

Hugh F. Durrant-Whyte

Department of Mechanical and Mechatronic Engineering
University of Sydney
Sydney NSW 2006, Australia

An Autonomous Guided Vehicle for Cargo Handling Applications

Abstract

This article describes an autonomous guided vehicle (AGV) system designed to transport ISO standard cargo containers in port environments. The AGV consists of a large 17.5-ton chassis, which is driven and steered through a diesel-hydraulic power set. The vehicle drives on pneumatic wheels over unprepared road surfaces at speeds up to 6 m/s. The navigation system is based on the use of millimeter-wave radar sensors detecting beacons placed at known locations in the environment. The navigation system reliably achieves accuracies of better than 3 cm. The planning and control systems allow the AGV to operate in the quay and yard area, to dock with gantry cranes, and to achieve a duty cycle comparable to that of a conventional manned vehicle. This article concentrates on describing three principal contributions of this work: the design of the overall vehicle and on-board systems architecture; the development of a navigation system appropriate for large, outdoor, high-speed vehicles; and the experimental program leading to validation of the vehicle system.

1. Introduction

This article describes the design of an autonomous guided vehicle (AGV) system able to transport ISO standard cargo containers in a port environment. A container terminal is, in many ways, an ideal environment to develop outdoor AGV technology. It is effectively a real version of "blocks-world," in which the blocks are standardized cargo containers, the manipulators are cranes, and the tasks consist of well-defined block moving and stacking operations. The principal difficulty in developing an AGV system for such an environment is simply to overcome problems of size, distance, and reliability in a commercially realistic manner.

There has been increasing interest in the development of outdoor AGV systems for a variety of applications. Notable is the work at Carnegie Mellon University

(Thorpe 1989; Thorpe et al. 1988), and VTT [Technical Research Centre of Finland] and the University of Helsinki in Finland (Halme and Koskinen 1993; Koskinen et al. 1993), both of which have looked at a number of applications in areas such as mining and agriculture. Much of this work has been concerned with the development of vehicles for rough terrain applications, using technology such as visual path following and Global Positioning System (GPS) based positioning for control and guidance. The many large programs in intelligent highway systems, such as PATH and Prometheus, are also having an impact on outdoor AGV technology, particularly in the design of sensing and control systems. The system described in this article differs substantially in both direction and technology from other outdoor vehicle and intelligent highway research. The vehicle described operates in a reasonably well-structured environment where accurate absolute position estimation and control are fundamental requirements. The vehicle is targeted specifically at a commercial-scale application that requires issues of autonomous reliability, integration with existing operation environments, and cost to be addressed from the outset.

In a conventional cargo container terminal, manned vehicles are used to transport containers from yard areas to the quay and back. In a typical medium-sized terminal executing 1 million box moves a year, there is a requirement for approximately 50 vehicles operating with 80% load factor in three shifts. The most important commercial criterion for a container terminal is the speed with which a ship can be unloaded and loaded. This in turn is limited by the efficiency with which the quay cranes are serviced by the vehicles. In attempting to improve ship turnaround, some of the largest container terminals have, for the past 7 years or more, been considering the possible use of AGV technology. There are three main operational reasons why AGV systems are considered of value: the ability to more precisely control the flow of container traffic in the terminal and particularly the quay area; the ability to reliably automate inventory control of containers in the terminal and so efficiently implement a required ship manifest; and the ability to reduce running

The International Journal of Robotics Research,
Vol. 15, No. 5, October 1996, pp. 407-440.
© 1996 Massachusetts Institute of Technology.

This work was conducted in the author's capacity as technical director of Firefly Ltd., North Devon, United Kingdom.

costs. In meeting these arguments, a cargo handling AGV must satisfy a number of functional requirements:

- *Size*: The AGV must have a capacity similar to that of a conventional manned vehicle and trailer. This requires a vehicle that resembles a conventional trailer, able to carry either a 40-foot or two 20-foot ISO containers with maximum load up to 60 tonnes.
- *Operation*: The AGV must be capable of achieving a duty cycle equivalent to or better than that of a manned vehicle. This means the vehicle must achieve speeds up to 6 m/s and first-time docking accuracy of better than 3 cm over the complete port area.
- *Environment*: The AGV must operate in the same environmental conditions as a manned vehicle. This requires operation both day and night in all weather conditions, and an ability to withstand both rough terrain driving and shocks from loading.
- *Reliability*: Reliability load factors call for system faults (requiring operator intervention) to be less than one in 100 hours of operation. System faults, particularly in sensing, navigation, and control systems must be minimized, and those that do occur must be identified.
- *Safety*: The vehicle must operate in areas where other equipment is used and ideally in areas that are manned. The vehicle therefore must be designed to operate in a fail-safe manner, and must be able to detect and respond to unexpected obstacles and other events.
- *Cost*: The capital cost of a unit vehicle system must be no more than 150% of the cost of a manned vehicle and when maintenance and manpower replacement costs are accounted for, must have a payback period of less than 2 years.¹

These requirements are generally met in the design described in this article. The most difficult issue to resolve was the problem of reliability. Defining and implementing quantifiable measures for reliability in commercial autonomous systems is, in practice, a very complex issue that remains unresolved.

The system described in this article represents an important step forward in the practical realization of large outdoor AGV systems. Elements of the vehicle design, including the mechanical chassis and control system, are based on conventional vehicle technologies. However, a number of other components of the system, including the hardware architecture and the navigation system, are considered to be significant advances in outdoor AGV technology. This article concentrates on describing those

elements of the AGV system that are considered to be important advances in guided vehicle technology.

In Section 2, the design of the overall vehicle system is described. The mechanical design of the vehicle is outlined, and the power and drive systems are briefly described. The kinematic properties of the vehicle and the equations governing the motion of the vehicle are introduced. These are subsequently used by the navigation and control systems for position estimation and trajectory tracking. The on-board system architecture is then described in detail. This architecture comprises navigation system, pilot, control system, and safety and condition monitoring systems. The issue of ensuring fail-safe and reliable operation in the design is discussed.

In Section 3, the vehicle navigation system is described. The navigation system is based on the use of millimeter-wave radar sensors to detect the range and bearing to a number of fixed known beacons located in the environment. The central navigation algorithm is an extended Kalman filter that exploits a model of the vehicle motion and radar observations to continuously provide estimates of the vehicle location. The radar system provides a solution to both the distance and accuracy requirements of the port environment. The Kalman filter algorithm provides a simple and direct means of combining this radar information to provide accurate and reliable estimates of vehicle location. The navigation system and the navigation radar are two of the most important contributions made in the work described in this article.

Section 4 briefly describes the task planning and trajectory control system. The task planning system allows the specification of high-level tasks consisting of container movements and docking operations. The trajectory generator then uses a map of the port and knowledge of vehicle kinematics to generate a sequence of control signals that will drive the vehicle along a desired path. The low-level vehicle control then simply compares the position estimates generated by the navigation system with the desired path to provide corrective drive signals.

In Section 5 the implementation and validation of the vehicle system are described. Attention is focused particularly on the operation of the navigation and control systems. The validation of these systems is developed primarily through an analysis of stationary position accuracy and dynamic covariance analysis. Some of the difficulties in producing a comprehensive validation suite for this type of outdoor AGV system are highlighted. The experience gained in applying this type of AGV technology in a commercial domain and the results from the control and navigation system trials are considered to be the most important contributions of this article.

The vehicle system described here was completed in October 1993 and has since been used in extensive user

1. This results in a target price of approximately \$200,000 at 1992 prices.



Fig. 1. Photograph of completed AGV with 40-foot container load. One of the two radars can be seen at the front of the vehicle below the main deck.

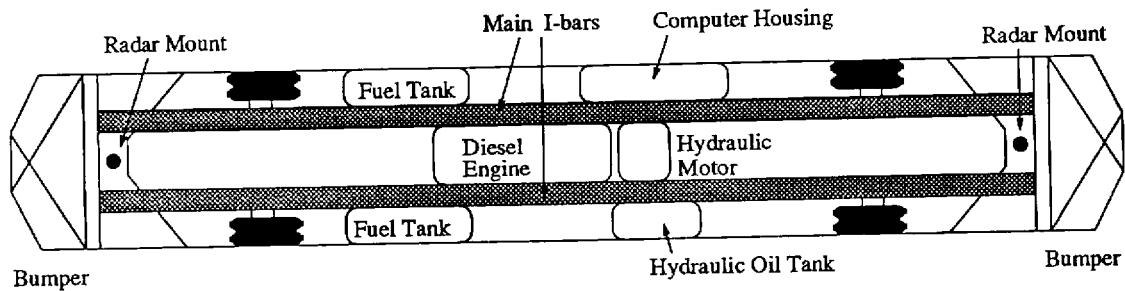


Fig. 2. Mechanical layout of main AGV chassis. The chassis is based around two structural I-bars running front to back. All main vehicle components are fixed to these bars. The front and rear axles are independently steered. The front right wheel and rear left wheel are driven, and the front left wheel and rear right wheel are instrumented with incremental encoders. The vehicle is thus diagonally symmetric and mechanically identical when traveling both forward and backward.

trials at an operating container terminal in the UK. By way of conclusion, some of the results of these trials and suggested future research directions essential to commercial exploitation of this type of large outdoor AGV technology are discussed.

2. The Vehicle System

2.1. Vehicle System

A photograph of the complete vehicle with load is shown in Figure 1. The essential mechanical layout of the ve-

hicle is shown in Figure 2. The vehicle is designed to carry either one ISO 40-foot container (with additional space for the new planned 50-foot containers), or two ISO 20-foot containers. This constrains the principal dimensions of the vehicle to resemble a conventional trailer. The main chassis is 15.5 m long, 2.9 m wide, and 1.6 m high. The distance between front and rear axles is 9 m, and each of the front and rear bumpers extend approximately 1.3 m from the main chassis. The main chassis has two structural I-bars running front to back. The loading platform, motor, and all other components are mounted from these

bars. The vehicle dry weight is 17.5 tonnes. In addition, the vehicle carries 1 tonne of diesel, 0.5 tonnes of hydraulic oil, and a maximum container load of 60 tonnes.

Each wheel has two conventional trailer tires and is independently pivoted around a steering axis. Each pair of front and rear wheels are mechanically linked and driven by a set of hydraulic rams with a maximum lock of 30°. The front and rear wheels can therefore be independently steered. This means that when the wheels are turned in opposite directions, the vehicle can execute short baseline curves, and when the wheels are turned in the same direction, the vehicle can move sideways without a change in orientation (crabbing). The front right wheel and the rear left wheel are powered. The front left wheel and rear right wheel are instrumented with incremental encoders. This means that the vehicle is diagonally symmetric and mechanically identical when traveling both forward and backward.

