to be carried out. A fourth input based on this information that would be related to the angular velocity command could be introduced into the second stage fuzzy controller.

Furthermore, the weight associated with the addition of the platform with all of its sensors has also affected the performance of the robot. This added weight has increased the demand on the engine and the electric motors beyond the manufacturer's recommendations.

Even though it is not apparent in Table 7.2.2c, inconsistencies in the velocity output were observed over long periods of testing, which were largely related to the hydraulic drive mechanism. It was often observed that the robot would not be able to match the given velocity command; for example, at a maximum velocity output the robot begins to creep and then shortly afterwards begins to accelerate to the desired speed.

Another important issue is the wheel alignment. It has been observed that alteration in the wheel alignment is inevitable. This effect causes the dynamics of the robot to change and the imbalance of forces acting on the wheel leads to eventual slip.

All of these effects show that matching the fuzzy controller outputs to the actual is a difficult task, and adjusting for these effects was beyond the scope of this research. In light of the above, it can therefore be seen that unless an optimal match between the true commands given to the robot and those measured, the true performance of this fuzzy controller cannot be appropriately judged.

It might not be immediately clear from the results that the GADF has an affect on the controller; however, the PDOP and number of satellites can change abruptly throughout a test, and if these changes are not accommodated for and the robot proceeds at its current velocity with this deterioration in GPS positional accuracy, then the robot may overshoot the waypoint. This holds especially true if it occurs when the robot is within the vicinity of the waypoints. Another benefit of the GADF is the fact that this variation in speed, even if it only lasts for a short span, can provide the robot with the necessary time to react to this change and to activate or rely on the use of reactive control for instance to ensure that it remains on course.

Given that the robot shows open loop stability, compared to an Ackermann steering where the presence of a disturbance on the front wheels could cause the car to swerve off, the fuzzy controller complements this characteristic of the Spider well. Since inaccuracies in the sensors are present, if the robot was to react to each of those changes it is likely that if a different control strategy had been employed a significant amount of instability in its motion could lead to the malfunctioning of the robot. The robot is still hindered by the presence of fixed velocity commands, as defined by the manufacturers, to ensure the proper handling of its components and drive mechanism. Therefore the presence of a highly sensitive control strategy would not be an appropriate choice unless the current control unit of the robot is replaced with a custom made system that would be able to react accordingly. Finally, it must be mentioned that the use of VB application and the Matlab engine during the experimental tests would inevitably affect the outcome of the path, given that real-time processing would not be possible. Even though the GPS receiver had an update rate of 5Hz, the control system was programmed to update at 2Hz due to the limitations of VB and the Matlab engine. A delay ranging between 300 and 350 ms was determined from the experimental tests. Therefore, it is believed that if the system is programmed to run on real time hardware and software, an improvement in the robots response would be foreseen.

Despite all the factors mentioned, the fuzzy controller successfully reached all of the waypoints and proved to be a good choice for the Spider. Even though the principle of revisiting the control strategy for the angular velocity was brought up, a more suitable solution would be tackling some of these technical issues or developing a learning algorithm (e.g. neuro-fuzzy) that adjusts the fuzzy controller outputs to match the actual.

7.3 Circular Stages of Closeness

The importance of using circular stages of closeness for the research had been shown, given that a model for circular interactions between the GPS and waypoint was needed. The previous discussion revolved around the effect of GPS corrected data, variation in the GADF and the response of the fuzzy control system. For this section, the results will discuss the difference between the use of circular models and a point-to-point interaction. Following this, a brief discussion on the variation of the Spider error radius (RA) is given.

7.3.1 Circular model vs. point

By referring to the left-hand image of Fig. 7.4.2a, it can be seen that the GPS struggles to reach the single point in space. The collective scatter around the point shows the robot in its attempt. Conversely, the right-hand image of Fig.7.4.2a shows the path of the robot having used the circular stages of closeness. Despite the undesirable radius of 10 cm, due to the presence of GPS positional fluctuations, the robot still managed to reach the waypoints, but not always on its first attempt.



Fig.7.4.2a: The path of the robot (yellow) compared to the path of the GPS (white). The left image shows the result of the point-to-point approach and the right-hand image that of circular stages of closeness. The results were shown for the same duration of time.

By altering the condition $(d_p < RB)$ from the circular stages of closeness, or increasing the waypoint radius, the sensitivity of the condition is reduced which allows for the waypoint to be reached at the first attempt. This work shows that, if an image with a higher resolution is obtained, and the waypoint radius (which includes variability and bias error)

was between 1-2 cm, then the condition $(d_p < RB)$ would be difficult to meet, due to fluctuations present in the GPS. Therefore the margin of sensitivity would need to be reduced, implying that some of the conditions by Wuersch, M. *et al* [2005] need to be further re-worked (see Table 5.3.2). Work by Bruch *et al* [2002] claims that the user would be able to determine the path with the same precision as the image resolution. However, this fails to take into account the presence of variability, bias error and the error induced by the orthorectification process.

On the other hand, the fact that the GPS had reached the first three waypoints successfully on the first attempt but not the remaining three shows that either the GPS was accurate to within 10 cm initially and then deteriorated slightly, or that the fuzzy controller margin of sensitivity for θ_{ef} or θ_{eb} was too low and therefore the fuzzy membership would need to be adjusted to ensure that the point is reached. This could come at the cost of the robot having to continuously adjust its angle that could effectively cause a negative impact on the functioning of the robot. It is necessary to keep the balance between operator's instructions and control strategy within a safe margin. Therefore it can be concluded that the circular stages of closeness conditions for small waypoint radii would need to be altered.

Perhaps the choice of a different controller that would accommodate the manufacturer's limitations would enable the GPS to reach the point, however, the resolution of the potentiometer, slack in the drive chain, and the presence of vibrations, all affect the sensory input data, in addition to the GPS positional fluctuations, which in turn reflect on the control system.

Furthermore, the fuzzy controller takes into account fluctuations in the potentiometer and in the GPS positional output and ignores them if they fall within a certain range, so with such resolutions it would be difficult to ensure point matching. In addition, the Spider in itself has several design drawbacks that would prevent such precise navigation. Fig 7.4.2b shows how the fuzzy controller accommodates for the fluctuations in the GPS output and of the potentiometer and leads to a smoother path.

This approach also ensures that the following waypoint is not triggered accidentally since the fluctuation could ensure that the condition for loading the waypoint had been met. Having a looser condition means that the path of the circles would be smoother, ensuring the safe loading of the following waypoint. The schematic below, Fig. 7.4.2b, is an exaggerated view of the possible path for demonstration.



Fig.7.4.2b: Fluctuations of the GPS are accommodated for by the fuzzy controller. The schematic is exaggerated for clarity.

By taking a look at the left-hand image of the simulation results (Fig 7.4.2b), it appears that the robot made no effort to reach the waypoint. This lies in the fact that the simulation's robot has no momentum (i.e. dynamics) taken into account, making it a purely computational attempt to hit the waypoint.



Fig.7.4.2c: The path of the robot (yellow) compared to the path of the GPS (white) of the simulation. The left image shows the result of the point-to-point approach and the right-hand image that of circular stages of closeness.

By referring to the right-hand image of Fig. 7.4.2c, the path appears to be smooth and passes through each waypoint without any hindrance. This draws back the idea that since dynamics were not incorporated, the true response of the robot cannot be judged. By

comparing the simulation results, the advantage of using circular stages of closeness over none is evident.

This is not the first work to employ the principle of adding a proximity area around a waypoint radius for decision-making. Vaneck [1997], Gonzalez *et al* [2004], Maalouf *et al* [2006], and others have created areas in order to ensure that a robot does not overshoot the waypoint, and to enhance the overall accuracy of waypoint-following.

However, the method being adopted here differs because the waypoint radius is being quantified by means of measureable elements (variability and bias of the waypoint selected from the aerial image), and to accommodate for the GPS fluctuation. The effect of varying the radii has also been shown in terms of waypoints hit. Furthermore, the Spider error circle has been created in order to ensure that the robot approaches the waypoint at a suitable velocity.

7.3.2 Varying RA

In section 6.3.1 the results for the seven different cases of circular stages of closeness are shown around waypoint 3. For each of the cases the corresponding simulation is presented. It would be difficult to obtain multiple conclusive answers, since the GPS working conditions were quite similar and therefore the variation of the radius RA is merely a demonstration of the path the circular model takes as it reaches and heads off to the next one. However, it can be noted that in none of the cases shown do the centres of any of the circles reach or meet the centre of the waypoint, but instead appear to brush past them. This in fact shows the benefit of having circular intersection models, which was previously discussed.

It is important to note that changing the radius of RA was performed manually for demonstration purposes, but throughout the research it is dependant on the 95th percentile of the static data collection. During the course of testing the circular stages of closeness do not change, since it is fixed at the beginning of each experiment.

By varying the radius of RA, the velocity outputs of the fuzzy controller also vary, since d_c is dependent on RA and RB. This has shown to be a good approach for small radii, because as the circles meet, the distance d_p governs the speed at which the waypoint is approached, and being quite small a low velocity results. Similarly it has shown benefits for larger radii too, but once the circles meet the distance d_p is still large and therefore the same fuzzy controller commands are activated. A larger benefit could be experienced by adding a third stage fuzzy controller that is triggered once d_p becomes the governing factor.

7.4 The heading control strategy

For this demonstration, one sample of GPS corrected and one of non-corrected is used. This difference does not play a role in this discussion. It was merely chosen to demonstrate the control strategy approach.

In Fig 7.4a, between waypoint 15 and 16 the robot does not appear to have taken the shortest distance path, as previously shown in 5.3.1 (*The heading control strategy*). However, by taking a closer look at the coloured rectangles in the velocity graphs of Fig.7.4b it can be seen that for the left image the robot continued travelling in the forward velocity direction from the previous waypoint, but in the right graph it is clear that the robot adopted the shortest path approach by travelling in the reverse velocity. Typically from the simulation it is expected that the angle between waypoints 15 and 16 should be acute and hence if the robot was travelling in the reverse direction it would move in the forward direction and vice-versa. As previously mentioned, the shortest travelled distance (improved) approach is the desired outcome of the heading control strategy.



Fig.7.4a: The left shows the result from the High GADF using the corrected data, and the right hand side shows the result of the High GADF non-corrected.



Fig.7.4b: Close up of the heading control strategy result, showing that the right hand image has actually taken the shortest distance (improved) approach.

The following scenario is the plausible reason behind this. Since the robot had been travelling in an oscillating way, then it is quite likely that the previous waypoint could have

been overshot and therefore θ_{ef} was smaller than θ_{eb} causing the robot to travel in the forward direction. Nevertheless, both approaches yield the same end result.

7.5 Review of Simulation

The simulation results have shown that in terms of the number of waypoints hit the actual results are significantly higher than the simulation, which implies that the simulation provides an underestimate of the true performance of the system. This suggests that the simulation model needs to be revisited and some changes need to be made. The most likely cause for this could be in the modelling of the GPS error. It is important to recall that the purpose of that function was to introduce an error margin. Given that real GPS simulation software plug-ins can be quite costly the GPS error model created appears to have served its purpose, in the sense that it created this element of uncertainty and unpredictability typically faced with a GPS receiver.

On the other hand, the simulation has shown a clear advantage in terms of the path smoothness compared to the actual. This means that dynamic factors need to be incorporated for future work.

7.6 Future Work

There are several elements that can be improved in this research to enhance the performance of the proposed system:

- The fuzzy controller approach needs to be re-visited. Having the angular velocity dependant on distance to the waypoint in addition to GADF could show improvements in the smoothness of the output.
- 2) The hydraulic system needs to be carefully monitored, as does the battery status, since the hydraulic system has often been creeping under conditions that require it to be travelling at high velocities. The same applies to the angular velocity (steering) that is dependent on the battery charge. Unless these areas are clearly investigated it will be difficult to judge the true performance of the control system.
- 3) Work on improving the simulation results since it has been shown that the simulation results provide an underestimate of the true performance of the actual robot in terms of the waypoints hit. Also, factoring in the dynamic effects will be necessary.
- 4) It has been noted that a further shift exists at the start of each run from waypoint one. If that additional shift is taken into account, an overall improvement in waypoint reaching is also expected. However, predicting how long it will last for is another matter on its own.
- 5) In essence this system would prove to be effective in conjunction with localised reactive obstacle avoidance. Once the robot reaches the general waypoint it would then rely on localised reactive control. This method would prove to be beneficial in large open scale landscapes.

