October 2000

Using LabVIEW to prototype an industrial-quality real-time solution for the Titan outdoor 4WD mobile robot controller

D. Ratner
University of Wollongong, dannyr@uow.edu.au

Phillip J. McKerrow
University of Wollongong, phillip@uow.edu.au

Abstract

In the Titan project we applied a new approach to prototyping mobile robots by choosing tools which are commonly used by leading aerospace manufacturers and many other industries. We have gained substantial experience when using the LabVIEW real-time programming environment coupled with the industrial quality data acquisition cards, both are made by National Instruments. The methodology of virtual instruments software tools combined with the graphical programming environment was found to be very efficient for interactive cycles of design and testing, which are at the core of robotics prototyping.

Keywords

control system CAD, engineering graphics, mobile robots, programming environments, real-time systems, virtual reality

Disciplines

Physical Sciences and Mathematics

Publication Details

This paper originally appeared as: Ratner, D & MCKerrow, P, Using LabVIEW to prototype an industrial-quality real-time solution for the Titan outdoor 4WD mobile robot controller, IROS 2000. Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems, 31 October-5 November 2000,vol 2, 1428-1433. Copyright IEEE 2000.

Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems

Using LabVIEW to prototype an industrial-quality real-time solution for the Titan outdoor 4WD mobile robot controller

Danny Ratner
Intelligent Robotics Laboratory
School of Information Technology and Computer Science
University of Wollongong
NSW 2522
Australia
email: dr19@uow.edu.au
www.uow.edu.au/~dr19/

Phillip McKerrow
Intelligent Robotics Laboratory
School of Information Technology and Computer Science
University of Wollongong
NSW 2522
Australia
email: phillip@uow.edu.au
www.itacs.uow.edu.au/people/phillip/robotics.html

Introduction

In the Titan project we tried to apply a new approach to prototyping mobile robots by choosing tools which are commonly used by leading aerospace manufacturers and many other industries. We have gained substantial experience when using the LabVIEW real-time programming environment coupled with the industrial-quality data acquisition cards, both made by National Instruments. The methodology of virtual instruments (VI) software tools combined with the graphical programming environment was found to be very efficient for interactive cycles of design and testing which are at the core of robotics prototyping.

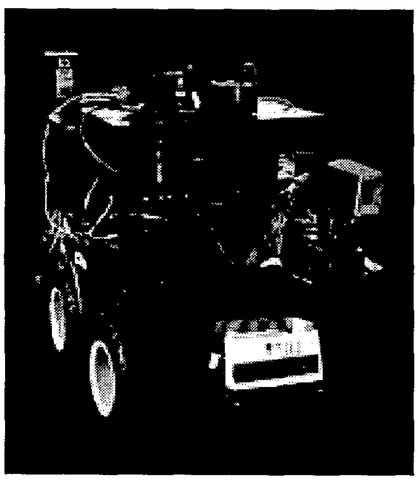

Figure 1 Titan 4WD outdoor mobile robot

Titan is a four-wheel drive outdoor mobile robot which we have developed during the last 3 years. The Titan platform has a unique floating Ackerman steering system [Pagett, 1996]. The robot is equipped with a custom [Kay L. (NZ) 1999] built CTFM (Continuous Transmission Frequency Modulation) sonar which continuously transmits chirps with a linearly decreasing frequency (100kHz to 50 kHz at 100 msec duration). The transmitter is a 20 cell phase array while the receiver consists of 4 single cells located on the 4 surfaces of a shallow pyramid.

Navigation is achieved using landmark recognition derived from feature extraction analysis in the frequency domain [Ratner & McKerrow,1998 & 2000]. Recently we have concluded a series of very encouraging experiments at the ANU (Australia National Uni. in Canberra) farm which serves as Australian Field Robotics Center (AFRC). During the experiments we could easily modify the controllers on the fly and monitor the effects on the robot's control and motion parameters. This paper will focus on Titan's motion control experience.