The vehicle drive and steer systems are diesel hydraulic. The choice of hydraulic over electrical was made purely on cost grounds, the hydraulic option being approximately half the cost of the equivalent diesel-electric generator set. A 168-kw constant speed diesel engine powers a hydraulic pump. The pump swash plate is used to control the volume flow rate and pressure of the hydraulic fluid (Viersma 1990). The hydraulic drive motors are mounted in the wheel hubs. This avoids the need for a drive shaft, a considerable cost saving. The drive motors are controlled using swash plates identical to those in the pump. The fact that the motors are driven from a common pump at equal pressure ensures that each wheel applies the same torque, regardless of rotational speed. Each steer axle is driven by a pair of conventional hydraulic rams. The rams are controlled by electro-mechanical valves. All low-level control is accomplished using proprietary PID control boards. The maximum full-load speed of the vehicle is restricted by the power of the diesel hydraulic unit to 6 m/s.

Mainly for reasons of cost, the vehicle has no suspension system, relying only on its tires and chassis flexibility for smooth rolling. However, the on-board computing system, which is housed in an environmentally sealed unit, is mechanically isolated from the main chassis by a local suspension system.

2.2. Kinematic Model

The kinematic model of the vehicle is central for both navigation and control of the AGV. At the speeds and loads required of the AGV, vehicle dynamics are also important. However, these have been ignored in the design of the navigation and control system because, when combined with the (substantial) dynamic effects

of the hydraulic drive train, the overall vehicle dynamics become very difficult to model with any accuracy (Viersma 1990). Dynamic effects are minimized by using an overpowered drive system, and a lookup table of maximum accelerations.

The kinematic arrangement of the AGV is shown in Figure 3. The vehicle has a wheelbase of length B and an axle length of W . The wheel steer angles and wheel velocities are denoted γ_{fl} , γ_{fr} , γ_{rl} , γ_{rr} , and V_{fl} , V_{fr} , V_{rl} , V_{rr} , for the front left, front right, rear left, and rear right wheels, respectively. As the wheels on each axle are mechanically linked, it is sufficient (kinematically) to consider each axle as comprising a single imaginary wheel placed along the centerline of the vehicle, the rear wheel having a steer angle γ_r and ground velocity V_r , and the front wheel having steer angle γ_f and ground velocity V_f . The steer angle γ_i and ground velocity V_i ($i = f, r$), of these imaginary wheels is related to the true steer angles and ground velocities by

$$\tan \gamma_i = \frac{B \tan \gamma_{il} - \tan \gamma_{ir}}{W \tan \gamma_{il} + \tan \gamma_{ir}}.$$

and

$$V_{il} \sin \gamma_{il} = V_{ir} \sin \gamma_{ir} = V_i \sin \gamma_i.$$

The vehicle location and orientation with respect to a fixed coordinate system is denoted by the vector $[x(t), y(t), \phi(t)]^T$. Although it is usual to reference all motions to the vehicle center, it is well known that for this AGV geometry, the vehicle center does not follow the path traveled by the imaginary center wheels, and indeed its velocity will change even when the wheel velocities remain constant. For this reason the kinematics and subsequent motion calculations are all referenced to the center of the front axle.

Subject to rolling and rigid body constraints, the equations of motion of this “double bicycle” type vehicle are well known (Alexander and Maddocks 1989) and are given by

$$\begin{aligned} \dot{x}(t) &= V_f \cos(\phi + \gamma_f) \\ \dot{y}(t) &= V_f \sin(\phi + \gamma_f) \\ \dot{\phi}(t) &= \frac{(V_f \sin \gamma_f - V_r \sin \gamma_r)}{B}. \end{aligned} \quad (1)$$

Simply, this states that the vehicle heading is dictated by the steer direction of the front wheels; the velocity, by the velocity of the front wheels; and the rate of change of orientation, by the difference in heading and speed of the front and rear of the vehicle. Equation (1) shows that the vehicle can execute three main types of motion: straight line motion with $\gamma_f = \gamma_r = 0$; a short baseline turn with $\gamma_f = -\gamma_r \neq 0$, in which both wheels turn to produce the same turn direction; and a sideways or crabbing motion

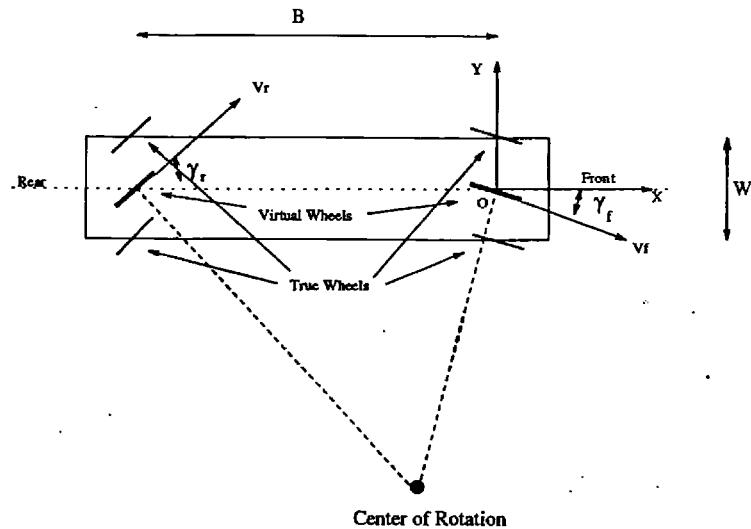


Fig. 3. AGV kinematic arrangement showing key dimensions, front and rear steer geometry, and "virtual" central wheels.

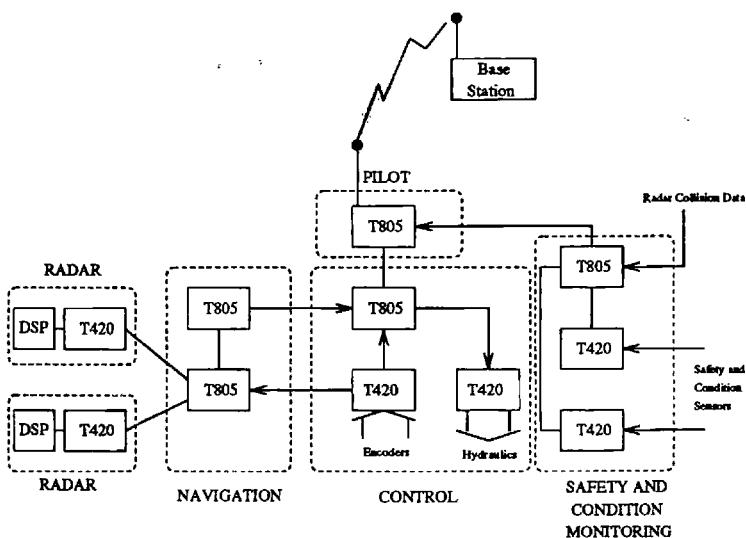


Fig. 4. Overall systems architecture showing base station, navigation, control, and safety and condition monitoring system.

with $\gamma_f = \gamma_r \neq 0$, in which the vehicle has a heading different from its orientation but in which the rate of change of orientation is zero. This last motion is valuable in vehicle path tracking.

2.3. System Architecture

Broadly, the AGV system architecture consists of five functional components: an off-board base station linked by a radio-data system to the vehicle, the navigation system, the pilot, the control system, and a safety and condition monitoring system. A functional diagram of the system architecture is shown in Figure 4.

The system architecture is based around a network of 11 Transputers. In addition, there is a dedicated digital signal processor for each of two radars, an on-board PC for vehicle monitoring, and an off-board PC to provide an interface for the tasking system. There were two main reasons for basing the design around Transputer technology. First, Transputers provide on-silicon facilities for multiple remote Transputers to communicate. Each Transputer has four high-speed (20 Mbits/s) bidirectional serial communication links that may be directly connected with no intervening hardware. Further, a one-chip solution is provided for connecting a link to a parallel port. This means that a large Transputer network, connected to a

variety of IO systems, may be constructed in an almost trivial manner. In practice, on the vehicle described here, remote links are implemented in optical fiber. A second critical advantage of Transputer technology in this application is the provision of explicit software support for parallel processing, on single and multiple processors, through message passing. Software support also exists to provide access to a real-time clock and functions for exact scheduling and timing of calculations. This makes both the design and implementation of software for the different on-board functions substantially simpler than would be the case if such parallel processing facilities were to be developed on a conventional bus-based architecture. The Transputer has two disadvantages: its cost, and the difficulty of integrating other types of processing units.

The base station PC simply provides an interface to the existing scheduling and port planning systems. It supports a radio-data link that allows communication to the vehicle. A typical task communicated to the vehicle consists of a required container motion with specified start and end docking conditions. The base station provides a simple display of vehicle location and status. Limited traffic management is also undertaken by the base station.

The on-board PC hosts the Transputer network and radio-data link. It provides a simple display of status, but takes no part in the operation of the vehicle. The on-board PC communicates directly to the pilot function. The pilot function is run on a single T805 (a 25-MHz floating point Transputer). It takes input from the navigation system and both input and output from the control system. The pilot has three main functions: it maintains a map of the port and a model of the vehicle, which allows it to generate a trajectory from a task specification; it sequences and sends drive and steer commands to the control function and monitors their execution; and it generates corrective control actions by comparing the estimated location output from the navigation system with the desired trajectory. All of these functions are described in Section 4 of this article.

The navigation system runs on four Transputers—two T420s (20-MHz integer Transputers) and two T805s—together with two DSP processors (Sharp LH911 family). The two T420s and the DSPs reside inside the radar housings, which are located at each end of the vehicle. Together these four processors pre-process the radar information to produce beacon and obstacle data. This information is communicated asynchronously to the two T805s which reside in the central computer housing near the center of the vehicle. These two processors implement the matching and Kalman filter algorithms, which produce the vehicle position estimate. The overall navigation system is treated as much as possible as a “black box.” It takes encoder information synchronously from

the control system and, after combining this with information from the radar sensors, outputs position estimates synchronously to the pilot. The navigation function is described in detail in Section 3 of this article.

The control system runs on three Transputers, one T805 and two T420s. The T805 takes drive sequences from the pilot and divides these into individual control signals for the steer and drive motors. These are derived from prespecified ramp sequences that ensure good tracking behavior from the hydraulic system. The T805 also takes care of the wheel kinematic relationships and generates velocity and steer angle information for the navigation system. One T420 sequences and writes output for the hydraulic drive system, and the other T420 reads in information from the encoders. The T805 reads in the encoder information from the second T420 and, by comparing this to the current demand output, generates an output sequence for the first T420. The output T420 drives a commercial PID board, which in turn controls hydraulic valves. The input T420 reads encoder information through a commercial encoder board.