- 6) A future improvement would also be in incorporating 3D terrain models for improved navigation in areas with greater topographical variation.
- Given the variation of the error across the image, it would be necessary to obtain the error distribution of the orthorectified image, in order to vary the waypoint radius across the image.
- A future area of work is determining the effect the distance between the waypoints has on the performance of the controller.

Chapter 8: Conclusion

The aim of this research was the introduction of autonomous mobile robots for agricultural and horticultural tasks, to tackle the issues that arise from hiring migrant labourers and the competition from cheap imported produce. The problems with the automation of these processes have been the costs associated with such equipment making it an unviable solution. Confronting this issue and introducing low-cost, manoeuvrable and robust robots would undoubtedly spur the interest of farmers.

The introduction of Ransomes-Jacobsen's Spider has opened up the potential for such research due to its desirable characteristics. The focussed aim of this research was to undertake work in the area of transport in large-open spaced agricultural areas with the added element of including a human-in-the-loop.

Using aerial images for the selection of waypoints, along with a low-cost GPS receiver, was an objective of this research. This broadened the spectrum of the research objectives by investigating the errors associated with a waypoint selected from an aerial image; extending the positional accuracy of the GPS receiver to avoid the need for expensive receivers; the implementation of a control strategy that would allow it to be used in a wide range of terrain properties and would exploit the full capability of the robot's drive mechanism. Furthermore, another objective was to create a simulation of the system as a means for comparison to the experimental results, and to create an error model for the GPS.

On the whole, the results obtained showed a promising outcome. The human-in-the-loop was met with the choice of selecting waypoints from an aerial image, providing the user with the flexibility of choosing the path to be taken by the robot. The errors associated with the waypoints had been identified. The GPS positional correction method implemented showed a significant improvement to the current WAAS/EGNOS differential signal. In addition the fuzzy controller had successfully navigated through a series of waypoints

using circular stages of closeness. The simulation had shown great similarity to the actual test results even though it often presented an underestimate of the true performance of the robot.

To recapitulate, the challenges and problems faced, the solutions provided and outcomes obtained are revisited once more to demonstrate how the ideas worked together:

Certain fundamental challenges arose in the implementation of GPS and aerial images for mobile robot navigation. First of all, how does one work within the current accuracy of GPS signals for mobile robots, and what is the relationship between GPS-gathered data points and surveyed points on corresponding aerial images? Furthermore, is it possible to extend or optimise the accuracy of these technologies?

The first topic to be approached was the use of aerial imagery, with the forward goal of using images for the selection of waypoints. It was imperative to discover the current, "raw", level of accuracy. In other words, when a waypoint was selected on an aerial image, how closely would it match the ground position?

Investigation showed that small scale inaccuracies arising from the 1936 re-triangulation of the UK can lead to significant positional errors of up to 20m. These inconsistencies would need to be accommodated for if the GPS positional output is to match the image coordinate. Testing revealed that, not only are image-selected points different from actual ground-surveyed points, but the accuracy of one aerial image of a land mass may differ from another image of the same land mass due the triangulation errors. Shifts ranging from 0.087 m - 0.732 m were recorded. These were discovered to be the result of variability (human-induced) and bias error (a result of the orthorectification process).

In order to accommodate for these inaccuracies, an *error circle* was created around each waypoint.

The next challenges to be approached were those relating to the GPS. It was previously determined that accuracies to within 3 m could be achieved, which was not sufficient for the chosen application. In addition, if GPS was to be used in conjunction with aerial images, it was imperative to discover how the GPS coordinates related to the aerial image coordinates.

The default coordinate system used by the GPS receiver is WGS84, whereas the National Grid uses OSGB36. This inconsistency immediately posed a problem for matching single points accurately. Furthermore, despite the fact that it was possible to select OSGB36 on the GPS receiver (which, in theory, would give outputs identical to those on the aerial image), the GPS was really only performing internal transformations from WGS84 using an approximate transformation – and these transformations themselves could contain errors of up to 20 m.

To avoid the errors associated with these internal transformations, it was discovered that choosing the user-defined functionality to determine localised parameters for the test area improved the overall spatial positioning. This showed greater proximity to the actual point than the internal OSGB36 coordinate system.

Nevertheless, the GPS was also found to be inconsistent from one day to the next, and, indeed, throughout the day. Although the relationships between a series of collected points were consistent, the entire set of points was shown in a slightly different position from one day to the next (i.e. the 'cluster' shifted). These inconsistencies can be (partially) explained by the changes in the number of available satellites at any given moment, and by the Dilution of Precision (DOP) of the GPS signals. Planning software (e.g. Trimble) can give the user an idea of how the GPS signals will be at different times of day, and during different times of the year.

These inconsistencies led to the idea of correcting the GPS data prior to any testing. To achieve this, sample data (static) was collected for 15-minute intervals before the test journey began. The average of these points was then compared to a known surveyed point,

and the spatial shift was then determined. By 'shifting' future data by this amount, the GPS signals could still be used, but more accurate results could be obtained. Using this approach, positional improvements of up to 48% were achieved, in relation to the percentage of waypoints reached. The waypoint had a radius of 0.5m and yielded for 17 test runs, a total travelled distance of 1.43 km, 83.6% waypoint hit.

This GPS positional correction gives a more accurate estimate for matching the GPS position to the corresponding aerial image. However, this still does not give a guaranteed true ground position, but merely a better estimate. To accommodate for this error, a GPS error circle was created to represent the area covering the possible locations of the GPS position with respect for the robot. As a result, a Spider error circle was created around the GPS position, which would encompass the robot's centre.

The next step was to determine an appropriate control strategy for having the robot follow waypoints, while taking into account the GPS position fluctuations. The control strategy ideally needed to take advantage of the capabilities offered by a synchronous drive robot, and also needed to be versatile in order to cope with varied environments.

A circular stages of closeness method was used to allow collusion between the waypoint (circle) and the Spider error circle. This method was used to control the speed of the robot as it approached the waypoint, and determined when the next waypoint was to be loaded, based on the proximity of one circle to another.

A fuzzy controller was implemented to accommodate for mechanical and sensor output fluctuations, in addition to the effect of dynamic disturbances which the robot could face in an outdoor environment. The fuzzy controller takes into account the GPS Accuracy Decision Factor (GADF), which is based on the Position Dilution of Precision (PDOP) and the Number of Satellites (NOS) at any given time.

Prior to testing the robot in the real outdoor environment, a simulation was created to mimic its waypoint-reaching capabilities. This simulation included the GPS error model (and the waypoint error model), and provided a platform for testing various control strategies.

Following tests with the Spider in the 'real' outdoor environment, it was shown that at a high GADF (i.e. high number of satellites, low DOP) through 18 waypoints, 100% of the waypoints were hit at a 0.6 m waypoint radius, and 94% of the waypoints were hit within a 0.4 - 0.5 m waypoint radius. Without the correction, only 22% of the waypoints (in a 0.6 m radius) were hit. In summary, results have shown that the GPS can achieve accuracies to within less than 1.3 m – a significant improvement to the WAAS/EGNOS positional estimate to within 3 m.

In addition, a heading control strategy was developed for the synchronous drive platform, which allowed manoeuvres within the shortest possible distance.

Both the simulations and the outdoor testing showed promising results. Using a correction technique, the GPS data was adjusted for real-time implementation for improved positional navigation. This avoided having to purchase DGPS signals, and proved to be a good low-cost alternative. The platform, the Spider, was successfully converted to a waypoint-following robot which, using a 2-stage fuzzy controller, responded to variations in signal strength using a Circular Stages of Closeness model.

This proposed system paves the way for implementing low-cost techniques for transport in large open-space agricultural and horticultural fields.

References

Abbott, E. and D. Powell (1999). Land-vehicle navigation using GPS. Proceedings of the IEEE, 87(1): 145-162. ISSN: 0018-9219

Alvarez, J. C., A. Shkel and A. Lumelsky (1998). Accounting for Mobile Robot Dynamics in Sensor-Based Motion Planning: Experimental Results. International Conference on Robotics & Automation, Proceedings of the 1998 IEEE, Leuven, Belgium. ISBN: 0-7803-4300-X

Analog Devices Inc. (2007). http://www.analog.com/en/

Antonelli, G., S. Chiaverini and G. Fusco (2007). A fuzzy-logic-based approach for mobile robot path tracking. IEEE Transactions on Fuzzy Systems, 15(2): 211-221. ISSN: 1063-6706

Ashkenazi, V., D. Park and M. Dumville (2000). Robot positioning and the global navigation satellite system. Industrial Robot, 27(6): 419-426. ISSN: 0143991X

Axis Communications. (2006). http://www.axis.com

Baerdemaeker, J. D., A. Munack, H. Ramon and H. Speckmann (2001). Mechatronic Systems, Communication, and Control in Precision Agriculture. IEEE Control Systems Magazine, 21(5): 48-70. ISSN: 0272-1708

Beard, R. W. (2003). Motion Planning using Waypoints. Department of Electrical and Computer Engineering, Bringham Young University. – Report
Bechar, A. and Y. Edan (2003). Human-robot collaboration for improved target recognition of agricultural robots. Industrial Robot, 30(5): 432-436. ISSN: 0143-991X

Bih, J. (2006). "Paradigm shift - an introduction to fuzzy logic".(2006).In: IEEE Potentials Magazine.25 (1): 6-21. ISSN: 0278-6648

Booth, D. T., S. E. Cox and R. D. Berryman (2006). Precision measurements from verylarge scale aerial digital imagery. Environmental Monitoring and Assessment, 112(293-307. ISSN: 0167-6369

Borenstein, J., H. R. Everett, L. Feng, S. W. Lee and R. H. Byrne. (1996)."Synchro Drive" In: Where am I? Sensors and Methods for Mobile Robot Positioning. University of Michigan.

Bretz, E. A. (2003). Precision navigation in European skies. IEEE Spectrum, 40(9): 16. ISSN: 0018-9235

Brockett, R. W. (1983). Asymptotic stability and feedback stabilization. Differential geometric control theory, Proceedings of the conference, Michigan Technological University. ISBN/ISSN: 0364330910

Bruch, M. H., G. A. Gilbreath, J.W. Muelhauser and J. Q. Lum (2002). Accurate waypoint navigation using non-differential GPS. AUVSI Unmanned Systems 2002, Lake Buena Vista, Florida, USA.

Bruch, M. H., J. Lum, S. Yee and N. Tran (2005). Advances in autonomy for small UGVs. Unmanned Ground Vehicle Technology VII Orlando, Florida. ISBN: 0277-786X Budkewitsch, P., C. Prevost, R. McGregor and D. Desnoyers (2004). Low cost, real-time GPS positional tracking on digital satellite images improves ground truth productivity. Proceedings IEEE International Geoscience and Remote Sensing Symposium, 2004. IGARSS '04., Anchorage, AK, United States ISBN: 0-7803-8742-2

Cameron, S. (1994). Obstacle Avoidance and Path Planning. Industrial Robot, 21(5): ISSN: 0143-991X

Chatila, R. (1994). Control Architectures for Autonomous Mobile Robots. Proceedings 1994 From Perception to Action Conference., ISBN: 0-8186-6482-7

Chatila R. and J.-P. Laumond (1985). Position referencing and consistent world modeling for mobile robots. In Proceedings of the 1985 IEEE International Conference on Robotics and Automation.

Cho, A., J. Kim, S. Lee and e. al (2007). Fully automatic taxiing, takeoff, and landing of a UAV using a single-antenna GPS receiver only. International Conference on Control, Automation and Systems 2007, Seoul, Korea. ISBN: 978-89-950038-6-2-98560

Chopra, A., M. Obsniuk and M. R. Jenkin (2006). The Nomad 200 and the Nomad SuperScout: Reverse engineered and resurrected. Computer and Robot Vision (CRV'06). Proceedings of the 3rd Canadian Conference Quebec City, Canada. ISBN: 0-7695-2542-3

Clarke, J. and J. Salt (2003). Work permits and foreign labour in the UK: a statistical review. Labour Market Trends, 111(11): 563-574. ISSN: 1361-4819

Cupertino, F., V. Giordano, D. Naso and L. Delfine (2006). "Fuzzy control of a mobile robot".(2006).In: IEEE Robotics and Automation Magazine.13 (4): 74-80. ISSN: 1070-9932

d'Andréa-Novel, B., G. Campion and e. al. (1995). Control of nonholonomic wheeled mobile robots by state feedback linearization. International Journal of Robotics Research, 14(6): 543-559. ISSN: 0278-3649

Devantech Ltd. (2007). http://www.robot-electronics.co.uk/

DePriest, D. (2003)."Chapter 5" In: A GPS User Manual: Working with Garmin Receivers. 1st Books.