1 Titan's control challenge

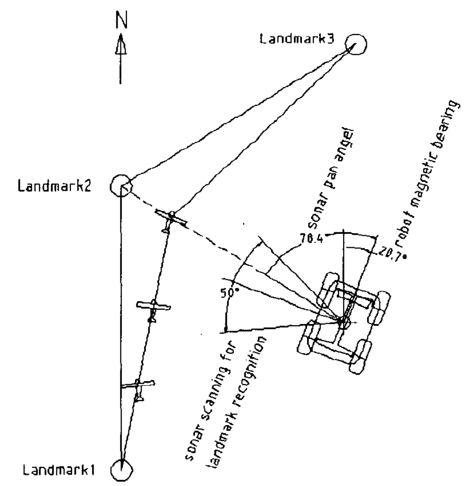

Figure 2 Recognition based navigation is similar to the strategy performed by a light aircraft flying from one landmark to the next one. The pilot recognizes the landmark, assesses the error and corrects the navigation.

The robot's basic navigation mode is to follow a sequence of legs whose parameters (distances, magnetic bearings and turning radiuses) are obtained from look-up table style navigation map. At the start of each leg, the robot scans the expected landmark area. Upon recognition, the robot calculates the location error with respect to the landmark and corrects the navigation map. Good motion control is important to the navigation success. If the open loop motion results in large location errors (over 0.5 meter in landmark range), the chance for reliable recognition is adversely diminished.

The terrain in the farm is quite rough. Simple proportional control based on emulating the joy stick

0-7803-6348-5/00/$10.00 ©2000 IEEE.

steering commands which worked well on the flat bitumen in the car park was not adequate at the farm. The floating Ackerman steering system does not have the resistance of a gear box, like in car steering, which make it sensitive to back-driven disturbances (like crops edges).

We are using a KVH Digital Gyro Compass which was originally design for yachts auto-pilots. Titan's rate of turn can be much faster than a yacht and hence the real-time difficulties of using it to control the magnetic bearing especially at the transitions between the legs.

2 Vehicle kinematics

Titan's unique kinematics and dynamics are discussed (using Matlab and Simulink models) in [Ratner & McKerrow,1999]. The unique steering system enables to achieve differential steering without skidding. The floating Ackerman steering is an hybrid between free caster wheels and the common car steering. The steering has an amount of built-in castering effect, and since no motor is driving it, the only damping is generated by the tires friction with the ground during turning. Unlike the strict car steering, Titan's may be described as "soft steering" since the turning is the result of the robot's differential velocity.

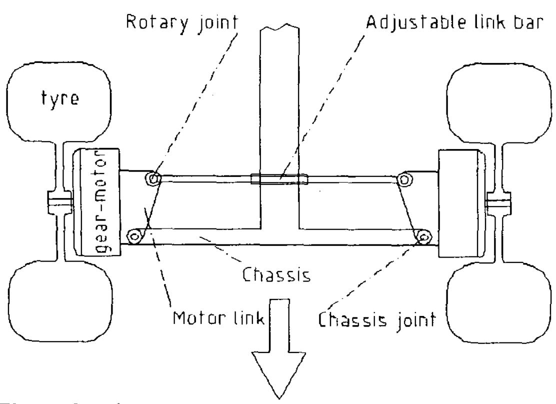

Figure 3 Titan's front drive is a unique floating Ackerman steering system. The wheels axes are located behind the front joints which contributes to the built in castering effect.

3 Titan's sensors

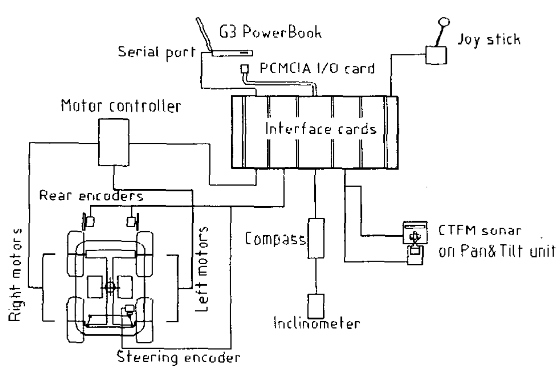

Figure 4 Titan modular design.