The safety and condition monitoring system consists of two T420s (condition monitoring) and one T805 (safety). The T420s read in information from a large, special purpose interface card that receives information from 42 on-board sensors. These sensors measure such parameters as hydraulic pressure, hydraulic temperature, diesel speed, top-deck load, fuel levels, and bumper limit switches. Some 120 separate tests are performed on these data at each sample period. Each test is implemented as an independent process, all running in parallel on the two T420 Transputers. The tests determine the health and status of different aspects of the vehicle. The output of these processes is always classified into one of four values: warm-up, healthy, maintenance, or emergency stop. This value and the complete status may be accessed by the base station through the pilot. The two T420s comprise the condition monitoring system. The status value is used by the T805 safety computer to determine if the vehicle should halt in its path. In addition, the safety computer has two other inputs: one from the radars for long-range collision detection and one from the bumpers. The collision detection operation is simply implemented by clipping radar range plots with a map of the path of the AGV over the next 30 to 40 m of travel. The result is thresholded to provide a detection of an obstacle. The information from the bumpers comes from a limit switch. The bumpers deform under impact. It should be noted that the safety and condition monitoring system operates independently of the rest of the system architecture.

The integrity of the system in detecting faults, hardware failure, and software errors is achieved in the following way. The vehicle incorporates brakes that require power to be applied for release. Each Transputer has an

event pin. The system is designed so that the event pin of all Transputers in the network must be held high for the brake release to operate. A hardware fault or power loss will cause this pin to be taken low. The software running on each Transputer is also required to pulse (send an arbitrary status byte) to adjacent processors at a fixed frequency. This signal is termed a "heart beat." The absence of this heart beat causes the processor to again pull the event pin low. Thus, if the software fails to deliver the expected heart beat, or when a fault or software failure occurs, then the brakes will be released. This also ensures against a deadlock condition in the communication network, although it does not entirely eliminate the possibility of live-lock.

The architecture described proved to be an efficient means of developing the AGV hardware and software. In particular, it allowed each of the major components, navigation, control, and safety and condition monitoring system to be developed largely independently of each other. The use of parallel programming concepts also proved to be an efficient way of dealing with the simultaneous operation of different AGV functions. The message-passing paradigm was also useful in organizing and specifying software components. These points are demonstrated by the fact that the vehicle software consists of only 4000 lines of executable source code, which took only 12 man-months to write and test.

3. Navigation System

The AGV navigation system is based on the use of millimeter-wave radar sensors detecting the range and bearing to a number of fixed known beacons located in the environment. The central navigation algorithm is an extended Kalman filter that exploits a model of the vehicle motion and radar observations to continuously provide estimates of the vehicle location.

The use of millimeter-wave radar in robot navigation problems is still quite novel, although its use in numerous "intelligent vehicle highway systems" is now well advanced. The radar systems described here have been specifically developed to provide maximum positional accuracy with respect to a map of fixed beacons placed around the environment. The primary advantage of millimeter-wave radar in this type of application is its relatively high accuracy and long ranges in all weather conditions. A brief description of the radars used in the AGV navigation system is provided.

The use of the extended Kalman filter to process observations of beacons and to estimate the location of a vehicle is both well understood and widely used in the robotics community (Leonard and Durrant-Whyte 1992; Crowley 1989; Chatila and Laumond 1985). The central elements in the Kalman filter are the process model and

the observation model. The process model provides a description of how the system (vehicle location) evolves over time. The observation model provides a description of how the observations that are made are related to the state of the system of interest. The (extended) Kalman filter provides a means of recursively updating an estimate of the state on the basis of predictions made from the process model and observations made according to the observation model. Here the filter developed for this navigation system is described. Familiarity with the essential Kalman filter algorithm is assumed (see Maybeck [1979] and Bar-Shalom and Fortmann [1988] for detailed derivations). The main contribution of the system described in this article lies in the use of a new, and relatively sophisticated, process model describing the motion of a large vehicle, and in the incorporation of this with a novel sensing system.

3.1. Navigation Sensing

The choice of navigation sensor for use on the AGV was dictated by a need for high-speed, accurate, all-weather operation. A wire-guide or transponder-based system was initially considered, but this was rejected because of very poor ground conditions typical of outdoor operation, resulting in motion and damage of ground-mounted navigation aids. Differential GPS was rejected as being too inaccurate, unreliable, and subject to obscuration by typical port structures. Laser (bearing only) navigation systems were tested and, although reliable and accurate, were found to have both too short a range and also to suffer unacceptable attenuation in bad weather conditions. Vision was also rejected as being too expensive and unreliable, particularly in heavy rain and fog. A number of radar systems were considered. These included a 4-GHz local triangulation system and a 10-GHz time-of-flight microwave system. Although these systems have more than enough range and potential accuracy, at these frequencies the large quantity of metal (containers, cranes, etc.) in the port caused severe distortion of radar signals.

After a substantial test program, millimeter-wave radar (MMWR) was chosen as the primary navigation sensor. The advantage of MMWR over other types of sensors in this application is that relatively long-range measurements can be made with high accuracy in all weather conditions. MMWR has substantial accuracy advantages over microwave systems, but has longer ranges and does not suffer from as much atmospheric attenuation as lasers or other optical systems. The MMWR chosen was also found to be largely immune to the large amount of clutter in a port environment because of its relatively narrow beam width and short wavelength. Although current MMWR systems are both difficult to procure and are at a substantial cost disadvantage with respect to many

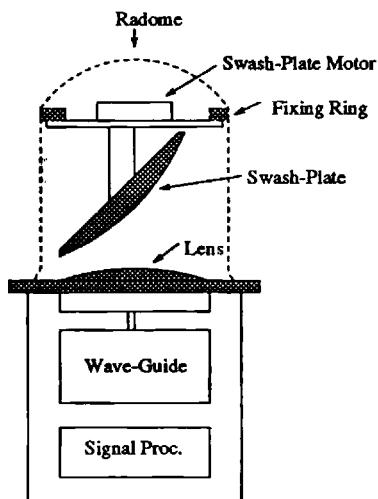


Fig. 5. Schematic showing arrangement of radar components. The wave guide generates a vertical beam that is scanned in azimuth by a rotating 45° swash plate.

other sensors, this should change as their use in the automotive industry becomes more commonplace.

The vehicle employs two MMWR units as the primary navigation sensors. These units are mounted at the front and rear of the vehicle below the main cargo deck, at a height of approximately 1.2 m from the ground, as shown in Figures 1 and 2. The housing for the radars is cut away at the sides of the vehicle to provide each radar with a field of view of approximately 270°. This two-sensor configuration, rather than one sensor mounted high up, is employed because the radars are also used for collision detection. The radar units were specifically designed for this AGV system.

Each unit consists of a wave guide and electronic assembly transmitting a beam vertically upward through a lens antenna (Fig. 5). The vertical beam is reflected off a 45° swash plate, mounted above the antenna, into a horizontal beam. As the swash plate rotates, so the beam is scanned in azimuth. The swash plate scan rate is approximately 6 Hz. The swash plate is instrumented with an encoder to provide measurements of beam bearing. The swash plate and antenna are housed in a half-wavelength Teflon radome. The two radars are synchronized so that when one is pointing directly ahead, the other is pointing directly into the vehicle. This ensures that a continuous stream of data is available from the radar pair.

The radar units are 77-GHz (European automotive band) frequency modulated continuous wave (FMCW, or chirp) devices (see Appendix). They are based on conventional E-plane wave guide technology. The radars transmit left circular polarized radiation but detect both left and right circular polarizations. This allows

the use of polarization information to identify objects and reject ground clutter. The radars have a swept bandwidth of 600 MHz and a sweep time (pulse duration) of 500 μ s. This fixes the range resolution of the radar at 25 cm, which may be increased through a super-resolution algorithm to approximately 10 cm. The radars each have a quasi-optic lens antenna providing 42 dB of gain, resulting in a beam width of less than 1°. The swash plate incorporates a slight curvature (a spoiler), which broadens the elevation beam to approximately 6°. This allows for dynamic variations in attitude of the vehicle ("bounce"). The radar transmits approximately 10 mW of power. The maximum practical range of the radars is restricted to approximately 200 m.

Figure 21 (in the Appendix) shows a complete range plot out to 150 m obtained from one pulse. The signal processing unit for the radar needs to compute these range plots in the pulse time (500 μ s). To compute a valid range plot, two 1-K FFTs (one for each polarization channel), associated windowing, magnitude squared, and thresholding operations must be performed. This requires substantial processing hardware and necessitates the use of a very high performance DSP unit.

The radar units provide range information from the chirp signal. Bearing information is obtained from encoder measurements of the swash plate. The navigation system works by detecting the range and bearing to a set of beacons placed at known, mapped locations about the environment. These beacons are radar trihedrals (effectively, internal corner reflectors). The trihedrals return an incident radiation beam directly back along the incident path over the capture angle of the trihedral. The trihedrals employed in the navigation system described have a physical area of 20 cm² but a radar cross section (apparent area to the radar) of 100 m². The beacons are covered with a special polarizing grid. This allows beacons to be uniquely distinguished from other background objects in the environment by the particular polarization of the reflected radiation. For navigation purposes, the observations made by the radar and fed to the navigation system consist of both range and bearing measurements to these beacons. Missing or obscured beacons, clutter, and false detections are common, but these are dealt with by the navigation algorithm and not through sensor data preprocessing.

The radar is also used as a collision detection sensor. In this case, the cross-polarized pulse return is recorded out to 35 m and thresholded to provide an indication of first reflection from a target. A complete 360° collision scan is performed at 6 Hz as the radar swash plate rotates. Figure 6 shows a typical polar plot of collision data returned by the radar. This is subsequently clipped to show only those obstacles in the AGV path.

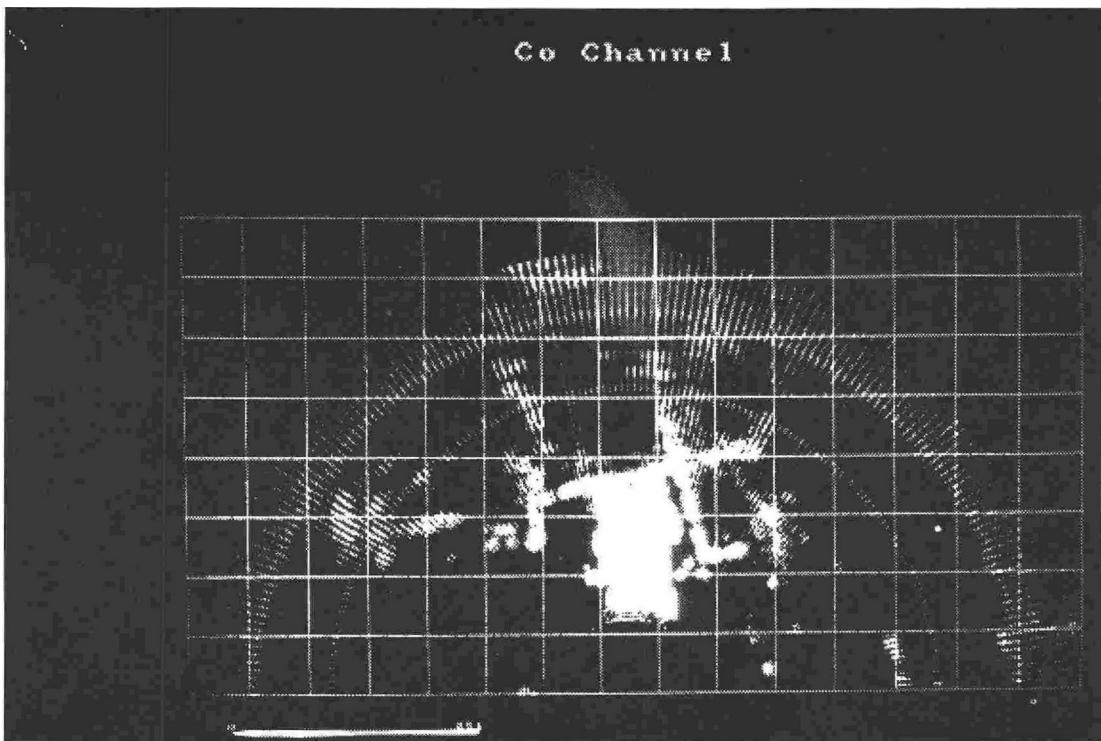


Fig. 6. A polar plot of collision data from the radar to a distance of 35 m. The regular plane surfaces are container sides. The signals at the lower left are traffic cones.