Dial, G., H. Bowen, F. Gerlach, J. Grodecki and R. Oleszczuk (2003). IKONOS satellite, imagery, and products. Remote Sensing of Environment, 88(1): 23-36. ISSN: 0034-4257

Ding, L. and A. Goshtasby (2001). On the Canny edge detector. Pattern Recognition, 34(3): 721-725. ISSN: 0051-3203

Doh, N. L., H. Choset and W. K. Chung (2006). Relative localization using path odometry information. Autonomous Robots, 21(2): 143-154. ISSN: 1573-7527

Dudek, G., M. Jenkin, E. Milios and D. Wilkes (1991). Robotic exploration as graph construction. IEEE Transactions on Robotics and Automation, 7(6): 859-865. ISSN: 1042-296X

Dudek, G., M. Jenkin, E. Milios and D. Wilkes (1997). Map validation and robot selflocation in a graph-like world. Robotics and Autonomous Systems, 22(2): 159-178. ISSN: 0921-8890

Dudek, G. and M. Jenkin (2000). Computational Principles of Mobile Robotics. Cambridge, UK, Cambridge University Press. ISBN: 0 521 56876 5 Durrant-Whyte, H. (2005). Autonomous land vehicles. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 219(1): 77 - 98 ISSN: 0959-6518

Earl, R., G. Thomas and B. S. Blackmore (2000). The potential role of GIS in autonomous field operations. Computers and Electronics in Agriculture, 25(1-2): 107–120. ISSN: 0168-1699

Elnagar, A. and L. Lulu (2004). A global path planning Java-based system for autonomous mobile robots. Science of Computer Programming, 53(1): 107-122. ISSN: 0167-6423

Feddema, J., C. Lewis and P. Klarer (1999). Control of multiple robotic sentry vehicles. Proceedings of SPIE - The International Society for Optical Engineering, Orlando, FL, USA. ISBN: 0277-786X

Fox, D., W. Burgard and S. Thrun (1997). The Dynamic Window Approach to Collision Avoidance. IEEE Robotics and Automation Magazine, 4(1): 23-33. ISSN: 1070-9932

Garmin Ltd. (2006). http://www.garmin.com

Fryer, J., H. Mitchell and J. Chandler (2007). "Applications of 3D measurement from Images" In: Whittles Publishing.Caithness, Scotland. ISBN: 978-1870325-69-1

Ge, S. S., X. Lai and A. Al Mamun (2005). Boundary following and globally convergent path planning using instant goals. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 35(2): 240-254. ISSN: 1083-4419

Gonzalez-Villela, V. J., R. M. Parkin, M. Lopez-Parra, J. M. Dorador-Gonzalez and L. Guadarrama (2004). A wheeled mobile robot with obstacle avoidance capability. Ingeniería Mecánica Tecnología y Desarrollo. Revista de la Sociedad Mexicana de Ingeniería Mecánica (SOMIM), 1(5): 159-166. ISSN: 1665-7381

Gopalakrishnan, B., A. Tirunellayi and R. Todkar (2004). Design and development of an autonomous mobile smart vehicle: a mechatronics approach. Mechatronics, 14(5): 491-514. ISSN: 0957-4158

Higuchi, H., M. Saito, T. Iwahashi and S. Usui (2004). Network based high accuracy realtime GPS positioning for GCP correction of high resolution satellite imagery. Proceedings of the 2004 IEEE International Geoscience and Remote Sensing Symposium Anchorage, Alaska, USA. ISBN: 0-7803-8742-2

Hodo, D. W., J. Y. Hung, D. M. Bevly and D. S. Millhouse (2007). Analysis of Trailer Position Error in an Autonomous Robot-Trailer System with Sensor Noise. 2007 IEEE international Symposium on Industrial Electronics, Vigo, Spain. ISBN: 1-4244-0755-9

Hokayem, P. F. and M. W. Spong (2006). Bilateral teleoperation: An historical survey. Automatica, 42(12): 2035 – 2057. ISSN: 0005-1098

Holden, M. E. (2004). Low-cost autonomous vehicles using just GPS Waypoint navigation. Proceedings of the 2004 ASEE Conference, Salt Lake City, Utah.

Hollingum, J. (1997). Fresh start sought for British robotics. Industrial Robot, 24(3): 216-218. ISSN: 0143-991X

Hollingum, J. (1999). Robots in agriculture. Industrial Robot, 26(6): 438-445. ISSN: 0143-991X

HortiBot. (2007). http://www.hortibot.dk

Howard, A., H. Seraji and B. Werger (2005). Global and Regional Path Planners for Integrated Planning and Navigation. Journal of Robotic Systems, 22(12): 767-778. ISSN: 0741-2223 Hwang, Y. K. and N. Ahuja (1992). A potential field approach to path planning. IEEE Transactions on Robotics and Automation, 8(1): 23-32. ISSN: 1042-296X

IENSYS Inc. (2006). http://www.iensys.com

IFR and UNECE (2005). World 2005 Robotics. Statistics, Market Analysis, Forecasts, Case Studies, and Profitibility of Robot Investment, ISSN: 1020-1076

Jensen, B., N. Tomatis, L. Mayor, A. Drygajlo and R. Siegwart (2005). Robots Meet Humans - Interaction in Public Spaces. IEEE Transactions on Industrial Electronics, 52(6): 1530-1546. ISSN: 0278-0046

Kassler, M. (2001). Agricultural automation in the new millennium. Computers and Electronics in Agriculture 30(1-3): 237-240. ISSN: 0168-1699

Kelly, A., A. Stentz, A. Omead, M. Bode, D. Bradley and A. Diaz-Calderon (2006). Toward reliable off road autonomous vehicles operating in challenging environments. The International Journal of Robotics Research, 25(5-6): 449-482. ISSN: 0278-3649

Kidono, K., J. Miura and Y. Shirai (2002). Autonomous visual navigation of a mobile robot using a human-guided experience. Robotics and Autonomous Systems, 40(2-3): 121-131. ISSN: 0921-8890

Kim, S. and J.-H. Kim (2001). Adaptive fuzzy-network-based C-measure map-matching algorithm for car navigation system. IEEE Transactions on Industrial Electronics, 48(2): 432-441. ISSN: 0278-0046

Komi, P., M. R. Jackson and R. M. Parkin (2007). Plant classification combining colour and spectral cameras for weed control purposes. Industrial Electronics, 2007 IEEE International Symposium on, Vigo, Spain. ISBN: 1-4244-0755-9 Kotani, S., K. Kaneko, T. Shinoda and H. Mori (1998). Mobile Robot Navigation Based on Vision and DGPS Information. Proceedings of the 1998 IEEE International Conference on Robotics & Automation. Part 3., Leuven, Belgium. ISBN: 0-7803-4300-X

Kumon, M., Y. Udo, H. Michihira, M. Nagata, I. Mizumoto and Z. Iwai (2006). Autopilot system for kiteplane. IEEE/ASME Transactions on Mechatronics, 11(5): 615-625.

Lee, K. W., W. S. Wijesoma and J. Ibanez-Guzman (2004). Map Aided SLAM in Neighbourhood Environments. 2004 Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy ISBN: 0-7803-8310-9

Leedy, B. M. and C. Reinholtz (2006). Facing the challenge [autonomous ground vehicle]. Computing & Control Engineering Journal 17(5): 42 - 43 ISSN: 0956-3385

Lenain, R., B. Thuilot, C. Cariou and P. Martinet (2006). High accuracy path tracking for vehicles in presence of sliding: Application to farm vehicle automatic guidance for agricultural tasks Autonomous Robots, 21(1): 79-97. ISSN: 1573-7527 (Online)

Lever, J. H., A. D. Streeter and L. E. Ray (2006). Performance of a solar-powered robot for polar instrument networks. Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Orlando, Florida. ISBN: 0-7803-9505-0

Lindsay, C. (2003). A century of labour market change: 1900 to 2000. Labour Market Trends, 111(13): 133-144. ISSN: 1361-4819

Maalouf, E., M. Saad and H. Saliah (2005). A higher level path tracking controller for a four-wheel differentially steered mobile robot. Robotics and Autonomous Systems, 54(1): 23-33. ISSN: 0921-8890

Martinelli, A. (2002). The Odometry Error of a Mobile Robot With a Synchronous Drive System. IEEE Transactions on Robotics and Automation, 18(3): 399-405. ISSN: 1042-296X

Meguro, J.-i., K. Ishikawa, Y. Amano, T. Hashizume, J.-i. Takiguchi, R. Kurosaki and M. Hatayama (2005). Creating Spatial Temporal Database by Autonomous Mobile Surveillance System (A Study of Mobile Robot Surveillance System using Spatial Temporal GIS Part 1). Proceedings of the 2005 IEEE International Workshop on Safety, Security and Rescue Robotics, Kobe, Japan. ISBN: 0-7803-8945-x

Michalski, A. and J. Czajewski (2004). The accuracy of the global positioning systems. IEEE Instrumentation and Measurement Magazine, 7(1): 56-60. ISSN: 1094-6969

Microchip Technology Inc. (2006). http://www.microchip.com

MikroElektronika. (2006). http://www.mikroelektronika.co.yu/

MobileRobots Inc. (2007). "THE INDOOR/OUTDOOR ALL-WEATHER ROBOT: SeekurTM." http://www.activrobots.com/ROBOTS/Seekur.html

Moore, P. and P. Crossley (1999). GPS applications in power systems: Introduction to GPS. Power Engineering Journal, 13(1): 33-9. ISSN: 0950-3366

Mulvaney, D., Y. Wang and I. Sillitoe (2006). Waypoint-based mobile robot navigation. The 6th World Congress on Intelligent Control, Dalian, China. ISBN: 1-4244-0332

Mumby, P. J., E. P. Green, A. J. Edwards and C. D. Clark (1997). Coral reef habitat mapping: how much detail can remote sensing provide? Marine Biology, 130(2): 193-202. ISSN: 1432-1793

Muscato, G., D. Caltabiano, S. Guccione, D. Longo, M. Coltelli, A. Cristaldi, E. Pecora, V. Sacco, P. Sim, G. S. Virk, P. Briole, A. Semerano and T. White (2003). ROBOVOLC: a robot for volcano exploration result of first test campaign. Industrial Robot, 30(3): 231-242. ISSN: 0143-991X

Nagasaka, Y., N. Umeda, Y. Kanetai, K. Taniwaki and Y. Sasaki (2004). Autonomous guidance for rice transplanting using global positioning and gyroscopes Computers and Electronics in Agriculture, 43(3): 223-234. ISSN: 0168-1699

NASA. (2006). "Robot Revolt." http://media.nasaexplores.com/lessons/03-024/5-8_2.pdf

Nemenyi, M., P. A. Mesterhazi, Z. Pecze and Z. Stepan (2003). The role of GIS and GPS in precision farming. Computers and Electronics in Agriculture, 40(1-3): 45-55. ISSN: 0168-1699

Ochieng, W. Y. and K. Sauer (2002). Urban road transport navigation: performance of the global positioning system after selective availability. Transportation Research Part C, 10(3): 171-187. ISSN: 0968-090X

Oh, P. Y. and W. E. Green (2004). Mechatronic kite and camera rig to rapidly acquire, Process, and distribute aerial images. IEEE/ASME Transactions on Mechatronics 9(4): 671 - 678. 1083-4435

O'Keefe, K. O., O. Julien, M. E. Cannon and G. Lachapelle (2006). Availability, accuracy, reliability, and carrier-phase ambiguity resolution with Galileo and GPS. Acta Astronautica, 58(8): 422-434. ISSN: 0094-5765

Ollero, A., B. C. Arrue, J. Ferruz, G. Heredia, F. Cuesta, F. López-Pichaco and C. Nogales (1999). Control and Perception Components for Autonomous Vehicle Guidance. Application to the ROMEO vehicles. Control Engineering Practice, 7(10): 1291-1299. ISSN: 0967-0661

OS. (2007). http://gps.ordnancesurvey.co.uk/convert.asp.