Figure 4 gives an over view of Titan's modular design. All the computing power is provided by a PowerBook G3. Data acquisition is achieved via NI DAQCard-1200 and the PowerBook's serial port (for switched communication with the digital compass or the pan & tilt unit). The two rear encoders provide the traveled distance. The steering encoder (mounted on the left front joint) provides steering angle for the left front wheel. The gyroscopic stabilized digital compass measures the magnetic bearing. The CTFM sonar is mounted on a pan & tilt unit. The sonar information is used for correcting the navigation errors.

4 Robot's controllers

The motion control is comprised of 3 PID VI's controllers (velocity control, bearing control and steering control) plus distance controller VI, turn controller VI and radius controller VI. During the navigation, the distance controller issues velocity commands with soft start and stop based on constant acceleration/deceleration formula (Velocity=Sqr(2*Deceleration*Distance). This gradual velocity profile is also linked to the PID clamp of the bearing controller to prevent the ramping up of the integral components. The same idea was implemented later in the turn control and radius control. The tight bearing control along the legs caused angular overshoot during turns as depicted in the upper part of Figure 5. The problem was solved (bottom part of Figure 5) by introducing feed forward turn control based on the encoders data.

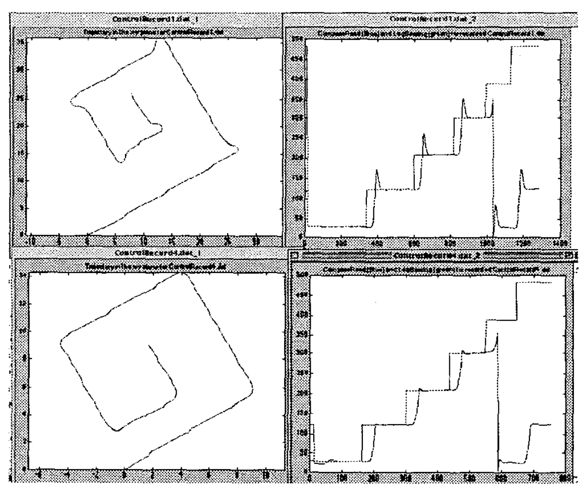

Figure 5 Trajectory from experiments on large cricket field showing the effect of feed forward turn control on reducing the bearing overshoot during sharp turns.

Figure 6 shows the front panel of the velocity controller and the block diagram of the distance controller. The velocity reading (loop time 100 msec) from the rear encoders is relative rough (0.5 mm linear motion for one pulse). Figure 7 shows the bearing controller. The loop time of this VI is 500 msec due to the slow time response of the compass. The output of the velocity and the bearing controllers are fed into the steering PID controller (closing the loop on the steering angle from the steering encoder) which has a loop time of 50 msec . This makes the steering controller very responsive.

Figure 8 shows the VI which reads the output from the KVH compass via the serial port. The compass bearing is extracted from a string which contains also the