3.2. The Navigation Algorithm

The navigation algorithm is based on the extended Kalman filter. The algorithm requires that both a *process model* and an *observation model* be defined. The process model describes the motion of the vehicle together with any other parameters of interest. The observation model describes how the observations that are made are related to the location of the vehicle. Given these models, the (extended) Kalman filter provides a recursive procedure for computing a minimum mean squared error estimate of the vehicle location on the basis of predictions made using the process model and measurements made according to the observation model. The Kalman filter algorithm is a widely used algorithm both in the robotics community and in many other areas. In the following, familiarity with the details of the Kalman filter and the assumptions necessary for its implementation are assumed. This section concentrates on developing the process and observation models appropriate for the AGV system described. The kinematic model of the vehicle developed in Section 2 is used to develop both the nominal prediction and error transfer equations for the process model. The observation model is introduced, and a procedure for matching observations to the beacon map is described. Finally, it is

shown how the observations are used to update vehicle location.

3.2.1. The Process Model

The process model is fundamentally important to the correct operation of the navigation filter. The process model describes the transition of the vehicle state over time. It provides essential information for predicting vehicle location and observing unmeasured vehicle parameters. The process model consists of a description of the nominal motion of the vehicle and a model of the way in which uncertainties arise in this prediction. The process model described in this section represents a significant improvement on previously published models and is appropriate for vehicles traveling at relatively high speeds on pneumatic tires over rough terrain.

The error model developed for the vehicle makes explicit the fact that vehicle location uncertainty is predominantly caused by errors in drive velocity (slipping), steer traction (skidding), and changes in wheel radius. Further, the model explicitly accounts for the fact that these errors vary with vehicle state. As a consequence the process error model does not follow the normal assumption

used in the Kalman filter literature of being linearly additive. For this reason, the derivation of the process model and corresponding error model is accomplished in three stages. First a “nominal” process model is developed that describes vehicle motion in the absence of control, modeling, and state errors. Second, this model is used to see how errors in velocity, steer, wheel radius, and previous state values are propagated through time. Finally, a linearized model of the error propagation equations are used to provide equations for state estimate covariance propagation.

A nominal vehicle model described in continuous time by equation (1) is assumed. An additional assumption is made that the front and rear wheels travel at approximately the same velocity given by the product of an average wheel angular speed $\omega(t)$ with an average wheel radius $R(t)$ so that

$$R(t)\omega(t) = V_f(t) = V_r(t).$$

Equation (1) then becomes

$$\begin{aligned}\dot{x}(t) &= R(t)\omega(t) \cos(\phi(t) + \gamma_f(t)) \\ \dot{y}(t) &= R(t)\omega(t) \sin(\phi(t) + \gamma_f(t)) \\ \dot{\phi}(t) &= \frac{R(t)\omega(t)}{B} (\sin \gamma_f(t) - \sin \gamma_r(t)) \\ \dot{R}(t) &= 0,\end{aligned}\quad (2)$$

where changes in wheel radius $R(t)$ are made explicit.

The continuous-time model of equation (2) is next converted into a discrete-time state transition equation. A synchronous sampling interval ΔT is assumed, all derivatives are approximated by first-order forward differences, all control signals ($\omega(t)$, $\gamma_f(t)$ and $\gamma_r(t)$) are assumed approximately constant over the sample period, and all continuous times are replaced with a discrete-time index as $t = k\Delta T \stackrel{\triangle}{=} k$. Equation (2) then becomes

$$\begin{aligned}x(k+1) &= x(k) + \Delta T R(k)\omega(k) \cos [\phi(k) + \gamma_f(k)] \\ y(k+1) &= y(k) + \Delta T R(k)\omega(k) \sin [\phi(k) + \gamma_f(k)] \\ \phi(k+1) &= \phi(k) + \Delta T \frac{R(k)\omega(k)}{B} [\sin \gamma_f(k) - \sin \gamma_r(k)] \\ R(k+1) &= R(k).\end{aligned}\quad (3)$$

The state vector at a time k is then defined as

$$\mathbf{x}(k) = [x(k), y(k), \phi(k), R(k)]^T,$$

the control vector as

$$\mathbf{u}(k) = [\omega(k), \gamma_f(k), \gamma_r(k)]^T,$$

and the nominal (error-free) state transition as

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)), \quad (4)$$

where the transition function $\mathbf{f}(\cdot)$ is defined in equation (3). This defines the nominal process model.

During operation, the true vehicle state $\mathbf{x}(k)$ will never be known. Instead, an estimate of the state is computed through beacon observation and knowledge of the drive signals. Following the notation used in Gelb (1974), let

$$\hat{\mathbf{x}}^+(k) = [\hat{x}^+(k), \hat{y}^+(k), \hat{\phi}^+(k), \hat{R}^+(k)]^T$$

denote the estimate made of the state $\mathbf{x}(k)$ at time k based on all observations and control knowledge up to time k . Further, let

$$\bar{\mathbf{u}}(k) = [\bar{\omega}(k), \bar{\gamma}_f(k), \bar{\gamma}_r(k)]^T$$

be the mean measured (from encoders) value of the true control vector $\mathbf{u}(k)$. Following standard practice in the development of the extended Kalman filter (Bar-Shalom and Fortmann 1988), we use equation (4) to generate a prediction $\hat{\mathbf{x}}^-(k+1)$,

$$\begin{aligned}\hat{\mathbf{x}}^-(k+1) &= [\hat{x}^-(k+1), \hat{y}^-(k+1), \\ &\quad \hat{\phi}^-(k+1), \hat{R}^-(k+1)]^T,\end{aligned}$$

of the true state $\mathbf{x}(k+1)$ at time $k+1$ as

$$\hat{\mathbf{x}}^-(k+1) = \mathbf{f}(\hat{\mathbf{x}}^+(k), \bar{\mathbf{u}}(k)). \quad (5)$$

The prediction stage of the extended Kalman filter also requires that the manner in which errors are injected into the system and the way in which these errors are propagated in time be accurately modeled. Errors are injected into the AGV system by three primary sources: forward drive signals, steer angle signals, and changes in wheel radius. The forward drive error is modeled as a combination of additive disturbance error $\delta\omega(k)$ and multiplicative slip error $\delta q(k)$:

$$\omega(k) = \bar{\omega}(k)[1 + \delta q(k)] + \delta\omega(k),$$

where $\bar{\omega}(k)$ is taken to be the mean measured wheel rotation rate as recorded by the vehicle encoders, and $\omega(k)$ is defined to be the true mean wheel rotation rate defined through equation (3). The steer drive error is similarly modeled as a combination of an additive disturbance error $\delta\gamma(k)$ and a multiplicative skid error $\delta s(k)$, assumed to be from an identical source for both front and rear axles

$$\begin{aligned}\gamma_f(k) &= \bar{\gamma}_f(k)[1 + \delta s(k)] + \delta\gamma(k), \\ \gamma_r(k) &= \bar{\gamma}_r(k)[1 + \delta s(k)] + \delta\gamma(k).\end{aligned}$$

where $\bar{\gamma}_f(k)$ and $\bar{\gamma}_r(k)$ are taken to be the mean measured axle steer angles as recorded by the steer encoders, with $\gamma_f(k)$ and $\gamma_r(k)$ defined to be the true mean axle steer angles defined through equation (3). The error in

wheel radius is modeled as a discrete additive disturbance rate error (a random walk) so that

$$R(k) = \hat{R}^+(k) + \Delta T \delta R(k).$$

The source errors $\delta q(k)$, $\delta \omega(k)$, $\delta s(k)$, $\delta \gamma(k)$, and $\delta R(k)$ are modeled as constant, zero mean, uncorrelated white sequences, with variances σ_q^2 , σ_ω^2 , σ_s^2 , σ_γ^2 , and σ_R^2 , respectively.

The error models for forward drive and steer signals are designed to reflect two important features. First, the multiplicative component of the error models reflects the increased uncertainty in vehicle motion as speed and steer angles increase (slipping and skidding).² Second, the additive component of the error models is designed to reflect both stationary uncertainty and motion model errors such as axle offsets. The additive error is also important to stabilize the estimator algorithm. The random walk model for wheel radius is intended to allow adaptation of the estimator to wheel radius changes caused by uneven terrain and by changes in vehicle load (the chassis and wheels deform substantially at loads of 30 to 60 tons). In theory, such changes are best modeled as abrupt variations in parameter values. In practice such models turned out to be very difficult to implement. The random walk model was found to accommodate typical variations in a sensible manner.

To develop the error model, it is necessary to consider how errors in the state estimate and knowledge of control input feed through equation (3) to generate prediction errors. The error between the true state and estimated state, and between the true state and the prediction, are given by

$$\delta \mathbf{x}^+(k) = \mathbf{x}(k) - \hat{\mathbf{x}}^+(k),$$

and

$$\delta \mathbf{x}^-(k+1) = \mathbf{x}(k+1) - \hat{\mathbf{x}}^-(k+1), \quad (6)$$

respectively, and the difference between true and measured control input is denoted by

$$\delta \mathbf{u}(k) = \mathbf{u}(k) - \bar{\mathbf{u}}(k). \quad (7)$$

With these definitions, together with equations (4) and (5), we have

$$\begin{aligned} \delta \hat{\mathbf{x}}^-(k+1) &= \mathbf{x}(k+1) - \hat{\mathbf{x}}^-(k+1) \\ &= \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)) - \mathbf{f}(\hat{\mathbf{x}}^+(k), \bar{\mathbf{u}}(k)) \\ &= \mathbf{f}(\hat{\mathbf{x}}^+(k) + \delta \mathbf{x}^+(k), \bar{\mathbf{u}}(k) + \delta \mathbf{u}(k)) \\ &\quad - \mathbf{f}(\hat{\mathbf{x}}^+(k), \bar{\mathbf{u}}(k)). \end{aligned} \quad (8)$$

2. Fundamentally, these errors are actually due to linear and rotational inertial forces acting at the interface between tire and road. A model that explicitly incorporates these effects is described in Julier and Durrant-Whyte (1995).