Ordnance Survey.(2007). A guide to coordinate systems in Great Britain. Southampton, UK. Panzieri, S., F. Pascucci and G. Ulivi (2002). An outdoor navigation system using GPS and inertial platform. IEEE/ASME Transactions on Mechatronics, 7(2): 134-142. ISSN: 1083-4435

Parasuraman, R., S. Galster, P. Squire, H. Furukawa and C. Miller (2005). A Flexible Delegation-Type Interface Enhances System Performance in Human Supervision of Multiple Robots: Empirical Studies With RoboFlag. IEEE Transactions on Systems, Man and Cybernetics, Part A., 35(4): 481-493. ISSN: 1083-4427

Ransomes-Jacobson. (2005). "PRODUCT SPECIFICATION SHEET: Spider - ILD01 Product Code."

http://www.ransomesjacobsen.com/specifications/b_municipal/specification_spider_ild01. pdf

Ray, L. E., J. H. Lever, A. D. Streeter and A. D. Price (2007). Design and power management of a solar-powered "cool robot" for polar instrument networks. Journal of Field Robotics, 24(7): 581-599. ISSN: 1556-4959

Ren, W., J.-S. Sun, R. Beard and T. McLain (2007). Experimental validation of an autonomous control system on a mobile robot platform. IET Control Theory Applications, 1(6): 1621-1629. ISSN: 1751-8644

Rogers, S. (1-3 October 2000). Creating and Evaluating Highly Accurate Maps with Probe Vehicles. IEEE Intelligent Transportation Systems, Dearborn (MI) USA. ISBN: 0-7803-5971

Romans, W., B. Poore and J. Mutziger (2000). Advanced instrumentation for agricultural equipment. IEEE Instrumentation and Measurement Magazine, 3(1): 26-29. ISSN: 1094-6969

Šafarič, R., M. Debevc, R. M. Parkin, and S. Uran (2001). Telerobotics experiments via internet. IEEE Transactions on Industrial Electronics, 48(2): 424-431. ISSN: 0278-0046

Satirapod, C. and P. Chalermwattanachai (December 6-8 2004). Impact of different tropospheric models on GPS baseline accuracy: Case study in Thailand. The 2004 International Symposium on GNSS/GPS, Sidney, Australia.

Satirapod, C., I. Trisirisatayawong and P. Homniam (2003). Establishing Ground Control Points for High-Resolution Satellite Imagery Using GPS Precise Point Positioning. 2003 Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Toulouse, France. ISBN: 0-7803-7929-2

Sensors & Satellites. "Earth Observation Sensors: Higher resolutions and shorter revisit times".(2007).In: Engineering Surveying Showcase 2007.(2): 19-25. ISBN: 0 94677974 0

Sethuramasamyraja, B. S. (2003). GPS based waypoint navigation for an autonomous guided vehicle - Bearcat III.Department of Mechanical, Industrial and Nuclear Engineering of the College of Engineering. University of Cincinnati. Master of Science

Shair, S., R. M. Parkin and M. R. Jackson (2006). A Robust Mobile Platform for Agricultural/Horticultural Applications CLAWAR 2006, 9th International Conference on Climbing and Walking Robots, Brussels, Belgium.

Sheridan, T. B. (1989). Telerobotics. Automatica, 25(4): 487-507. ISSN: 0005-1098

Shiller, Z. and Y.-R. Gwo (1991). Dynamic Motion Planning of Autonomous Vehicles. IEEE Transactions on Robotics and Automation, 7(1): 241-249. ISSN: 1042-296X Siegwart, R. and I. R. Nourbakhsh (2004). Introduction to Autonomous Mobile Robots. Cambridge, MA, USA, The MIT Press. ISBN: 0-262-19502-X

Sigrimis, N., P. Antsaklis and P. P. Groumpos (2001). Advances in control of agriculture and the environment. IEEE Control Systems Magazine 21(5): 8-12. ISSN: 0272-1708

Silver, D., B. Sofman, N. Vandapel, J. A. Bagnell and A. Stentz (2006). Experimental Analysis of Overhead Data Processing To Support Long Range Navigation. International Conference on Intelligent Robots and Systems, Proceedings of the 2006 IEEE/RSJ, Beijing, China. ISBN: 1-4244-0259

Sim, D., S.-Y. Jeong, D.-H. Lee, R.-H. Park, R.-C. Kim, S. U. Lee and I. C. Kim (1999). Hybrid estimation of navigation parameters from aerial image sequence. IEEE Transactions on Image Processing, 8(3): 429-435. ISSN: 1057-7149

Sim, P., V. Sacco, G. S. Virk and X. W. (2003). Robot navigation in volcanic environments. Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE, Shanghai, China. ISBN: 0-7803-8125-4

Sim, P., V. Sacco and G. S. Virk (2004). The user interface system for the Robovolc exploration robot. Industrial Robot, 31(2): 189-200. ISSN: 0143-991X

Snider, J. M., A. E. Snider and R. N. Riggins (2004). Vasilius: the design of an autonomous ground robotic vehicle. Journal of Robotic Systems, 21(9): 481-492. ISSN: 0741-2223

Sofman, B., E. Lin, J. Bagnell, N. Vandapel and A. Stentz (2006). Improving Robot Navigation Through Self-Supervised Online Learning. Proceedings of Robotics: Science and Systems, Philadelphia, Pennsylvania. ISBN: 0-262-69348-8

Stentz, A., C. Dima, C. Wellington, H. Herman and D. Stager (2002). A System for Semi-Autonomous Tractor Operations. Autonomous Robots, 13(1): 87-104. ISSN: 0929-5593

Thrun, S. (2002). "Robotic mapping: A survey" In: Exploring Artificial Intelligence in the New Millenium. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. ISBN: 1-55860-811-7

Thrun, S., M. Montemerlo, H. Dahlkamp and e. al (2006). Stanley: the robot that won the DARPA Grand Challenge. Journal of Field Robotics, 23(9): 661-692. ISSN: 0741-2223

Torii, T. (2000). Research in autonomous agriculture vehicles in Japan. Computers and Electronics in Agriculture, 25(1-2): 133-153. ISSN: 0168-1699

Trimble. (2007). http://www.trimble.com

Trisirisatayawong, I., T. Jongrugenun, B. Phalakarn, C. Satirapod and C. Fraser (2004). Enhancing The Prospects for Mapping From High-Resolution Satellite Imagery In the Developing World. International Society for Photogrammetry and Remote Sensing, Istanbul, Turkey.

Tsourveloudis, N. C., K. P. Valavanis and T. Hebert (2001). Autonomous Vehicle Navigation Utilizing Electrostatic Potential Fields and Fuzzy Logic. Robotics and Automation, IEEE Transactions On 17(4): 490-497. ISSN: 1042–296X

Vandapel, N., R. Donamukkala and M. Hebert (2003). Experimental Results in Using Aerial LADAR Data for Mobile Robot Navigation. Field and Service Robotics, The 4th International Conference, Fuji, Japan.

Vaneck, T. W. (1997). Fuzzy Guidance Controller for an Autonomous Boat. IEEE Control Systems Magazine, 17(2): 43-51. ISSN: 0272-1708 Wang, J., S. Schroedl, K. Mezger, R. Ortloff, A. Joos and T. Passegger (2005). Lane keeping based on location technology. IEEE Transactions on Intelligent Transportation Systems, 6(3): 351-356. ISSN: 1524-9050

Wei, Y., Z. Zhao and J. Song (2004). Urban building extraction from high-resolution satellite panchromatic image using clustering and edge detection. 2004 IEEE International Geoscience and Remote Sensing Symposium Proceedings: Science for Society: Exploring and Managing a Changing Planet. IGARSS 2004, Anchorage, AK, United States 0-7803-8742-2

Witte, T. H. and A. M. Wilson (2005). Accuracy of WAAS-enabled GPS for the determination of position and speed over ground. Journal of Biomechanics, 38(8): 1717-1722. ISSN: 0021-9290

WMG.(2007). http://www2.warwick.ac.uk/fac/sci/wimrc/projects/student/ agricultural_robot

Wolf, P. R. (1983). "Elements of Photogrammetry" In: McGraw-Hill Companies. Singapore.

Wu, J., D. Wang and M. A. Bauer (1999). Image-based atmospheric correction of QuickBird imagery of Minnesota cropland. Remote Sensing of Environment, 99(3): 315-325. ISSN: 0034-4257

Wuersch, M. and D. Caduff (2005). "Refined Route Instructions Using Topological Stages of Closeness" In: Lecture Notes in Computer Science. Berlin/Heidelberg, Germany, SpringerLink. ISBN: 978-3-540-30848-5

WWF-UK. (2006). "WWF Scotland: Rural Development: Lettuce." http://www.wwf.org.uk/core/about/scotland/sc_0000001315.asp Zein-Sabatto, S., O. Taiwo and P. Koseeyaorn (2004). Integrated Simulation Software for Outdoor Robots Planning and Coordination. 2004 Proceedings of IEEE SoutheastCon, Greensboro, North Carolina. ISBN: 0-7803-8368-0

Zhou, G., W. Chen, J. A. Kelmelis and D. Zhang (2005). A comprehensive study on urban true orthorectification. IEEE Transactions on Geoscience and Remote Sensing, 43(9): 2138-2147. ISSN: 0196-2892

Židek, K., T. Saloky and I. Polanecká (2006). Usability of GPS Systems for Mobile Robots Navigation. 4th Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence SAMI 2006, Herlany, Slovakia

Publications

- S. Shair, J. H. Chandler, V. J. Gonzalez-Villela, R.M. Parkin and M.R. Jackson. The Use of Aerial Images and GPS for Mobile Robot Waypoint Navigation. IEEE Transactions on Mechatronics. 13 (6): 692-699. ISSN: 1083-4435. (See Appendix C)
- Shair, S., R. M. Parkin and M. R. Jackson (2006). A Robust Mobile Platform for Agricultural/Horticultural Applications CLAWAR 2006, 9th International Conference on Climbing and Walking Robots, Brussels, Belgium
- Shair, S., R. M. Parkin and M. R. Jackson (2006). A Robust Mobile Platform for Autonomous and Teleoperative Navigation. 10th International Conference on Mechatronics Technology (ICMT 2006). Mexico City, Mexico.

Appendix A

(A) Technical specifications

This section touches upon some other technical specifications and work conducted on the Spider that is not directly related to the research in this thesis, yet deserves mentioning nonetheless.



Fig. A0: The Spider with its hardware

Hardware Architecture

The schematic in Fig. A1 shows the additional proposed hardware that has been set up (but not used for this research) on the lower platform area, and Fig. A2 shows the additional (unused) hardware on the rotating platform. A description follows:





Fig. A2: Upper platform hardware

Rotating Platform

March 2007 - August 2007

This was part of an MSc project that involved the development of a closed loop PID controller for the rotary platform to ensure that it would consistently maintain the same angular velocity and direction as the steering of the wheels. Fig. A3 is a schematic showing the overall layout of the proposed architecture.



Fig. A3: Schematic of the hardware used for controlling the position of the rotating platform

Data Handling Equipment

April 2006 – September 2006

As part of an internship program, a final year German Diplom student worked on developing the server operating system in the C programming language to handle data from the IRSYS infrared camera to a Visual Basic application on the host PC. A description of the server and the infrared camera follow in their corresponding sections.

Axis Servers (Distributed Network)

The AXIS 82 board is a 100MIPS Linux-operated network server (see Fig. A4). The user has the option of accessing two RS-232 ports and one RS-485 port, one USB slot, two Ethernet ports, and general I/O ports [Axis Communications 2006]. This server has been

implemented and will be used for the Spider's on-board distributed network of processors in future work. The main working program is written in the C programming language.



Fig. A4: AXIS server: a) front view and b) back view

Wireless communication

September 2006 - January 2007

A final year French undergraduate student worked on setting up a three-way wireless communication network between the Spider and host PC, modifying the Linux server for the integration of other components, and enhancing the Visual Application program of the IRSYS infrared camera.

In order to realise the communication handling with an off-board PC, the Spider is equipped with two wireless routers. Since the communication handling between the upper and lower sections of the platform are hindered by the design of the rotating platform, having two separate servers and routers in each area ultimately solves this problem. Fig. A5 shows images of the routers.



Fig. A5: Wireless routers: a) for the lower platform and b) for the upper platform

Sensors

Infrared Camera

For future research activity involving obstacle avoidance, human tracing, or other applications, a passive infrared sensor was mounted on the robot. The IRISYS Thermal Imager (IRI 1002 Multipoint radiometer) provides real time temperature monitoring of 256 data points (see Fig. A6).



Fig. A6: IRISYS thermal imager

The detector used in the IRI 1002 Multipoint radiometer is an IRISYS proprietary pyroelectric array consisting of 16 x 16 pixels. Data is handled by means of the serial communication protocol (RS232). The IR camera provides a temperature range between 23° C to 157° C.