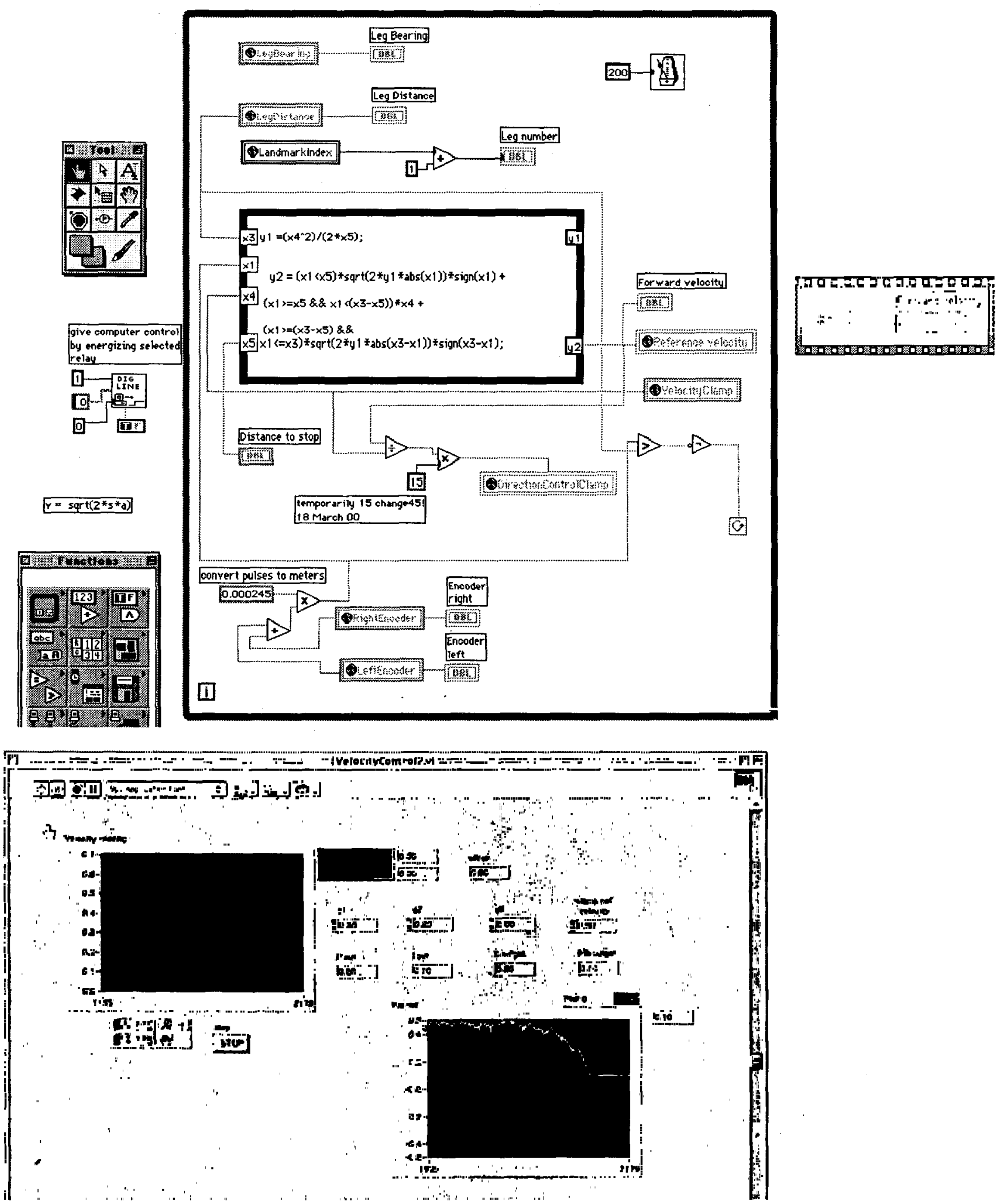

Figure 6 Diagram block for DistanceControl1.vi (upper part) . The formula node calculates the reference velocity to provide a constant acceleration and deceleration, based on distance to stop formula: Velocity = SQRT(2*Distance*Acceleration). The distance from start or distance to the stop is measured continuously from the encoders readings. The result is distance error smaller than 0.5 percent. The lower part of the figure shows the front panel of the velocity controller. The velocity reading from the encoder pulses is relative rough, but the robot follows the planned trapezoidal shape for the velocity profile.