Evaluating this expression using equation (3), and neglecting all second-order error products, we obtain an error transfer model as

$$\begin{aligned} \delta \hat{x}^-(k+1) &= \delta \hat{x}^+(k) + \Delta T \cos(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ &\quad \times [\delta \Omega(k) + \bar{\omega}(k) \hat{R}^+(k)] \\ &\quad - \Delta T \sin(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ &\quad \times [\delta \Gamma(k) + \hat{R}^+(k) \bar{\omega}(k) \delta \phi(k)] \\ \delta \hat{y}^-(k+1) &= \delta \hat{y}^+(k) + \Delta T \sin(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ &\quad \times [\delta \Omega(k) + \bar{\omega}(k) \hat{R}^+(k)] \\ &\quad + \Delta T \cos(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ &\quad \times [\delta \Gamma(k) + \hat{R}^+(k) \bar{\omega}(k) \delta \phi(k)] \\ \delta \hat{\phi}^-(k+1) &= \delta \hat{\phi}^+(k) + \Delta T \frac{\sin \bar{\gamma}_f(k) - \sin \bar{\gamma}_r(k)}{B} \\ &\quad \times [\delta \Omega(k) + \bar{\omega}(k) \hat{R}^+(k)] \\ &\quad + \Delta T \frac{\cos \bar{\gamma}_f(k) - \cos \bar{\gamma}_r(k)}{B} \delta \Gamma(k) \\ \delta \hat{R}^-(k+1) &= \delta \hat{R}^+(k) + \Delta T \delta R(k) \end{aligned} \quad (9)$$

where

$$\delta \Omega(k) = \hat{R}^+(k) \bar{\omega}(k) \delta q(k) + \hat{R}^+(k) \delta \omega(k) \quad (10)$$

is the composite longitudinal rate error describing control-induced error propagation along the direction of travel, and

$$\delta \Gamma(k) = \hat{R}^+(k) \bar{\omega}(k) (\bar{\gamma}_f(k) - \bar{\gamma}_r(k)) \delta s(k) + \hat{R}^+(k) \bar{\omega}(k) \delta \gamma(k) \quad (11)$$

is the composite lateral rate error describing control-induced error propagation perpendicular to the direction of travel.

These error transfer equations may be written in state-vector notation as follows:

Define the state error transfer matrix as

$$\mathbf{F}(k) = \begin{bmatrix} 1 & 0 & -\Delta T \hat{R}^+(k) \bar{\omega}(k) & \Delta T \bar{\omega}(k) \cos(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ 0 & 1 & \Delta T \hat{R}^+(k) \bar{\omega}(k) & \Delta T \bar{\omega}(k) \sin(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) \\ 0 & 0 & 1 & \Delta T \bar{\omega}(k) \frac{\sin \bar{\gamma}_f(k) - \sin \bar{\gamma}_r(k)}{B} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

and the source error transfer matrix as

$$\mathbf{G}(k) = \begin{bmatrix} \cos(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) & -\sin(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) & 0 \\ \sin(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) & \cos(\hat{\phi}^+(k) + \bar{\gamma}_f(k)) & 0 \\ \sin \bar{\gamma}_f(k) & \cos \bar{\gamma}_f(k) & 0 \\ \frac{-\sin \bar{\gamma}_r(k)}{B} & \frac{-\cos \bar{\gamma}_r(k)}{B} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

With $\delta\mathbf{w}(k) = [\delta\Omega(k), \delta\Gamma(k), \deltaR(k)]^T$, the error transfer equations now become

$$\delta\mathbf{x}^-(k+1) = \mathbf{F}(k)\delta\mathbf{x}^+(k) + \Delta T\mathbf{G}(k)\delta\mathbf{w}(k). \quad (12)$$

Define

$$\begin{aligned}\mathbf{P}^-(k+1) &= E[\delta\mathbf{x}^-(k+1)\delta\mathbf{x}^-(k+1)^T], \\ \mathbf{P}^+(k) &= E[\delta\mathbf{x}^+(k)\delta\mathbf{x}^+(k)^T], \\ \Sigma_k &= E[\delta\mathbf{w}(k)\delta\mathbf{w}(k)^T],\end{aligned}$$

and assume that $E[\delta\mathbf{x}^+(k)\delta\mathbf{w}(k)^T] = \mathbf{0}$. Squaring equation (12) and taking expectations then gives an equation for the propagation of covariance information as

$$\mathbf{P}^-(k+1) = \mathbf{F}(k)\mathbf{P}^+(k)\mathbf{F}^T(k) + \Delta T^2\mathbf{G}(k)\Sigma(k)\mathbf{G}^T(k), \quad (13)$$

where, on the assumption that the source errors are uncorrelated, $\Sigma(k)$ is given by

$$\Sigma(k) = \begin{bmatrix} [\hat{R}^+(k)]^2 & \mathbf{0} & 0 \\ \times([\bar{\omega}(k)]^2\sigma_q^2 + \sigma_{\omega}^2) & & \\ 0 & [\hat{R}^+(k)\bar{\omega}(k)]^2 & 0 \\ & \times((\bar{\gamma}_f(k) - \bar{\gamma}_r(k))^2\sigma_s^2 + \sigma_{\gamma}^2) & \\ 0 & \mathbf{0} & \sigma_R^2 \end{bmatrix} \quad (14)$$

3.2.2. Observation Model

After predicting the vehicle location, the next step in the navigation process is to take an observation and to combine this information together with the prediction to produce an updated estimate of the vehicle location. The essential observation information used by the radar consists of measurements of range and bearing made by the radar units to a number of beacons placed at fixed and known locations in the environment. The basic layout of the observation process is shown in Figure 7.

Processing of observations occurs in four stages.

1. The measurement is converted into a Cartesian observation referenced to the vehicle coordinate system.
2. The vehicle-centered observation is transformed into base coordinates using knowledge of the predicted vehicle location at the time the observation was obtained.
3. The observation is then matched to a map of beacons maintained by the AGV in base coordinates.
4. The matched beacon is transformed back into a vehicle-centered coordinate system where it is used

to update vehicle location according to the standard extended Kalman filter equations.

In the following, we assume that the measurements are taken at a discrete-time instant k when a prediction of vehicle location is already available. The true asynchronicity of the measurement and prediction cycle is considered in a later section. We further make extensive use of the standard result from the literature on the extended Kalman filter that if two random variables \mathbf{a} and \mathbf{b} are related by the nonlinear equation $\mathbf{a} = \mathbf{g}(\mathbf{b})$, then the mean $\bar{\mathbf{a}}$ of \mathbf{a} may be approximated in terms of the mean $\bar{\mathbf{b}}$ of \mathbf{b} by

$$\bar{\mathbf{a}} = \mathbf{g}(\bar{\mathbf{b}}) \quad (15)$$

and that the variance Σ_a of \mathbf{a} may be approximated in terms of the variance Σ_b of \mathbf{b} by

$$\Sigma_a = \nabla \mathbf{g}_b \Sigma_b \nabla \mathbf{g}_b^T, \quad (16)$$

where $\nabla \mathbf{g}_b$ is the Jacobian of $\mathbf{g}(\cdot)$ taken with respect to \mathbf{b} , evaluated at the mean $\bar{\mathbf{b}}$.

We now detail these four processing stages.

Stage 1 The radar provides observations of range $r(k)$ and bearing $\theta(k)$ to a fixed target in the environment, as shown in Figure 7. The radar itself is located on the centerline of the vehicle with a longitudinal offset d from the vehicle-centered coordinate system. The observations $\mathbf{z}_v(k)$, in Cartesian coordinates and referred to the vehicle frame, are given by

$$\mathbf{z}_v(k) = \begin{bmatrix} z_{xv}(k) \\ z_{yv}(k) \end{bmatrix} = \begin{bmatrix} d + r(k) \cos \theta(k) \\ r(k) \sin \theta(k) \end{bmatrix}. \quad (17)$$

We assume that the errors in range and bearing may be modeled as a Gaussian uncorrelated white sequence with constant variances σ_r^2 and σ_θ^2 , respectively. The observation variance $\Sigma_z(k)$ in vehicle coordinates may be obtained by applying equation (16) to equation (17) to obtain

$$\Sigma_z(k) = \begin{bmatrix} \cos \theta(k) & -\sin \theta(k) \\ \sin \theta(k) & \cos \theta(k) \end{bmatrix} \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & r^2 \sigma_\theta^2 \end{bmatrix} \begin{bmatrix} \cos \theta(k) & \sin \theta(k) \\ -\sin \theta(k) & \cos \theta(k) \end{bmatrix} \quad (18)$$

Stage 2 The observation, defined in vehicle coordinates, now needs to be transformed into absolute world coordinates so that it can be matched to the map of beacon locations. The map of beacon locations must be maintained in a fixed coordinate system; otherwise, every element of the map would have to be transformed into vehicle coordinates before a decision on a best match

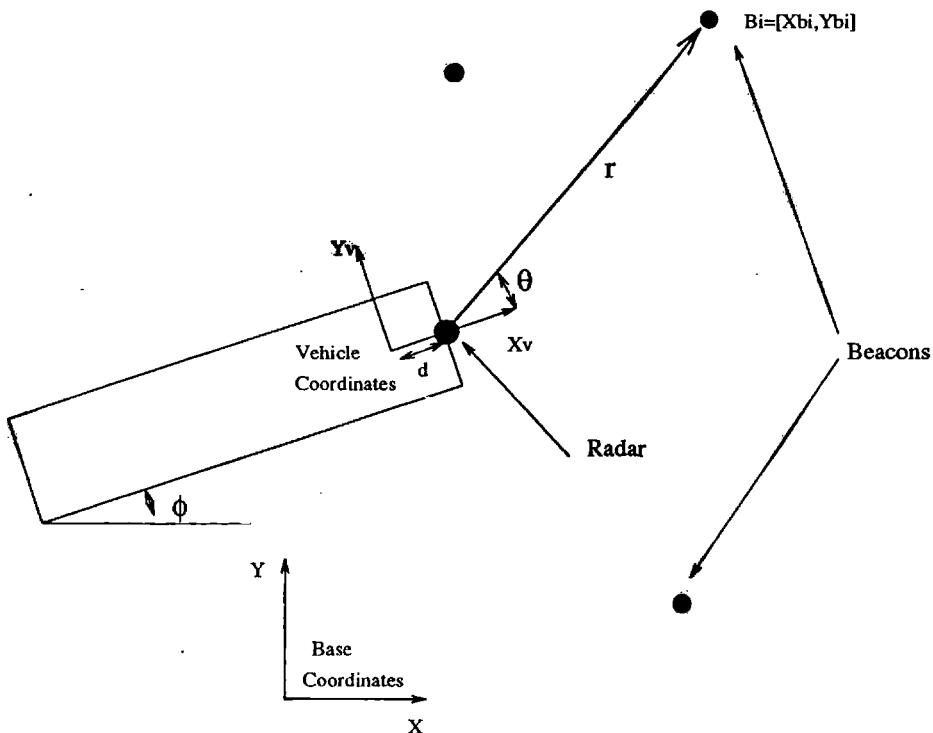


Fig. 7. Vehicle and beacon layout defining the observation model.