Axis Network Camera

The AXIS network camera (Fig. A7) is typically used for IP-surveillance and for remote monitoring applications. The camera acts like an AXIS 82 Device Server, where the RS232 and I/O ports can be used for other applications and accessed through the network [Axis Communications 2006]. The camera will be used to provide a video-link for teleoperation or for vision guidance.



Fig. A7: Axis network camera

Fig. A8 shows a pan and tilt unit with the IRYSIS infrared camera mounted on the AXIS network camera. These units are fixed to the rotating platform.



Fig. A8: The IRYSIS infrared and AXIS cameras mounted on the Spider

October 2006 – June 2007

An undergraduate BEng student worked on the integration of a compass, tilt sensor and fuel gauge as part of the Spider's instrumentation. They were wirelessly linked through the previously developed distributed network to a host PC for data processing.

Compass (Slipping)

The CMPS03 from Devantech has an accuracy between 3 $^{\circ}$ and 4 $^{\circ}$ with a resolution of 0.1 $^{\circ}$. It has been specifically designed for mobile robot applications (see Fig. A9). It interfaces with the host microcontroller using the I²C protocol [Devantech Ltd. 2007].



Fig. A9: The CMPS03 compass by Devantech. [Devantech Ltd. 2007]

Pan and Tilt Module

A two axis accelerometer MEMS pan and tilt module from Analog Devices, ADXL203, was used [Analog Devices Inc. 2007]. It is capable of measuring between -45 ° and 45 °, with a sensitivity accuracy of \pm 4% (see Fig. A10).



Fig. A10: Dual-Axis MEMS pan and tilt sensor ADXL203 [Analog Devices Inc. 2007]

Fuel Gauge

A standard off-the-shelf fuel gauge sensor is used, which changes its resistance depending on the level of fuel remaining in the tank. The sensor has 10 fixed outputs associated with it. The sensor works based on resistance between the magnet attached to the float and the rod which operates a reed switch (see Fig. A11).



Fig. A11: A fuel gauge with 10 discrete outputs

User Interface

Fig. A12 shows a screenshot of the teleoperated control interface for joystick control of the robot.



Fig. A12: A screenshot of the interface program. Clockwise from top left: AXIS camera, joystick position, GPS data output and the IRYSIS infrared camera.

Appendix B



Fig.B1: First stage fuzzy controller



Fig.B2: The first stage fuzzy controller layout - Matlab



Fig.B3: The fuzzy rules for the primary fuzzy stage - Matlab

Inputs:



Fig.B4: The input membership functions for the PDOP - Matlab



Fig.B5: The input membership functions for the number of satellites - Matlab





Fig.B6: The output membership functions of the GADF - Matlab



Fig.B7: Second stage fuzzy controller



Fig.B8: Second stage fuzzy controller layout - Matlab



Fig.B9: The fuzzy rules for the secondary fuzzy stage - Matlab





Fig.B10: The input membership functions for the Theta error - Matlab



Fig.B11: The input membership functions for the GADF - Matlab



Fig.B12: The input membership functions for the dc-Matlab

Outputs:



Fig.B13: The output membership functions for the angular velocity (w) - Matlab



Fig.B14: The output membership functions for the linear velocity (v) – Matlab



Fig.B15: Main model of the Spider's simulation - Simulink


Fig.B16: First sub-model of the Spider's simulation - Simulink



Fig.B17: Second sub-model of the Spider's simulation - Simulink



Fig.B18: Third sub-model of the Spider's simulation - Simulink

Appendix C

29/10/2007

Before correction:

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	5.263158	5.263158	5.263158	26.31579	36.84210526	42.10526	52.63158	63.15789
2	5.263158	5.263158	5.263158	36.84211	47.36842105	57.89474	63.15789	63.15789
3	0	0	15.78947	31.57895	52.63157895	57.89474	73.68421	78.94737
4	5.263158	5.263158	5.263158	15.78947	31.57894737	31.57895	42.10526	52.63158
5	0	0	0	68.42105	84.21052632	84.21053	84.21053	84.21053
6	0	10.52632	21.05263	36.84211	42.10526316	42.10526	47.36842	63.15789
7	5.263158	5.263158	15.78947	26.31579	31.57894737	36.84211	47.36842	52.63158
8	57.89474	78.94737	78.94737	84.21053	84.21052632	84.21053	84.21053	89.47368
9	68.42105	68.42105	84.21053	84.21053	84.21052632	84.21053	84.21053	84.21053
10	5.263158	5.263158	5.263158	10.52632	15.78947368	36.84211	42.10526	52.63158
Average	15.26316	18.42105	23.68421	42.10526	51.05263158	55.78947	62.10526	68.42105

Waypoint radius (m) - Percentage hit

After Correction:

Waypoint radius (m) - Percentage hit

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	5.263158	10.52632	26.31579	36.84211	42.10526316	63.15789	68.42105	78.94737
2	5.263158	31.57895	47.36842	57.89474	63.15789474	63.15789	68.42105	89.47368
3	10.52632	15.78947	42.10526	57.89474	68.42105263	89.47368	89.47368	94.73684
4	5.263158	15.78947	15.78947	26.31579	47.36842105	52.63158	78.94737	89.47368
5	5.263158	57.89474	73.68421	84.21053	89.47368421	94.73684	100	100
6	15.78947	31.57895	42.10526	42.10526	47.36842105	47.36842	68.42105	68.42105
7	15.78947	21.05263	31.57895	36.84211	42.10526316	52.63158	73.68421	89.47368
8	73.68421	78.94737	84.21053	94.73684	100	100	100	100
9	84.21053	84.21053	100	100	100	100	100	100
10	5.263158	5.263158	5.263158	26.31579	36.84210526	42.10526	57.89474	68.42105
Average	22.63158	35.26316	46.84211	56.31579	63.68421053	70.52632	80.52632	87.89474

30/10/2007

Before correction:

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	78.9474	84.2105	84.2105	89.4737	94.7368	94.7368	94.7368	94.7368
2	84.2105	89.4737	89.4737	89.4737	89.4737	89.4737	94.7368	94.7368
3	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737	89.4737	89.4737
4	15.7895	21.0526	36.8421	36.8421	78.9474	84.2105	84.2105	84.2105
5	26.3158	42.1053	63.1579	68.4211	73.6842	84.2105	84.2105	84.2105
6	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737
7	78.9474	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105
8	5.2632	15.7895	31.5789	52.6316	78.9474	78.9474	84.2105	84.2105
9	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105
10	52.6316	63.1579	73.6842	84.2105	84.2105	84.2105	84.2105	89.4737
11	31.5789	31.5789	42.1053	63.1579	68.4211	73.6842	78.9474	78.9474
12	21.0526	31.5789	47.3684	68.4211	78.9474	84.2105	84.2105	89.4737
13	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737
14	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105
15	52.6316	68.4211	68.4211	78.9474	84.2105	84.2105	89.4737	89.4737
16	47.3684	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737
17	57.8947	68.4211	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105
Average	57.27554	65.01547	71.20742	76,78018	82.66252941	84.52011	85.75849	87.30649

Waypoint radius (m) - Percentage hit

After Correction:

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	36.84211	73.68421	84.21053	89.47368	89.47368421	94.73684	100	100
2	57.89474	94.73684	94.73684	94.73684	100	100	100	100
3	94.73684	100	100	100	100	100	100	100
4	0	5.263158	26.31579	42.10526	52.63157895	63.15789	100	100
5	21.05263	21.05263	31.57895	57.89474	73.68421053	84.21053	100	100
6	68.42105	84.21053	89.47368	94.73684	94.73684211	94.73684	94.73684	100
7	10.52632	73.68421	89.47368	94.73684	100	100	100	100
8	0	5.263158	15.78947	31.57895	42.10526316	78.94737	94.73684	94.73684
9	73.68421	84.21053	89.47368	100	100	100	100	100
10	68.42105	89.47368	89.47368	89.47368	89.47368421	94.73684	100	100
11	36.84211	52.63158	57.89474	68.42105	89.47368421	100	100	100
12	10.52632	21.05263	31.57895	36.84211	63.15789474	78.94737	94.73684	100
13	78.94737	94.73684	94.73684	100	100	100	100	100
14	78.94737	84.21053	89.47368	94.73684	94.73684211	100	100	100
15	68.42105	84.21053	89.47368	94.73684	94.73684211	94.73684	94.73684	94.73684
16	36.84211	47.36842	73.68421	94.73684	94.73684211	100	100	100
17	68.42105	68.42105	73.68421	84.21053	89.47368421	100	100	100
Average	47 67802	63 77709	71 82663	80 49536	86 37770898	93 18885	98 76161	99 3808

Waypoint radius (m) - Percentage hit

4/11/2007

Before correction:

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	0	10.5263	15.7895	31.5789	78.9474	78.9474	89.4737	89.4737
2	26.3158	47.3684	57.8947	73.6842	84.2105	89.4737	89.4737	89.4737
3	42.1053	52.6316	63.1579	68.4211	73.6842	73.6842	84.2105	89.4737
4	0	5.2632	31.5789	89.4737	89.4737	89.4737	89.4737	89.4737
5	15.7895	47.3684	73.6842	78.9474	84.2105	89.4737	89.4737	89.4737
6	0	0	21.0526	47.3684	68.4211	89.4737	89.4737	89.4737
7	0	0	5.2632	15.7895	47.3684	84.2105	89.4737	89.4737
8	5.2632	5.2632	5.2632	21.0526	36.8421	42.1053	52.6316	63.1579
9	5.2632	5.2632	5.2632	5.2632	10.5263	31.5789	73.6842	84.2105
10	47.3684	63.1579	78.9474	78.9474	89.4737	89.4737	89.4737	89.4737
11	78.9474	84.2105	84.2105	89.4737	89.4737	89.4737	89.4737	89.4737
12	15.7895	36.8421	42.1053	68.4211	89.4737	89.4737	89.4737	89.4737
13	78.9474	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737	89.4737
14	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737
15	0	0	0	10.5263	47.3684	73.6842	89.4737	89.4737
16	0	0	0	10.5263	21.0526	42.1053	52.6316	57.8947
17	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	84.2105	89.4737
Average	28.48298	35.91331	43.34365	55.41796	68.42104706	76.78019	83.28174	85.75852

Waypoint radius (m) - Percentage hit

After Correction:

Waypoint radius (m) - Percentage hit

Runs	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1
1	84.2105	89.4737	100	100	100	100	100	100
2	100	100	100	100	100	100	100	100
3	84.2105	89.4737	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100
5	89.4737	100	100	100	100	100	100	100
6	94.7368	100	100	100	100	100	100	100
7	68.4211	100	100	100	100	100	100	100
8	42.1053	57.8947	68.4211	73.6842	94.7368	100	100	100
9	21.0526	73.6842	89.4737	100	100	100	100	100
10	94.7368	94.7368	100	100	100	100	100	100
11	84.2105	89.4737	89.4737	100	100	100	100	100
12	94.7368	100	100	100	100	100	100	100
13	57.8947	73.6842	73.6842	94.7368	100	100	100	100
14	36.8421	47.3684	52.6316	84.2105	100	100	100	100
15	78.9474	89.4737	89.4737	94.7368	94.7368	94.7368	94.7368	100
16	31.5789	47.3684	52.6316	78.9474	100	100	100	100
17	52.6316	68.4211	78.9474	89.4737	94.7368	94.7368	100	100
Average	71,51702	83,59133	87.92571	95.04644	99.0712	99.3808	99.6904	100

The Use of Aerial Images and GPS for Mobile Robot Waypoint Navigation

S. Shair, J. H. Chandler, V. J. González-Villela, Member, IEEE, R. M. Parkin, Member, IEEE, and M. R. Jackson

Abstract-The application of aerial and satellite imagery for mobile robot path planning and navigation has shown potential in recent years. Their uses vary from identifying terrain properties for creating traversability maps to extracting landmarks for autonomous navigation. With the freely available differential positioning system, Wide Area Augmentation System (WAAS)/ European Geostationary Navigation Overlay Service (EGNOS), the use of the GPS with aerial images providing valuable contextual data demonstrates potential in waypoint-based navigation of mobile robots. However, important issues relating to the spatial accuracies of image, waypoint, and GPS-derived data, vital for obtaining accurate navigation results, are often overseen. This paper defines the causes of spatial inaccuracies in order to develop optimal waypoint navigation parameters and provides researchers with sufficient knowledge to reproduce similar results. An improvement of up to 48% in the number of waypoints reached, depending on the radius, was determined for the positional correction of the GPS. The results are shown with a simulated synchronous drive robot in Matlab's Simulink environment. The reader is presented with a method for easily creating waypoints from aerial images, yielding results to a similar level of accuracy to conventional and often tedious manual methods.