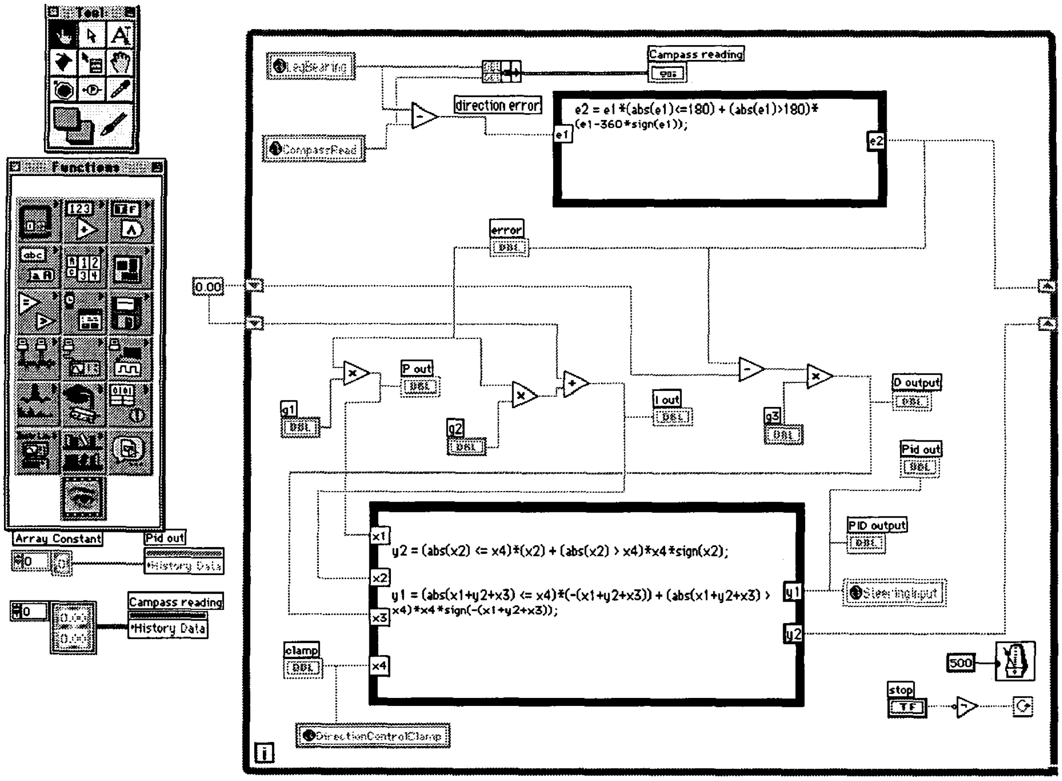

Figure 7 Diagram block (the graphical program) for BearingConroll.vi .The loop contains two formula nodes. The upper formula node compensates for the compass 0 to 360 degrees range and transforms the input bearing error to plus or minus 0 to 180 degrees input. The lower formula node calculates the PID components. The PID clamp is linked to the reference velocity to avoid the integral component to ramp up during the sonar scanning and recognition at the start of each leg.