could be obtained. If the current predicted vehicle location in base coordinates is $[\hat{x}^-(k), \hat{y}^-(k), \hat{\phi}^-(k)]^T$, then from Figure 7, the observation $\mathbf{z}_b(k)$ in Cartesian base coordinates is given by

$$\begin{aligned}\mathbf{z}_b(k) &= \begin{bmatrix} z_{xb}(k) \\ z_{yb}(k) \end{bmatrix} \\ &= \begin{bmatrix} \hat{x}^-(k) + z_{xv}(k) \cos \hat{\phi}^-(k) - z_{yv}(k) \sin \hat{\phi}^-(k) \\ \hat{y}^-(k) + z_{xv}(k) \sin \hat{\phi}^-(k) + z_{yv}(k) \cos \hat{\phi}^-(k) \end{bmatrix} \quad (19)\end{aligned}$$

The observation variance may also be transformed into base coordinates by applying equation (16) to equation (19) (with respect to both $\mathbf{x}(k)$ and $\mathbf{z}_v(k)$), to give

$$\Sigma_b(k) = \mathbf{T}_x(k) \mathbf{P}^-(k) \mathbf{T}_x^T(k) + \mathbf{T}_z(k) \Sigma_z(k) \mathbf{T}_z^T(k), \quad (20)$$

where

$$\mathbf{T}_x(k) = \begin{bmatrix} 1 & 0 & -z_{xv}(k) \sin \hat{\phi}^-(k) & 0 \\ 0 & 1 & -z_{yv}(k) \cos \hat{\phi}^-(k) & 0 \\ 0 & 1 & -z_{xv}(k) \cos \hat{\phi}^-(k) & 0 \\ 0 & 1 & -z_{yv}(k) \sin \hat{\phi}^-(k) & 0 \end{bmatrix}, \quad (21)$$

and

$$\mathbf{T}_z(k) = \begin{bmatrix} \cos \hat{\phi}^-(k) & -\sin \hat{\phi}^-(k) \\ \sin \hat{\phi}^-(k) & \cos \hat{\phi}^-(k) \end{bmatrix}, \quad (22)$$

and where $\mathbf{P}^-(k)$ is the predicted vehicle state covariance.

Stage 3 With the observation and observation covariance now in base coordinates, we are in a position to match the measurement with the beacon map. In the AGV application, the beacons are surveyed into the site using a theodolite, and so it is assumed that no error is associated with their location. The beacon locations $\mathbf{b}_i = [x_{bi}, y_{bi}], i = 1, \dots, N$ are loaded into the AGV at run time. To limit search time in the map, the beacons are stored as a pair of ordered lists in x and y coordinates. The maximum eigenvalue λ_b of the symmetric square root of $\Sigma_b(k)$ is computed, and these two lists of beacons are checked to find a beacon (x_{bi}, y_{bi}) that satisfies

$$[x_{bi} - \lambda_b < z_{xb}(k) < x_{bi} + \lambda_b]$$

and

$$[y_{bi} - \lambda_b < z_{yb}(k) < y_{bi} + \lambda_b]$$

If no beacons are found that satisfy this test, then the observation is discarded. For those beacons that pass the test, a simple gate is used as the matching criterion:

$$(\mathbf{b}_i - \mathbf{z}_b(k))^T \Sigma_b^{-1}(k) (\mathbf{b}_i - \mathbf{z}_b(k)) < \alpha. \quad (23)$$

If only one beacon passes both tests, then the update algorithm continues to stage 4. However, if more than one beacon passes the test, then an error is declared. This is to avoid the possibility of generating an incorrect association, which can have catastrophic effects on the

location estimated and on subsequent vehicle control. The gate size α is normally taken to be quite small (0.5). This is because the radar, particularly in an environment consisting of large metal containers, generates a large number of false alarms (as many as 80% of readings).

Stage 4 Once a correct association with a single beacon $\mathbf{b} = [x_b, y_b]^T$ has been made, a vehicle-centered observation prediction must be generated so that the vehicle location may be updated. This update is not done in base coordinates because the location estimates become very sensitive to approximations in orientation uncertainty in this coordinate system. Vehicle-centered observations can be generated by transforming the matched beacon from base coordinates back through the predicted vehicle location to the estimated vehicle-centered coordinate system according to

$$\hat{\mathbf{z}}_v = \begin{bmatrix} \hat{z}_{vx} \\ \hat{z}_{vy} \end{bmatrix} = \begin{bmatrix} \cos \hat{\phi}^-(k) & \sin \hat{\phi}^-(k) \\ -\sin \hat{\phi}^-(k) & \cos \hat{\phi}^-(k) \end{bmatrix} \begin{bmatrix} x_b - \hat{x}^-(k) \\ y_b - \hat{y}^-(k) \end{bmatrix}. \quad (24)$$

3.2.3. Vehicle Update

The vehicle estimate may now be updated using the usual Kalman filter equations. The updated state estimate is computed from

$$\hat{\mathbf{x}}^+(k) = \hat{\mathbf{x}}^-(k) + \mathbf{W}(k)[\mathbf{z}_v(k) - \hat{\mathbf{z}}_v], \quad (25)$$

and the covariance from

$$\mathbf{P}^+(k) = \mathbf{P}^-(k) - \mathbf{W}(k)\mathbf{S}(k)\mathbf{W}^T(k), \quad (26)$$

where the gain matrix is given by

$$\mathbf{W}(k) = \mathbf{P}^-(k)\mathbf{H}^T(k)\mathbf{S}^{-1}(k), \quad (27)$$

the innovation covariance by

$$\mathbf{S}(k) = \mathbf{H}(k)\mathbf{P}^-(k)\mathbf{H}^T(k) + \Sigma_z(k), \quad (28)$$

and where

$$\mathbf{H}(k) = \begin{bmatrix} -\cos \hat{\phi}^-(k) & -\sin \hat{\phi}^-(k) & -(x_b - \hat{x}^-(k)) \sin \hat{\phi}^-(k) \\ \sin \hat{\phi}^-(k) & -\cos \hat{\phi}^-(k) & +(y_b - \hat{y}^-(k)) \cos \hat{\phi}^-(k) \end{bmatrix} \quad (29)$$

is the Jacobian of the transform matrix given in equation (24), taken with respect to the state and evaluated at the prediction as required in equation (16).

3.3. The Navigation Cycle

The navigation system runs on a system comprising two DSPs and four Transputers. The DSPs and two Transputers are dedicated to preprocessing of radar signals and indeed are physically located in the radar housings. The other two Transputers run the main navigation code. The rate at which data arrive from the radars and the update rate required by the navigation algorithm make the timing and communication between the different elements of the system crucial.

The timing of the navigation cycle is described in Figure 8. The cycle rate for the filter is 20 Hz. Broadly, the state prediction component of the navigation filter, running on the first Transputer, generates synchronous predictions of vehicle location and its associated covariance every 50 ms. The prediction is based on synchronous measurements of drive and steer signals together with the previous state prediction. If a matched observation is made during the prediction period, the prediction is back-propagated to the observation time-stamp, and a state update is performed. The updated estimate is then resynchronized with the filter cycle rate by predicting forward the remaining time since the observation. In this way, the observation information is correctly integrated with the estimate at the correct time in the cycle. The necessity of this is clear when one realizes that the vehicle can travel as much as 30 cm in a sample period.

Observation information is processed on the second Transputer, which reads in data from the radars asynchronously. All observations are accurately time-stamped on reception. Prediction information from the first Transputer is used to generate an estimated vehicle location at the observation time. This is used to transform the observation to base coordinates, generate a match with the beacon map, and translate this back into vehicle-centered coordinates. Successful matches, together with the observation time-stamp, are passed back to the first Transputer to be used for updating. Communication between the two navigation Transputers is clearly asynchronous; it is not known in advance when an observation will be made, or how long it will take to generate an appropriate match. The essential design principle is that synchronous operations occur on the first Transputer (which may or may not receive observation information in any given cycle), and asynchronous operations and all hardware (radar) interface functions are dealt with on the second Transputer.

All communication is handled using the synchronized message-passing facilities of the Transputers (which aids software verification). All control inputs and position outputs must occur synchronously at the navigation cycle frequency. In addition, both navigation Transputers are required to pulse the “event pin” from software at the

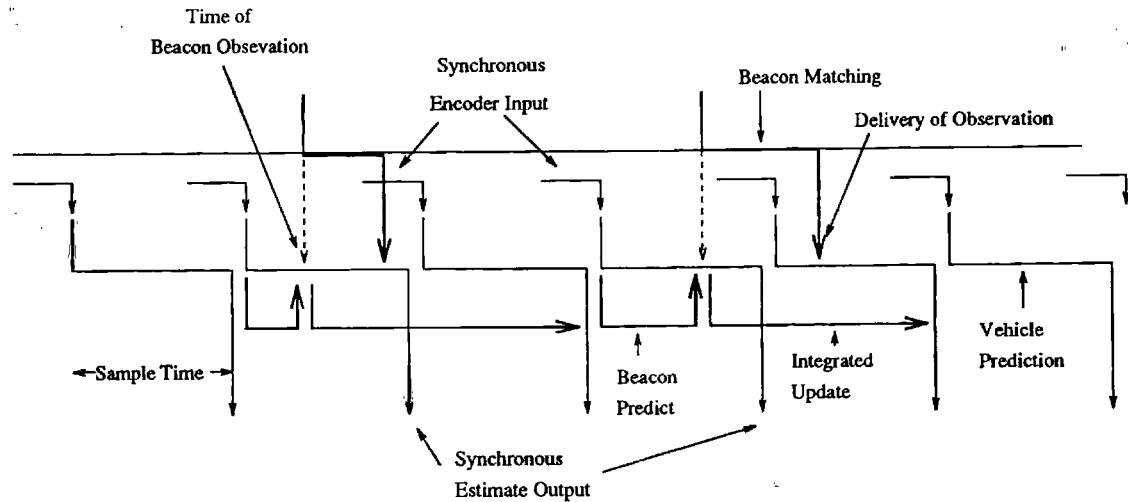


Fig. 8. Overall timing sequence of the navigation filter.

cycle rate. Failure of any of these communications to occur generates a hardware error that, within the next cycle time, releases electromechanical brakes, causing the vehicle to come to an immediate halt.