Index Terms—Aerial imagery, European geostationary navigation overlay service (EGNOS), GPS, map datum, mobile robot, orthorectification, overhead images, synchronous drive robot, waypoints, wide area augmentation system (WAAS).

I. INTRODUCTION

T HE USE OF aerial and satellite imagery for agricultural and horticultural applications such as precision farming and for long-range autonomous terrain navigation has been a strong motivation behind research conducted at Loughborough University [1], in which a grass-cutting mower, the Ransomes Spider, has been refitted for autonomous navigation (Fig. 1). In recent years, there has been growing interest in imagery (aerial, satellite, laser detection and ranging (LADAR)/light detection and ranging (LIDAR), digital elevation model (DEM),

Manuscript received January 29, 2008; revised July 16, 2008. Current version published December 17, 2008. Recommended by Technical Editor H. Hashimoto. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC), UK, and in part by the Innovative Manufacturing and Construction Research Centre (IMCRC), Loughborough University, UK.

S. Shair, R. M. Parkin, and M. R. Jackson are with the Mechatronics Research Centre, Loughborough University, Loughborough LE11 3UZ, UK (e-mail: s.shair@lboro.ac.uk; r.m.parkin@lboro.ac.uk; m.r.jackson@lboro.ac. uk).

J. H. Chandler is with the Department of Civil Engineering, Loughborough University, Loughborough LE11 3TU, UK (e-mail: j.h.chandler@lboro.ac.uk).

V. J. González-Villela is with the Department of Mechatronics Engineering, National Autonomous University of Mexico (UNAM), Mexico City 04510, Mexico (e-mail: vjgv@servidor.unam.mx).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMECH.2008.2005197



Fig. 1. Spider refitted for autonomous navigation.

and more) for robot path planning [2]–[9]. Each of these research activities addresses path planning and navigation differently. Researchers need to comprehend a range of complex issues involved in navigation, even though these may not be directly related to their prime research. One such task is the process of collecting a series of waypoints for mobile robot path planning. The most common procedure is through manual collection (surveying) of a series of waypoints using a high-precision differential GPS (DGPS) receiver [4], [10], either using real-time kinematics (RTK) or postprocessing the data. Even though this is a simple task, it is time-consuming and requires thorough knowledge of the robot's working environment.

The concept of using imagery for defining waypoints is not a new idea. Freely available geographic information system (GIS) tools such as *Google Earth* are often used by civilians in order to define their own route of travel [11], whether it be for hiking or driving. For in-car GPS navigation, the accuracy of these points is not critical since the waypoints are often conveyed relative to a global fixed street network, and are not required for autonomous navigation—therefore, positional inaccuracies from the GPS receiver and the waypoint positional resolution do not act as a hindrance on the system's overall performance. However, for applications requiring higher navigation precision and autonomy, such as in mobile robots, greater significance must be attributed to image settings and coordinate reference systems to improve the waypoint accuracy, and GPS settings to ensure that the waypoints are reached.

The freely available DGPS signal (Wide Area Augmentation System (WAAS)/European Geostationary Navigation Overlay Service (EGNOS)), shows potential for mobile robots as it offers positional accuracy to within 3 m. It can be used in conjunction with aerial images for mobile robot waypoint navigation and is an exciting area of development. The upcoming deployment of the Galileo system (Europe's alternative to the GPS, which promises positional accuracy to within 1 m with no signal degradation all the year round [12]) shows further potential.

1083-4435/\$25.00 © 2008 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on February 23, 2009 at 18:31 from IEEE Xplore. Restrictions apply.

It is important to note that this system is not intended to replace the need for an inertial measurement unit (IMU), and will not provide submeter accuracy like those obtained from the subscription-based differential global positioning system (DGPS); however, the interest in this research is to extend the capabilities of the currently available WAAS/EGNOS signal using a low-cost GPS sensor for localization. As will be clarified in Section III-A, the assumed working environment will be 2-D, and therefore, relatively flat landscape has been used. The projected use of this system in its current form will be in open-space agricultural environments and uncluttered urban landscapes.

This paper is divided into four main sections. The first, Waypoint Navigation (Section II) will discuss recent work that uses waypoints in mobile robot path planning. That will then lead the reader into the current uses of waypoint determination using imagery. The second, Imagery (Section III) will discuss the relative advantages and disadvantages of aerial over satellite imagery, the critical process of georeferencing (i.e., calibration) of these images, and finally, the selection of the correct map settings (map datum and projection) for the working area. Both steps are critical if direct spatial comparison is desired. The third section, Robotic System (Section IV), will briefly discuss the GPS, and provide the reader with a novel derivation of the kinematics for a synchronous drive robot and present the controller used for the simulation. The fourth, entitled Experimental Results (Section V) will demonstrate the effect of GPS positional correction, accuracy, and repeatability on waypoints reached. It will also show the waypoint accuracy that can be typically obtained using an orthorectified aerial image. Finally, a simulation using the defined kinematics and controller will be used to demonstrate a working system that combines selected waypoints with simulated GPS positional output error from the robot, with results presented in terms of the percentage of waypoints reached with respect to radius size.

II. WAYPOINT NAVIGATION

In mobile robots, it is quite difficult to separate the concepts of path planning and navigation, since a path is often planned with an appropriate controller in mind. A vast number of path planning techniques are in existence, and in this paper, a deliberative approach is used, wherein the robot follows a predefined trajectory or a series of points.

Predefined path planning is termed the "railway track algorithm" in [13] because the vehicle is confined to specific paths or roadways (the "tracks"). This is usually done when the coordinates of the path to be traversed are given to the robot in a series of known coordinates (*waypoints*). Classical path planning techniques assume a full knowledge of the robot's environment, which is believed to be correct and complete, but since complete knowledge of the environment for outdoor robots is not possible, a method employing waypoint-type algorithms is suitable [14].

It has been shown that a low-cost educational robot, equipped with only a GPS receiver as its sensor, has obtained good results for waypoint navigation [15]. Waypoint navigation has also been used in an autonomous boat, yielding satisfactory results through the use of DGPS [16]. With positional fix updates roughly once per second, the author was able to achieve positional accuracy up to 1 m for the application. Furthermore, work conducted on the use of waypoints for an autonomous Kiteplane achieved successful maneuvering under wind disturbances using low-cost sensors [17].

The use of waypoints derived from aerial imagery has also been well received by some researchers working on mobile robots in rugged outdoor environments [18], [19].

In summary, the use of waypoints and GPS have been shown to be powerful tools for outdoor mobile robot navigation. For this research, *waypoints* will refer to outdoor points within a predefined positional coordinate system, which will be clarified in the following sections.

III. IMAGERY

A. Aerial Images

As mentioned previously, a recent trend in navigation and area representation methods has been the use of various types of imagery. This paper will focus on the use of aerial images (photographs) and not on 3-D DEMs such as LIDAR/LADAR, since low-cost GPS units do not provide accurate altitude data. Freely available or low-cost imagery (e.g., Google Earth) can be several years old and of variable image resolution, rendering it useless for many applications; however, freely available data remain useful for conveying the landscape for various purposes.

There are many types of orbital satellites that collect images, such as Landsat, Satellite Pour l'Observation de la Terre (SPOT), and Indian Remote Sensing (IRS); however, most have a lower resolution (i.e., less detail) than the recently launched IKONOS and QuickBird. The latter two were developed to provide high-resolution imagery for both civil and government use. Many (>30) new remote sensing satellite systems are now operational in addition to 12 further planned launches within the next year [20], which boast even higher image resolution and positional accuracy. IKONOS provides spatial resolution of up to 0.8 m panchromatic ground sample distance (GSD) and 4 m multispectral GSD, whereas QuickBird's resolution is sharper at 0.6 and 2.4 m [20]. Several agencies sell these high-resolution images; however, they are often too expensive for the average user, as a minimum purchase area applies.

Aerial photographs provide a useful alternative to satellite imagery, because they have the advantage of being acquired at closer range than satellites, and consequently provide higher *scale* and *detail/resolution*. These two attributes are necessary to assist enhanced waypoint identification. For example, an aerial photograph taken at 300 m above ground level with the "normal" 150 mm focal length lens has a resolution on the ground of 0.08 m per pixel [21], which is more precise than both IKONOS and QuickBird. Another low-cost approach for acquiring aerial imagery is a system for remote sensing, deployed in times of disaster [22], which could be used for waypointbased navigation. In this, a mechatronic kite equipped with a teleoperated camera and other sensors have been used for live data capture with the advantage of rapid deployment. Finally, another method of capturing aerial images includes using an unmanned aerial vehicle that obtains aerial LADAR data [8]. Irrespective of the image used, postimage processing is required for georeferencing.

B. Georeferencing

This is the process in which the image is related to a suitable ground coordinate system. Since the earth is not a perfect sphere, setting these factors to a fixed universal mathematical index, such as the widely used World Geodetic System 1984 (WGS84), could lead to inaccuracies of several meters, depending on the geographical location of the image in the global frame [23]. This leads to two concepts: map datum and map projection [23]. It is important to set the aerial images to the datum, and projection used to represent the country in which the image was taken. In the UK, for example, the map projection used is known as the transverse mercator (TM), and the map datum as the Ordnance Survey Great Britain 1936, which is based on a geographic representation known as the Airy 1830 ellipsoid. Direct transformations between various map datums (e.g., OSGB36 to WGS84) can be achieved using, for example, the Helmert transformation. Unfortunately, such transformations are only approximate at the local scale. In the UK, for example, small-scale inaccuracies arising from the 1936 retriangulation lead to significant positional errors up to 20 m [23]; therefore, using simple global transformations and published constants is not advisable. It is important to ensure that a consistent underlying coordinate system for the aerial image being used, and that the GPS positional output matches its corresponding location on the image. The fusion of the GPS positional data output to the underlying coordinate system in the aerial image is explained in detail in Section V-B.

C. Photogrammetry

The science developed to relate measurements of imagery to a ground coordinate system is known as photogrammetry [24], the impetus for development being primarily the production of the World's National Mapping series [25]. There are two types of distortion inherent in any aerial or satellite image that prevent direct correspondence between the 2-D image and a 3-D ground coordinate system: tilt and relief distortion. Distortions created by the light rays leaving the object, passing through the lens center, before creating an image point in the focal plane of the camera are modeled explicitly using the collinearity equations [24], [25]. These equations model distortions completely due to nonverticality of the sensor. A distortion is also introduced into the image if the terrain is nonplanar. Such "relief displacements" are related to the flying height and focal length of the sensor, and can be highly significant for aerial photography. Only a true "orthorectification" procedure implementing the collinearity equations removes the distortions due to both relief and tilt displacement. Unfortunately, there are a range of aerial image products marketed that have not been generated using the required rigorous mathematical procedures. Although such "map accurate" products are fit for many purposes/applications, they should always be used with caution, particularly when used in conjunction with GPS.

The orthorectification procedure can be accomplished by using ground control points (GCP[s]) clearly visible on the aerial images. The 3-D coordinates of the GCPs should be established using a survey grade differential grade GPS and linked to the ordnance survey (OS) "passive network." These coordinates should subsequently be transformed to OSGB36 using the OSTN02 and the OSGM02 models provided by the OS [23]. Unfortunately, there is no single solution available, and different approaches are required in different countries. Advice should be sought from National Mapping Agencies.

The process of orthorectification can introduce discrepancies if the DEM is inaccurate. Therefore, it is important to consider such uncertainties when judging the inaccuracy of the waypoint selected from an aerial image.

IV. ROBOTIC SYSTEM

A. Global Positioning System

Since the GPS was selected as a stand-alone localization sensor, this section is dedicated to providing a brief description of its capabilities. Currently, there exist two truly global satellite positioning systems, the U.S. GPS and the former USSR GLONASS [26]; however, the first satellite of the "Galileo" European positioning system has been launched.

Inaccuracies stemming from atmospheric conditions, orbit instability, and disturbances in the satellite constellation are adjusted by accurately georeferenced ground stations, which act as beacons and transmit corrected GPS signals [27]. This is known as DGPS. However, the accuracy of the corrected signals degrades as the distance from these stations increases, and subscription can be costly. The reduced accuracy of the GPS system has been handled by the introduction of geostationary satellites that transmit differentially corrected signals. In the U.S., this system is known as WAAS, and in Europe, as the EGNOS, and can provide civilians with positioning accuracies to within 3 m. The GPS has become a topical subject among researchers; however, it is rarely used independently for localization.