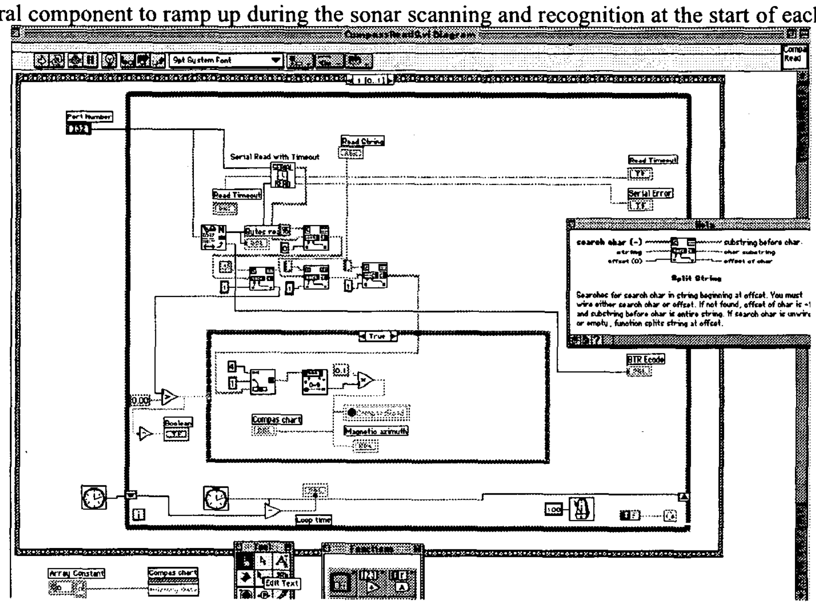

Figure 8 Reading the KVH compass and inclinometer from a data string via the serial port. If the VI encounters an incomplete string, it ignores the new reading until legal inputs arrives. The problem happens arbitrary as memory become overloaded due to multi tasking.