4. Planning and Control

The essential requirements for the vehicle planning and control system are that it should allow free-ranging specification of paths within the terminal, accurate path tracking, and accurate docking.

The planning system maintains a simple map of the roadways in the terminal. This map incorporates information about junctions, maximum speeds, corner radii, and distances. The planning system also links to the existing terminal software that defines container yard locations and the quay area. Paths are initially specified in terms of straight-line segments together with maximum speeds and required turning radii. This is combined with knowledge of the vehicle geometry and kinematics to produce a set of nominal drive and steer commands that will get the vehicle from a start location to the required end location. These motion commands are selected from only a small repertoire of prototype motions appropriately parameterized for the desired path. This nominal motion command set is checked and validated prior to the execution of the trajectory. During execution, these motion commands are modified through a comparison of the estimated position of the vehicle from the navigation system with the desired position. In motion, the vehicle is velocity controlled to track the desired trajectory. Close to the terminal location, the vehicle reverts to a position control mode to bring it to an accurate stop at the required location. The planning and path specification system has been kept as simple as

possible to ensure that safe operation can be assured at all times, despite the fact that the vehicle kinematics allow much more complex path following.

This section briefly describes the specification, generation, and execution of vehicle trajectories.

4.1. Trajectory Specification

In principle, given a drive function $\omega_f(t)$ and steer functions $\gamma_f(t)$ and $\gamma_r(t)$, the location of the vehicle may be computed by simply integrating equation (1) as

$$\begin{aligned} x(t_2) - x(t_1) &= R \int_{t_1}^{t_2} \omega_f(t) \cos(\phi(t) + \gamma_f(t)) dt \\ y(t_2) - y(t_1) &= R \int_{t_1}^{t_2} \omega_f(t) \sin(\phi(t) + \gamma_f(t)) dt \\ \phi(t_2) - \phi(t_1) &= \frac{R}{B} \int_{t_1}^{t_2} \omega_f(t) \sin \gamma_f(t) - \omega_r(t) \sin \gamma_r(t) dt. \end{aligned} \quad (30)$$

In practice, analytic solutions for these equations can only be obtained for very simple steer functions. Further, given a desired change in location and orientation (a path specification), it is almost impossible to compute a demanded set of drive and steer functions that will achieve this change. This is a well-known problem with this type of equation (Kane and Levinson 1990).

To overcome this problem, the trajectory specification problem is simplified in the obvious way. Trajectories are specified only in terms of a series of straight-line motions. Motions between one line segment and another are implemented at constant speed using only simple trapezoidal steer functions. This has the effect of decoupling the specification of vehicle speed from vehicle steer and

further limiting consideration only to those steer functions for which equation (30) can be solved.

Figure 9 shows a map of a container terminal on the Medway estuary in southern England. Shown on this map are the quay area and the stacking yards. A set of paths is also shown. These paths are simple straight-line specifications joined by turns of known radius. The map is maintained as both a graph of nodes corresponding to each segment and curve and as a geometric description of length, curvature start, and end location. Path specifications also include a description of maximum allowable speeds. Paths may be added, deleted, or changed from the base station.

A task is simply specified as a start and end location on one of the paths. Start and end conditions are also usually constrained by a docking signature of a specific crane (described below). This allows for a degree of along-path motion of the crane. The cross-path crane motion is accommodated by the crane itself, which has a trolley mechanism allowing lift and motion of containers in this direction. A simple graph search is first performed to find the shortest set of straight-line paths that will get the vehicle from the start to the end location. The resulting set of straight-line paths and start and end locations, together with turning and speed constraints, form the specification of the vehicle trajectory.

4.2. Trajectory Generation

From the path specification, a nominal set of drive and steer functions must be generated that will implement the desired trajectory. The drive and steer functions chosen are constrained in three ways:

1. Only antisymmetric steer functions with $\gamma(t) = \gamma_f(t) = -\gamma_r(t)$ are used in trajectory generation. This means that crabbing motions are not used in planning. In this case changes in orientation are given by

$$\phi(t_2) - \phi(t_1) = \frac{2R}{B} \int_{t_1}^{t_2} \omega(t) \sin \gamma(t) dt. \quad (31)$$

2. The vehicle may only change velocity when not engaged in a turning motion. This means that $\omega(t)$ can change only when $\gamma(t)$ is zero, and that $\omega(t)$ must be a constant when $\gamma(t)$ is nonzero. This decouples the effect of $\omega(t)$ and $\gamma(t)$ on the solution to the inverse kinematics equations.
3. The only steer functions allowed are $\gamma = \text{constant}$ or $\dot{\gamma}(t) = d\gamma/dt = \text{constant}$. That is, the steer angle is either constant or changes at a constant rate. Simple analytic solutions to equation (31) exist for these two cases. These two solutions can

be combined to generate general trapezoidal steer functions.

Given the kinematics of the vehicle and these restrictions, it is well known that it is not possible to independently specify a required orientation and location change. This can be appreciated by noting that for a given change in orientation, $\phi(t_2) - \phi(t_1)$, the steer function trapezoid $\gamma(t)$ will be uniquely determined. Because the change in location, for a constant speed, is determined only by this steer function and the vehicle orientation, the location and orientation are not independent and indeed are both uniquely determined by $\gamma(t)$.

The problem of trajectory generation can now be broken down into four simple steps. For each pair of line segment paths:

1. Compute the required orientation change between the line segment pair.
2. Compute a steer trapezoid that will achieve this orientation change.
3. Find the locations at which the resulting turning motion begins and ends.
4. Fill in the remaining parts of the line segment with straight-line motion.

A number of further constraints must be accounted for during trajectory generation. In particular, the maximum steer γ_m angle is restricted by the mechanics of the vehicle. γ_m is taken to be slightly smaller than is physically possible to allow for a certain amount of oversteer correction during trajectory execution. A maximum steer rate $\dot{\gamma}_m$ in ramping and a maximum acceleration $\ddot{\omega}_m(t)$ must be observed. Although the maximum steer rate is constant, the maximum acceleration is specified as a function of vehicle velocity by a lookup table. A further constraint occurs because of the intrinsic sloppiness of the hydraulic drive system; when changing between demand signals, a settling time ΔT_s must be allowed for in both drive and steer functions to allow the executed control signal to stabilize to the demanded signal.

Trajectory generation now proceeds as follows. Let \mathbf{p}_i^s , \mathbf{p}_i^e , \mathbf{p}_j^s , and \mathbf{p}_j^e , with $\mathbf{p}_r^q = [x_r^q, y_r^q]$, be the start and end location vectors of two adjacent plan line segments. The orientation of the two segments, and thus of the vehicle, before and after the turning motion are clearly

$$\phi_i = \tan^{-1} \frac{y_i^e - y_i^s}{x_i^e - x_i^s}, \quad \phi_j = \tan^{-1} \frac{y_j^e - y_j^s}{x_j^e - x_j^s},$$

and so the required orientation change is given by $\Delta\phi = \phi_j - \phi_i$. During turning motions, the speed of the vehicle $R\omega$ is assumed constant. The steer trapezoid and resulting position change is now computed from constant steer and constant steer rate turning motions as follows:

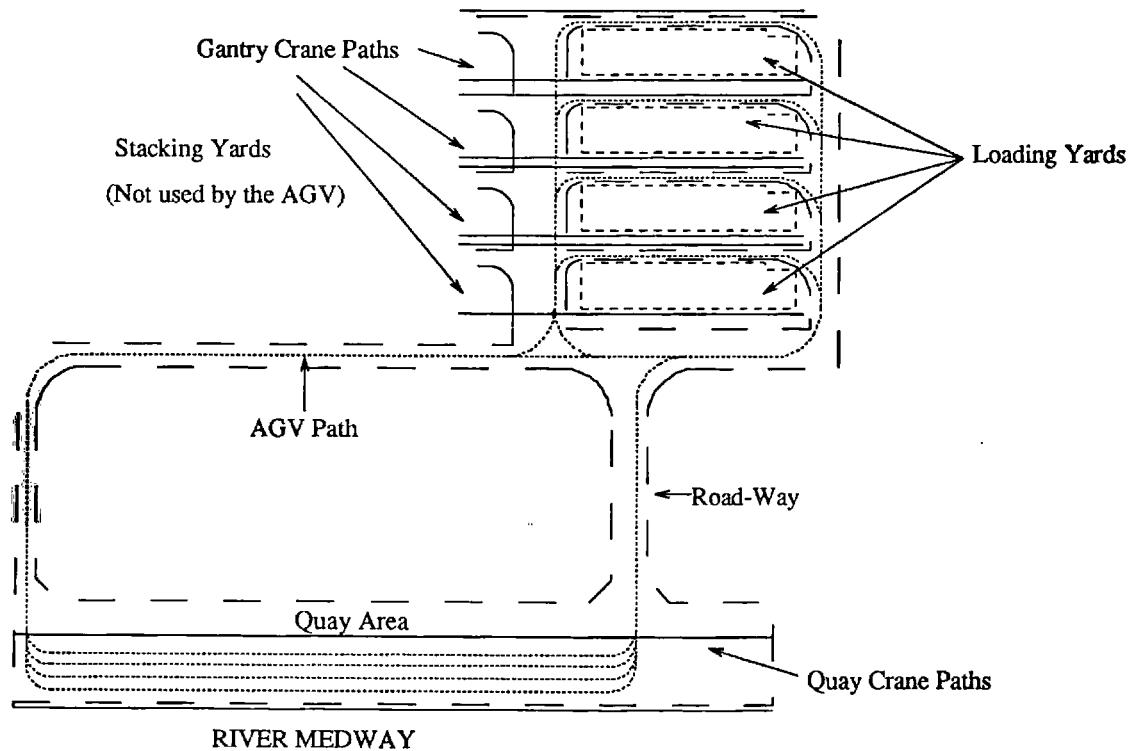


Fig. 9. Map of a container port showing the quay, yard and stacking area, and main AGV paths.

Constant Steer Turns When $\gamma = \text{constant}$ the change in orientation $\Delta\phi$ of the vehicle over the time interval ΔT is given by

$$\Delta\phi = \Delta T \frac{2R\omega}{B} \sin \gamma. \quad (32)$$

When the starting orientation is taken as zero, the corresponding changes in location can be found from equation (30) as

$$\Delta x = \frac{B}{2 \sin \gamma} [\sin(\Delta\phi + \gamma) - \sin \gamma] \quad (33)$$

and

$$\Delta y = \frac{B}{2 \sin \gamma} [\cos \gamma - \cos(\Delta\phi + \gamma)], \quad (34)$$

except when $\gamma = 0.0$, in which case

$$\Delta\phi = 0, \quad \Delta x = R\omega\Delta T, \quad \Delta y = 0.$$

Constant Steer Rate Turns When $\dot{\gamma} = \text{constant}$, the change in orientation is given by

$$\Delta\phi = \frac{2R\omega}{B\dot{\gamma}} [\cos \gamma_s - \cos \gamma_e], \quad (35)$$

where γ_s is the starting steer angle, and $\gamma_e = \dot{\gamma}\Delta T + \gamma_s$ the terminal steer angle. Equation (30) cannot be solved analytically for location change (except in terms of a J_1 Bessel function), and so the integrals must be evaluated numerically.