B. Kinematics and Controller

Reference [28] presents the controller that has been used for the simulation of the synchronous drive robot. It is implemented in order to verify the efficiency of the control system in reaching waypoints by means of a simulated positional output from the GPS. One general kinematic model for a synchronous drive robot can be found in work done in [29]. However, given that a convenient model to work within the Simulink environment is not readily available, a detailed step-by-step derivation is presented. The following assumptions are made: the robot has synchronous wheel rotation, a symmetric square wheel configuration, homogeneous wheel radii, no lateral or longitudinal wheel slip, no wheel misalignments, no pressure differences in tires, and moves along a 2-D plane. This simplifies the kinematic model to the basic constraints acting on the robot. A schematic is used for the derivation (see Fig. 2). Table I summarizes the variables.

Symbol	Description	Units
θ	Steering angle	Radians
$\dot{\theta}$	Steering velocity	Radians/Second
α	Angle of rotation of Spider [*] frame to global frame	Radians
d,-d	Robot dimension	Meters
	along X _{Spider} axis	
b,-b	Robot dimension	Meters
	along Y_{Spider} axis	
φ	Wheel angle	Radians
ø	Wheel angular velocity	Radians/Second
v	Robot linear velocity	Meters/Second
Ę	Robot posture vector in global frame	(Meters, Meters, Radians)
Ψ.	Robot velocity vector in global frame	(Meters/Second, Meters/Second, Radians/Second)

TABLE I SYMBOLS, DESCRIPTION, AND UNITS FOR Spider's KINEMATIC MODEL



Fig. 2. Kinematic model of the four-wheeled synchronous drive robot.

The robot has two decoupled synchronous mechanisms: the synchronous wheel rotation around each wheel's axle and the synchronous wheel steering mechanism. The wheels can be steered simultaneously 360° continuously and unhindered at the same angular velocity and direction. For that reason, the instantaneous center of curvature (rotation) (ICC or ICR) is at infinity. The wheels also travel at the same linear velocity.

This leads to the fact that the robot's frame will remain constant by an angle (α) to the global reference frame, unless wheel slipping or other unforeseen external dynamic factors occur.

The *robot posture* (ξ_{Global}) can be defined by the following vector representation:

$$\xi_{\text{Global}} = \begin{bmatrix} x & y & \alpha \end{bmatrix}^T. \tag{1}$$

Since the global reference frame and the robot frame are not aligned, it is necessary to map the motion of the global frame to



Fig. 3. Wheel's kinematic constraints frames: orthogonal and along wheel plane.

that of the robot. To achieve this, an *orthogonal rotation matrix* $(R(\alpha))$ is needed

$$R(\alpha) = \begin{bmatrix} \cos \alpha & \sin \alpha & 0\\ -\sin \alpha & \cos \alpha & 0\\ 0 & 0 & 1 \end{bmatrix}.$$
 (2)

The calculation is denoted by

$$\xi_{\text{Spider}} = R(\alpha)\xi_{\text{Global}}.$$
(3)

The next stage is to calculate the wheel's kinematic constraints. Since this is a synchronous mechanism, the calculation of one wheel is sufficient. For this, both constraints orthogonal to and along the wheel plane need to be determined. Refer to Fig. 3.

In order to compute the correct constraints, it is vital to determine the type of wheel being used. For this robot, it belongs to the class of *steered standard wheels*. Further explanation is provided in [30]. The resolved equations are as follows.

Along the wheel plane:

$$\lfloor \cos(\theta_i) & \sin(\theta_i) & d_i \sin(\theta_i) - b_i \cos(\theta_i) \rfloor \cdot R(\alpha) \dot{\xi}_{\text{Global}} - r \dot{\phi}_i = 0.$$
(4)

Orthogonal to the wheel plane:

$$\begin{bmatrix} -\sin(\theta_i) & \cos(\theta_i) & d_i \cos(\theta_i) + b_i \sin(\theta_i) \end{bmatrix}$$

$$\cdot R(\alpha) \dot{\xi}_{\text{Global}} = 0$$
 (5)

where $\dot{\xi}_{\text{Global}} = [\dot{x} \quad \dot{y} \quad \dot{\alpha}]^T$ is the robot's posture velocity vector, θ_i is the steering angle at a certain instant in time, and d_i, b_i are the positions of the wheels with respect to point *P* along the robot's frame, where the subscript (*i*) corresponds to each individual wheel. Given that the Spider has a symmetric

Authorized licensed use limited to: IEEE Xplore. Downloaded on February 23, 2009 at 18:31 from IEEE Xplore. Restrictions apply.

four wheel configuration (d = b), then

$$d_{\rm fl} = d_{\rm fr} = b_{\rm bl} = b_{\rm fl} = d$$

$$d_{\rm br} = d_{\rm bl} = b_{\rm br} = b_{\rm fr} = -d.$$
 (6)

Therefore, (4) and (5) can be modified with the dimensions presented in (6) to obtain the full kinematic wheel constraints acting on the wheel frame.

Following this, the kinematic constraints need to be expressed in the matrix form $A(q)\dot{q} = 0$. In order to obtain the *state-space representation* of the robot, it is important to determine the *null space* of A(q) for $v = r\dot{\phi}$ and include the steering velocity $\dot{\theta}$ in the form of $\dot{q} = S_q u$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\theta}_{br} \\ \dot{\theta}_{br} \\ \dot{\theta}_{fr} \\ \dot{\theta}_{fr} \\ \dot{\theta}_{fr} \\ \dot{\phi}_{br} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fr} \end{bmatrix} = \begin{bmatrix} \cos(\theta + \alpha) & 0 \\ \sin(\theta + \alpha) & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1/r & 0 \\ 1/r & 0 \\ 1/r & 0 \\ 1/r & 0 \end{bmatrix} \begin{bmatrix} v \\ \dot{\theta} \end{bmatrix}.$$
(7)

Since $\dot{\theta}_i = \dot{\theta} \,\forall i, \, \dot{\phi}_i = \dot{\phi} \,\forall i, \, \text{and} \, v = r\dot{\phi}$, (7) can be reduced to

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} r\cos(\theta + \alpha) & 0 \\ r\sin(\theta + \alpha) & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix}.$$
 (8)

Referring back to (8), it can be seen that five factors are needed to determine the robot's velocity components in the x-y plane (forward kinematics), where the robot's steering velocity $\dot{\alpha}$ is zero (i.e., the orientation of the robot's platform α never changes under the aforementioned conditions).

For simplicity, the Spider is assumed here to move in the forward translational velocity direction only. The control strategy (9) is based on a slightly modified version of the "Reaching the Goal" approach [28], as previously mentioned. Fig. 4 shows a schematic of that approach. The angular velocity is a function of the sine of the error (θ_e), where the maximum angular velocity will be achieved at $\pm 90^{\circ}$

$$P_{\text{GPS}} = (x_{\text{GPS}}, y_{\text{GPS}})$$
$$\theta_g = \tan^{-1} \left(\frac{y_w - y_{\text{GPS}}}{x_w - x_{\text{GPS}}} \right) \qquad \theta_e = \theta_g - \theta_i.$$
$$\dot{\theta}_x = \dot{\theta}_{\max} \sin(\theta_e) \qquad v = v_{\max} \qquad (9)$$



Fig. 4. Waypoint reaching model based on [28].

V. EXPERIMENTAL RESULTS

In order to conduct the set of tests presented in this paper, the following were used: 1) a 0.18-m/pixel resolution aerial image, obtained commercially, of the Holywell car park at Loughborough University, orthorectified into OS coordinates; 2) two Leica System 500 receivers for precise differential point positioning using static data postprocessing (horizontal accuracy 5 mm + 1 ppm, vertical accuracy 10 mm + 1 ppm); 3) Garmin 18 5Hz GPS unit; 4) Matlab Simulink model for a synchronous drive robot; 5) Erdas imagine 9.0 by Leica Geosystems; and 6) freely available GPS planning software (trimble planning software). Figs. 5, 7, 8, and 10 are in Eastings and Northings in OSGB36, the National Mapping Framework for the UK.

A. Aerial Image and Waypoint Accuracy

In order to show the disparities between a georeferenced aerial image and waypoints, two tests were performed. In the first one, the Leica System was used to collect 54 points using a survey style "stop-and-go" approach to measure points covering the majority of the parking lot. These points are superimposed on the aerial image using Matlab (Fig. 5), which is also used for superimposing the GPS positional results on the aerial images. Clearly recognizable and identifiable landmarks on the image (marked as waypoints) were chosen as points to be surveyed by the high-precision GPS on their corresponding points in the car park. Superimposing the user-selected waypoints alongside these surveyed points on the aerial image shows that discrepancies in the position are apparent, two of which can be clearly seen in Fig. 6. It was determined that, for the clearly recognizable points (37/54), the surveyed points had an average 0.37 m North East shift from the user selected waypoints (varying from 0.087 to 0.732 m) [see Fig. 6(a) and (b)] for a waypoint comparison.

On the other hand, for the entire dataset (54/54), an average 0.446 m N/E shift from the user-selected waypoints was obtained (varying from 0.087 to 2.085 m). Such differences can be accounted by the presence of *variability* and *bias error*.

Authorized licensed use limited to: IEEE Xplore. Downloaded on February 23, 2009 at 18:31 from IEEE Xplore. Restrictions apply.

696

SHAIR et al.: USE OF AERIAL IMAGES AND GPS FOR MOBILE ROBOT WAYPOINT NAVIGATION



Fig. 5. Fiftyfour accurately surveyed points using static post data processing.



Fig. 6. (a) and (b) Discrepancy between the user-selected point (dark colored) and the surveyed point (light colored).

The bias error arises from small inaccuracies involved in the measurement process, most significant being slightly varying parameters in the processing between the established photocontrol and checkpoints. There is also a small and systematic height bias in the extracted DEM, which causes a systematic shift in the position of the pixels comprising the orthorectified image. The *variability* usually relates to natural human induced variation; waypoints selected from an image by one person may differ from a set collected by another. This is represented by the range or standard deviation.

Given the variation of the shift throughout the image, it is evident that it is not entirely possible to match an image waypoint to the actual location in the car park. Therefore, it is important to define a proximity error around each waypoint. This proximity error, however, is left up to the user to define since it should be based on the image resolution, the image positional inconsistencies due to orthorectification, and human error concerning waypoint selection. It is possible to recalibrate the image to the standard needed; however, this would be a daunting task for the average user, and might be beyond the accuracy needed.

This leads to the next set of experimental results that demonstrate the importance of adjusting the GPS receivers' positional shift, to improve the spatial match between the GPS data and the orthorectified image.



Fig. 7. Dark lines indicate the GPS results prior to positional correction, and the light lines indicate the post positional correction.

B. GPS Positional Correction

WGS84 is the default coordinate system adopted by the GPS receiver. Any other coordinate system selected would be based on a mathematical transformation from the default-which yields erroneous results (see Section III-B). Because the GPS showed positional variation for a single spot from one day to the next, irrespective of the coordinate system chosen, it was determined that adopting a mathematical spatial shift would inevitably provide significantly improved positional accuracy. This would overcome some of the computational errors obtained due to the receiver's internal Molodensky coordinate system transformation [31]. This would provide "corrected" (or tuned) positional data, suitable for a certain time period and geographic location. The approach adopted showed positional stability for approximately 1.5 h of testing. Since it was also determined that the OSGB36 coordinate system was spatially not as close to the position being measured as anticipated, parameters local to the test area were obtained to improve the overall spatial position. This would therefore reduce the amount of mathematical compensation needed. The constants used for the "user-defined settings" were: inverse flattening factor (Df): 299.3249646; the semimajor axis, equatorial radius (Da): 6377563.396; positional shift along x-axis (dx): 371; positional shift along y-axis (dy): -112; and the positional shift along z-axis (dz): 434. This is based on the Airy 1830 ellipsoid.

The GPS positional tuning was conducted using the following method: one point in a relatively open area was precisely surveyed. The Garmin GPS was then placed on the same location, at a height equivalent to the robot's GPS height of 1.5 m, to determine an average value over a 15-min sample time. The data were then converted to Eastings/Northings, and compared to its corresponding surveyed point. The positional shift was then used to compensate for the positional output from the GPS during forthcoming tests.

One test, conducted on the car park, shows the result of the path data before and after this GPS positional correction approach. The GPS unit was attached to a trolley and driven around a designated marked line in the road's center. Fig. 7 shows the effect of this positional correction. This test was conducted for 30 min for a total traveled distance of 1.3 km (each turn 420.8 m). In order to test the repeatability, the error of the GPS,

697



Fig. 8. Nineteen waypoints used for testing GPS positional correction approach, repeatability, and accuracy.