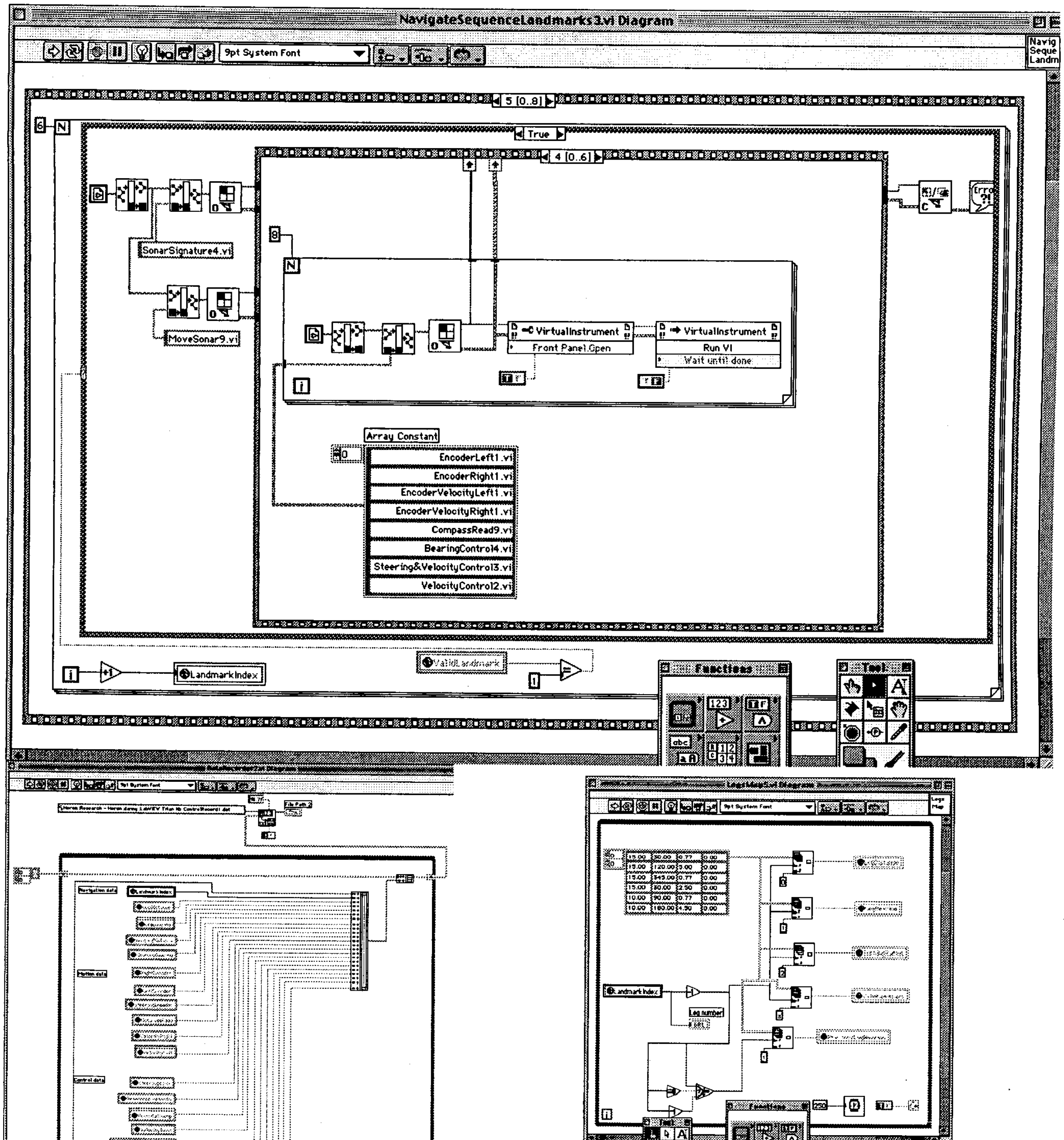

Figure 9 LabVIEW implementation for the navigation program (top). The program follows a sequence of legs using set of distances and magnetic bearings from a look up table (bottom/left). The table also provides a flag telling the program if the leg starts with a landmark which needs to be scanned and recognized or it is only an extra bridging leg to be traveled. There is a separation between motion sessions and recognition sessions. At the start of each leg, the program opens the 8 VIs responsible for the motion control. The 8 VIs are closed at the end of each leg. The only two VIs which are continuously running are the map VI and the data recorder VI (bottom/right - records 25 global parametersand found to be essential for analyzing the results of the experiment). After the 8 motion control VIs are opened, a distance control VI starts to control the velocity as a function of the traveled distance. The recognition session includes angular scanning of the area for an expected landmark with 51 steps of 1 degree. Each landmark is represented with a "personal signature" in the frequency domain. We call this "signature" an Acoustic Density Profile (ADP). The ADP retains the features which reflect the complex geometry of the landmark and its substance. Processing the data from the 51 "signatures" includes statistical signal processing for spectrum estimation and removing dummy signatures by thresholding techniques. The recognition and relocation of the robot is achieved by comparing the "landmark signature" to the 51 scanned "signatures". The relocation information is used by the program to correct the navigation (new distance and new magnetic bearing) for the next leg.

inclinometer data. The problem here was to avoid arbitrary erroneous data caused by momentary over-load surges on the memory resources.

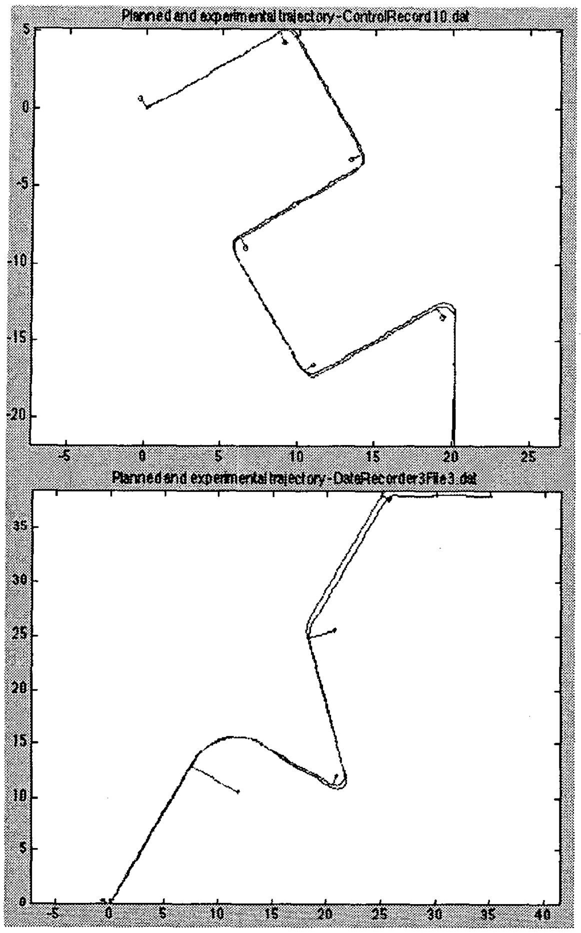

Figure 10 Effect of radius control. The experiments results compared to the planned trajectory (using Matlab simulation). Legs (in top plot) are 10 meters each. Top plot shows a 6 leg navigation with the same small 0.765 meter radius. Bottom plot shows a 5 leg navigation with the radius varying from 5 meter to 0.765 meter.

Figure 9 (upper part) shows the navigation VI for sequence of legs. Bottom/right shows the map VI and bottom/left shows the data logger VI for all Titan's parameters.

5 Experiments results

Figure 10 shows trajectory results from experiment recorded on a large cricket field. We found the distance control based on "distance to stop" algorithm works well with errors less than 0.25% of leg distance. The bearing control achieved very accurate direction following. The 90 degree turning angle between consecutive legs has less than 0.5 degree error. As seen from Figure 10 - the error between the planned trajectory to the actual one is very small - around 0.25 meter after 6 legs 10 meters each. The controllers were also tested at the farm (250 km trip from UOW) and showed good robustness while travelling at constant speed within confine areas fenced with crops and tackling bumpy ground.

6 Conclusions

Using LabVIEW with an industrial quality data acquisition cards turned to be a very effeienct way for robotics prototyping. Our navigation program manipulates concurrently more than 15 user defined VIs (and many more library defined VIs) - yet it is very modular and transparent. Our 3 years experience is very positive that such approach is suitable for designing complex mobile robots for research and teaching. During the development we used extensively the feature of LabVIEW portability over the Internet for technical consultations with NI national and global centers [Ratner & McKerrow,1999].

References

Pagett, J. M., 1966. "Steerable four wheel drive vehicle', U.S. Patent 5482125, Jan. 9, 1996.

Ratner D. and McKerrow, P.J. 1997. 'Landmark recognition with ultrasonic sensing for autonomous mobile robot navigation', FSR'97, December, Canberra, pp 552-558.

Ratner D. and McKerrow, P.J. 1998. 'Extracting Features from CTFM Sonar', in Chugani, M.L., Samat, A.R. and Cerna, M. eds, LabVIEW Signal Processing, Prentice Hall, ISBN 0-13-972449-4, pp 124- 131

McKerrow, P.J. 1999. 'Calibrating Mobile Robot Odometry with Ultrasonic Sensors', submitted to IEEE Transactions on Robotics and Automation.

Sekhavat, Svestka, Laumond and Overmars, Multilevel Path Planning for Nonholonomic Robots Using Semiholonomic Subsystems. The International Journal of Robotics Research, Volume 17 Number 8 August 1998, pages 840-857.

Harper N. and McKerrow, P.J. 1999. Recognition Plants with Ultrasonic Sensing for Mobile Robot Navigation, submitted to Eurobot'1999

Ratner D. and McKerrow, P.J. 1999. 'Dynamics of the Titan four-wheel drive mobile robot with floating Ackerman steering', ACRA'99, March, Brisbane, pages 144-149.

Kay L. (NZ) 1999. (http://www.sonicvision.co.nz/bat/)

Ratner D. and McKerrow, P.J. 1999. 'Advancing mobile robot research by using global engineering software tools with technical support over the Internet', FSR'99, August, Pittsburgh, pp 332-338.

Ratner D. and McKerrow, P.J. 2000. 'Navigation strategy for the Titan outdoor 4WD mobile robot using CTFM sonar guidance', ACRA'2000, August, Melbourne, Monash University.

Ratner D. and McKerrow, P.J. 2000. 'Recognition experiments with a narrow beam CTFM sonar mounted on the Titan outdoor 4WD mobile robot', SIRS'2000, July, Reading, Uni. of Reading.