Fig. 9. Percentage of waypoints hit for 17 runs before and after positional correction.

and the number of waypoints hit, a further test was performed. A series of waypoints, seen in Fig. 8, were created and the GPSmounted trolley was also driven through a designated marked line for 17 runs (\sim 1.43 km) for 55 min. In order to ensure optimal results, the Trimble planning software was used to determine the most suitable time for testing. An open space area was used to ensure an unobstructed sky view, and no vehicles were present.

For the majority of the time, there were ten satellites in view with a horizontal dilution of precision (Hdop) ranging from 0.9 to 1.1, occasionally reaching 1.3.

The results for varying the waypoint radius, for both postand precorrections can be seen in Fig. 9. Following the testing, the GPS positional data were checked once more to determine that an Easting shift of 0.07 m and approximately 0.3 m in the Northing had occurred. The percentage of waypoints hit may vary from one day to another, and therefore, depending on the accuracy required, the proximity error (radius) can be adjusted. Present GPS positioning data shows improved positional accuracy for mobile robot navigation compared to results prior to the deployment of the geostationary satellites (EGNOS/WAAS) [32].

C. Simulation Results

To test the effectiveness of the control system in reaching a waypoint with proximity error in the presence of GPS inaccu-



Fig. 10. Robot passing through the waypoints in a simulated GPS environment.

racy, the following simulation is presented. A challenge with such a simulation is the ability to mimic the positional accuracy of the GPS. Therefore, the GPS results from the fieldwork were used to obtain an estimate of this positional behavior after mathematical compensation. A continuous random number generator function was used to produce an overall uniform normal distribution, both along the robot's *x*- and *y*-axes. This, paired with a random positional shift within a user selected radius, yielded results that mimic the real GPS output. With the addition of this simulated positional response, a level of uncertainty and unpredictability was added, making the simulation more realistic.

For consistency, the robot's simulated linear velocity was fixed at 5 km/h, and its angular velocity at 0.76 rad/s (one revolution in ~9 s). The robot has a square configuration of 1.3 m \times 1.3 m. The proximity error of the GPS was set to 0.8 m and the individual positional data to within 0.2 m.

The results yielded a waypoint hit of 94.73% for 18 test runs at a waypoint radius of 0.7 m. A hit was considered only when the robot's center passed through the proximity error of the waypoint. The efficiency of that controller, however, also depends on the speed and the angular velocity of the robot. A higher hit count was achieved at a lower linear velocity. The majority of the misses occurred during turning maneuvers (due to overshoot). A future modified version of the controller would vary its translational and angular velocity during turning to ensure that the waypoint is reached. Fig. 10 is the view of one of the simulations.

The simulation result shows that with this degree of GPS accuracy from a low-cost GPS receiver, a working prototype of a unified system of selecting waypoints from an aerial image can provide satisfactory waypoint navigation. Furthermore, it can be used as a reasonable benchmark for testing various control systems prior to deployment.

ACKNOWLEDGMENT

The authors would like to thank the Department of Civil Engineering, Loughborough University, Loughborough, UK, for SHAIR et al.: USE OF AERIAL IMAGES AND GPS FOR MOBILE ROBOT WAYPOINT NAVIGATION

allowing the use of their survey grade equipment, and the technicians D. Liddell and T. Harrison for their valuable assistance.

REFERENCES

- [1] S. Shair, R. M. Parkin, and M. R. Jackson, "A robust mobile platform for autonomous and teleoperative navigation," in Proc. 10th Int. Conf. Mechatronics Technol., México City, México, 2006, pp. 1-6.
- [2] J. Meguro, K. Ishikawa, Y. Amano, T. Hashizume, J. Takiguchi, R. Kurosaki, and M. Hatayama, "Creating spatial temporal database by autonomous mobile surveillance system (a study of mobile robot surveillance system using spatial temporal GIS part 1)," in Proc. 2005 IEEE Int. Workshop Saf., Security Rescue Robot., Kobe, Japan, Jun. 6-9, pp. 143-150
- [3] D. Silver, B. Sofman, N. Vandapel, J. A. Bagnell, and A. Stentz, "Experimental analysis of overhead data processing to support long range navigation," in Proc. 2006 IEEE/RSJ Int. Conf. Intell. Robots Syst., Beijing, China, Oct., pp. 2443-2450.
- [4] P. Sim, V. Sacco, and G. S. Virk, "Robot navigation in volcanic environments," in Proc. 2003 IEEE Conf. Intell. Transp. Syst., Shanghai, China, Oct.12-15, vol. 2, pp. 1010-1015.
- [5] S. Zein-Sabatto, O. Taiwo, and P. Koseeyaorn, "Integrated simulation software for outdoor robots planning and coordination," in Proc. 2004 IEEE Southeast Conf., Greensboro, NC, Mar. 26-29, pp. 215-221.
- [6] A. Howard, H. Seraji, and B. Werger, "Global and regional path planners for integrated planning and navigation," J. Robot. Syst., vol. 22, no. 12, pp. 767-778, 2005.
- [7] G. Muscato, D. Caltabiano, S. Guccione, D. Longo, M. Coltelli, Cristaldi, E. Pecora, V. Sacco, P. Sim, G. S. Virk, P. Briole, A. Semerano, and T. White, "ROBOVOLC: A robot for volcano exploration result of first test campaign," Ind. Robot, vol. 30, no. 3, pp. 231-242, 2003
- [8] N. Vandapel, R. Donamukkala, and M. Hebert, "Experimental results in using aerial ladar data for mobile robot navigation," presented at the 4th Int. Conf. Field Service Robot., Fuji, Japan, 2003.
- B. Sofman, E. Lin, J. Bagnell, N. Vandapel, and A. Stentz, "Improving robot navigation through self-supervised online learning," presented at the Robot.: Sci. Syst. Conf., Philadelphia, PA, Aug. 2006.
- [10] B. Sethuramasamyraja, "GPS based waypoint navigation for an autonomous guided vehicle-Bearcat III," M.S. thesis, Dept. Mech., Ind. Nucl. Eng., Univ. Cincinnati. Cincinnati, OH, 2003.
- [11] M. Wuersch and D. Caduff, "Refined route instructions using topological stages of closeness," in Lecture Notes in Computer Science. Germany: Springer-Verlag, 2005. Berlin.
- [12] V. Ashkenazi, D. Park, and M. Dumville, "Robot positioning and the global navigation satellite system," Ind. Robot, vol. 27, no. 6, pp. 419-426, 2000.
- [13] S. Cameron, "Obstacle avoidance and path planning," Ind. Robot, vol. 21, no. 5, pp. 9-14, 1994.
- [14] H. Durrant-Whyte, "Autonomous land vehicles," in Proc. Inst. Mech. Eng. I: J. Syst. Control Eng., 2005, vol. 219, no. 1, pp. 77-98.
- [15] M. E. Holden, "Low-cost autonomous vehicles using just GPS waypoint navigation," presented at the 2004 ASEE Conf., Salt Lake City, UT.
- T. W. Vaneck, "Fuzzy guidance controller for an autonomous boat," IEEE [16] Control Syst. Mag., vol. 17, no. 2, pp. 43-51, Apr. 1997.
- [17] M. Kumon, Y. Udo, H. Michihira, M. Nagata, I. Mizumoto, and Z. Iwai, "Autopilot system for Kiteplane," *IEEE/ASME Trans. Mechatronics*, vol. 11, no. 5, pp. 615-624, Oct. 2006.
- [18] M. H. Bruch, G. A. Gilbreath, J. W. Muelhauser, and J. Q. Lum, "Accurate waypoint navigation using non-differential GPS," in Proc. AUVSI Unmanned Syst. 2002, Lake Buena Vista, FL, Jul. 9-11, pp. 9-11.
- [19] J. Feddema, C. Lewis, and P. Klarer, "Control of multiple robotic sentry vehicles," in Proc. SPIE: Int. Soc. Opt. Eng., Orlando, FL, 1999, vol. 3693, pp. 212-223.
- [20] S. Booth, Ed., "Earth observation sensors: Higher resolutions and shorter revisit times," Eng. Survey. Showcase 2007, no. 2, pp. 19-25.
- [21] D. T. Booth, S. E. Cox, and R. D. Berryman, "Precision measurements from very large-scale aerial digital imagery," Environ. Monit. Assess., vol. 112, no. 1-3, pp. 293-307, 2006.
- [22] P. Y. Oh and W. E. Green, "Mechatronic kite and camera rig to rapidly ac-quire, process, and distribute aerial images," *IEEE/ASME Trans. Mecha*tronics, vol. 9, no. 4, pp. 671-678, Dec. 2004.
- A Guide to Coordinate Systems in Great Britain, vol. 1.7, D00659. [23] (2007, Oct.). Ordnance Survey, Southampton, UK [Online]. Available: http://www.ordnancesurvey.co.uk

- [24] J. Fryer, H. Mitchell, and J. Chandler, Applications of 3D Measurement From Images. Caithness, UK: Whittles Publishing, 2007, p. 304.
- [25] P. R. Wolf, Elements of Photogrammetry. Singapore: McGraw-Hill, 1983, p. 628.
- [26] A. Michalski and J. Czajewski, "The accuracy of the global positioning systems," IEEE Instrum. Meas. Mag., vol. 7, no. 1, pp. 56-60, Mar. 2004.
- [27] C. Satirapod, I. Trisirisatayawong, and P. Homniam, "Establishing ground control points for high-resolution satellite imagery using GPS precise point positioning," in Proc. IEEE Int. Geoscience and Remote Sens. Symp., IGARSS'03, Toulouse, France, Jul. 21-25, 2003, vol. 7, pp. 4486-4488.
- V. J González-Villela, R. M. Parkin, M. Lopez-Parra, J. M. Dorador-González, and L. Guadarrama, "A wheeled mobile robot with obstacle avoidance capability," Ing. Mec. Tecnol. Desarrollo. Rev. Soc. Mex. Ing. Mec. (SOMIM), vol. 1, no. 5, pp. 159–166, 2004. [29] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to
- collision avoidance," IEEE Robot. Autom. Mag., vol. 4, no. 1, pp. 23-33, Mar. 1997.
- [30] R. Siegwart and I. R. Nourbakhsh, Introduction to Autonomous Mobile Robots. Cambridge, MA: MIT Press, 2004.
 [31] D. DePriest, A GPS User Manual: Working With Garmin Receivers.
- Fairfield, CA: 1st Books Library, 2003, ch. 5.
- [32] S. Panzieri, F. Pascucci, and G. Ulivi, "An outdoor navigation system using GPS and inertial platform," IEEE/ASME Trans. Mechatronics, vol. 7, no. 2, pp. 134-142, Jun. 2002.

S. Shair received the B.Sc. degree in mechanical engineering from Queen's University, Kingston, ON, Canada, in 2005. He is currently working toward the Ph.D. degree at Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, UK.

J. H. Chandler received the B.Sc. degree in land surveying and geography from Newcastle University, Newcastle, UK, in 1983, and the Ph.D. degree in photogrammetry and geomorphology from City University, London, UK, in 1989

He is currently a Senior Lecturer in the Department of Civil and Building Engineering, Loughborough University, Loughborough, UK.

V. J. González-Villela (M'08) received the B.Eng. degree in mechanical and electrical engineering and the M.Eng. degree in electrical engineering from the National Autonomous University of Mexico (UNAM), Mexico City, Mexico, in 1987 and 1993, respectively, and the Ph.D. degree in kinematics, dynamics, and nonlinear systems applied to mobile robotics modelling and control from Loughborough University, Loughborough, UK, in 2006.

He is currently a Professor in the Department of Mechatronics Engineering, UNAM.

R. M. Parkin (M'94) received the B.Sc. degree in engineering science from Leicester University, Leicester, UK, and the Ph.D. degree in diesel gas turbines from Leicester Polytechnic, Leicester, in 1976 and 1980, respectively.

He is currently Professor of mechatronics at Loughborough University, Loughborough, UK, where he is also the Head of the Wolfson School of Mechanical and Manufacturing Engineering.

Prof. Parkin was a founder member of the UK Mechatronics Forum.

M. R. Jackson received the B.Sc. (Hons.) degree in mechanical engineering from Salford University, Salford, UK, in 1975, the M.Sc. degree in flexible manufacturing systems and robotics from Loughborough University, Loughborough, UK, in 1987, and the Ph.D. degree in machine tool vibration and surface finish from the Council for National Academic Awards (CNAA) and Leicester Polytechnic, Leicester, UK, in 1986.

He is currently Professor of machine systems at Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University.