Combined Steer Turns Each steer operation is trapezoidal, comprising a ramp up to a particular steer angle, a time at this steer angle, and a ramp down to a final steer angle. The time spent at the constant steer must be at least equal to the settling time of the drive. From equations (32) and (35), the total orientation change $\Delta\phi$ during a steer operation is given by

$$\Delta\phi = \frac{2R\omega}{B} \left[\frac{\cos \gamma_1 + \cos \gamma_2}{\dot{\gamma}} - \frac{2 \cos \gamma_3}{\dot{\gamma}} + \Delta T_3 \sin \gamma_3 \right] \quad (36)$$

where $\gamma_1, \gamma_2, \gamma_3$, are the start, end, and peak steer angles, respectively, and ΔT_3 is the time spent at peak steer angle. The maximum steer angle $\gamma_m = \gamma_3$ will be attained if

$$\left\| \frac{\Delta\phi B \dot{\gamma}}{4R\omega} \right\| \geq \frac{\Delta T_s \dot{\gamma}}{2} \sin \gamma_m + \frac{\cos \gamma_1 + \cos \gamma_2}{2} - \cos \gamma_m, \quad (37)$$

where ΔT_s is the drive settling time (and where γ_m is always taken as positive). The time at the maximum steer angle is found from

$$\Delta T = \frac{\left[\left\| \frac{\Delta \phi B \dot{\gamma}}{4R\omega} \right\| + \cos \gamma_m - \frac{\cos \gamma_1 + \cos \gamma_2}{2} \right]}{\frac{\dot{\gamma} \sin \gamma_m}{2}}. \quad (38)$$

If equation (37) is not satisfied, then the peak steer angle attained is given by

$$\pm \gamma_3 = \cos^{-1} \left[\frac{\frac{\cos \gamma_1 + \cos \gamma_2}{2} - \left\| \frac{\Delta \phi B \dot{\gamma}}{4R\omega} \right\|}{\sqrt{\left(\frac{\Delta T_s \dot{\gamma}}{2} \right)^2 + 1}} \right] - \tan^{-1} \left[\frac{\Delta T_s \dot{\gamma}}{2} \right], \quad (39)$$

where γ_3 is taken to be the same sign as $\Delta \phi$. The steer angle remains at γ_3 for the settling time ΔT_s .

Start and End Location of Turning Motion Once the orientation change has been specified and the combined steer function computed, the position changes may be found from equations (33) and (34) and numerical integration of equation (30). The changes in location and orientation computed are independent of start location and orientation of the vehicle. These changes are referenced to an absolute location in the vehicle plan through the specification of a "hit-box" or location stamp. A change in orientation is always location independent. The changes in position in the global coordinate frame must be referenced to a starting orientation ϕ_s , as follows:

$$\begin{bmatrix} \Delta x' \\ \Delta y' \end{bmatrix} = \begin{bmatrix} \cos \phi_s & -\sin \phi_s \\ \sin \phi_s & \cos \phi_s \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}.$$

This must be computed for each individual steer command and the total change for each steer operation computed. This is done as follows: Given location changes Δx and Δy , the start hit-box \mathbf{x}_i must lie on the plan segment i , and the end hit-box on the plan segment j :

$$\mathbf{x}_i = \mathbf{p}_i^e + \lambda_i (\mathbf{p}_i^s - \mathbf{p}_i^e), \quad \mathbf{x}_j = \mathbf{p}_j^s + \lambda_j (\mathbf{p}_j^e - \mathbf{p}_j^s). \quad (40)$$

The location change must be equal to the vector difference between these hit-boxes:

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = [\mathbf{p}_1^e - \mathbf{p}_1^s \quad \mathbf{p}_1^e - \mathbf{p}_1^s] \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}.$$

Solving for the variables λ_1 and λ_2

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = [\mathbf{p}_1^e - \mathbf{p}_1^s \quad \mathbf{p}_1^e - \mathbf{p}_1^s]^{-1} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

and substituting back into equation (40) gives the location of the hit boxes.

Straight-Line Drive Functions When the steer functions have been computed, the start and end hit-boxes in a segment define the start and end of the drive function that connects them. The drive function is simply computed from prespecified acceleration tables. The vehicle is allowed to accelerate up to a maximum velocity specified by the path. It typically must also decelerate into a corner. The terminal motion when the vehicle must come to a stop is treated separately. The vehicle first decelerates to 0.5 m/s at a distance of 1.5 m from the target destination, and then ramps down speed linearly to a stop at the desired location.

4.3. Trajectory Execution

The trajectory specified by the computed drive and steer functions is computed prior to execution. It is independently checked by the pilot through a simulation of the motion to verify that the vehicle will indeed end up at the desired location. Once the trajectory has been verified, it is executed.

The control sequence of drive and steer signals is specified as a "ladder diagram" that is passed to the control system for execution. Each rung in the ladder contains steer and drive information, as well as duration, location, and orientation change data. This permits a variety of on-line checks to take place. Each rung in the ladder is stamped by the location at which the drive must be executed. The control system executes each rung in the ladder in sequence. Each rung is contained in a "control-box," a physical area in which the vehicle must be for the rung to be executed. Adjacent rungs are linked by a hit-box, a physical location in which the vehicle must be for control to pass from one rung to the next. This position stamping ensures that any preconditions of the control action are met prior to execution and allows for unforeseen time delays in execution or monitoring. The location information is obtained independently from the navigation subsystem. Each rung in the ladder is specified in terms of a location-independent change in location and orientation. This means that dynamic changes in start conditions (on entering a hit-box) do not require that the control sequence be recomputed.

During execution, path tracking is accomplished by comparing the vehicle location as estimated by the navigation system with the desired vehicle location. This comparison is done in a coordinate system centered on the center of the vehicle and not on the usual center front-axle coordinate system. This is because when the vehicle is traveling in reverse, a correction in location of the front axle leads to an initial opposite correction in the forward traveling rear axle. Two errors are computed for path tracking: the perpendicular distance of the estimated vehicle center from the desired path δd , and the estimated

vehicle orientation error $\delta\phi$. No account is taken of how far along the path the vehicle is. The vehicle is made to follow the path by applying corrective steer signals $\delta\gamma_f$ and $\delta\gamma_r$ for both front and rear axles in the form

$$\delta\gamma_f = K_p\delta d + K_o\delta\phi, \quad \delta\gamma_r = K_p\delta d - K_o\delta\phi,$$

where K_p and K_o are experimentally determined constant gains. The effect of this control is clear: with a pure position error ($\delta\phi = 0$), the corrective steer is the same for both front and rear axles, and so the vehicle crabs sideways with no change in orientation. Conversely, with a pure orientation error, the two axles turn in opposite directions providing a change in orientation (and also position). This simple proportional position control law was found to provide more than adequate tracking performance. The reason for this is that the kinematic properties of the vehicle itself provide a degree of both velocity and integral control through the orientation constraint.

Terminal position control is effected only when the vehicle has decelerated to 0.5 m/s and is approximately 1.5 m from the desired end position. The exact end position is usually modified slightly from the original plan, as the vehicle must dock with a crane that is also mobile. The crane will move slightly from its normal station to accommodate motion of the ship being loaded or unloaded. The exact crane location is determined by fixing a set of beacons to the crane structure that are detected by the navigation algorithm. A simple PID control is used to bring the vehicle to a stop at the correct location.

5. Implementation and Validation

The system design allowed the four main vehicle components to be initially developed and tested independently. The pilot process, including trajectory specification and generation, can reasonably be tested in simulation. The navigation system was first tested simply as a position-reporting device with no feedback into the control system. In this case the vehicle was simply moved using a joystick. The control system was also initially tested with no position feedback from the navigation system. This was done using a dummy navigation process based only on encoder measurements and a simulation of the vehicle. These initial tests allowed determination of parameters for PID loops and minor corrections to the hydraulic drive train. Following these tests, the navigation and control systems were linked and an integrated trials program undertaken. The condition monitoring and safety system was first tested in tandem with the navigation and control systems before being fully integrated. The reliability of the condition monitoring sensor system was found to be quite poor in the first instance, making it difficult to operate extended trials. Most initial faults in the vehicle

were caused by the failure of sensor connections. Finally, the pilot and base station were linked in to the control system, and fully integrated trials were conducted.

In this section we concentrate on describing the validation process and the final performance of the navigation and control systems. These are considered to be the central new elements of the vehicle system. The validation process is first described. This involved a number of stages for testing the radar and the operation of the navigation system on the vehicle. One of the main difficulties to be overcome in validation is that the true vehicle motion is difficult to determine, and thus, absolute measures of performance for both the navigation and control system are difficult to obtain. The operation of the navigation and control system for a specific vehicle trajectory is then described in detail. This allows a detailed analysis of the main features and performance of the vehicle system and highlights some of the principal benefits of the process model and control algorithms presented in Sections 3 and 4. Finally, the implementation of the vehicle system in a container terminal is described. The principal elements of this implementation, the problems encountered, and the experience gained in operation are described.

5.1. Validation Process

The overall complexity of the system and the overriding problem of safety with a vehicle of this size and speed made it essential to validate key elements of the design prior to full implementation. In particular, the simple practical problems of data logging and code debugging in a system that is moving with substantial kinetic energy proved to be quite complex. The on-vehicle validation of navigation and control systems was helped by being able to initially separate these two systems and to test each independently.

The control system was first validated using a simple hardware simulator. The on-vehicle tests were first conducted with the vehicle on jacks and subsequently in a large empty area (a former oil refinery). The validation of the navigation system, although theoretically straightforward, turned out to be quite difficult in practice. This was for two main reasons:

1. Except when the vehicle is stationary, it is impossible to know what the "true" location of the AGV is at any given time. Consequently, it is difficult to find any absolute measure of how well the navigation system is performing.
2. Standard innovation type measures (the sequence of errors between predicted observations and true measurements) of performance are difficult to use, as the observation prediction process depends heavily on

which beacon is being observed. Further, an innovations analysis does not readily allow the allocation of errors between vehicle and sensor.

To overcome these problems, the validation of the navigation systems and software was accomplished in three phases.

1. Independently calibrated trials of the radar units were performed. The radars were placed on a buggy on a 200-m-long railway line, calibrated for absolute distance to 5 mm. A number of different beacon layouts were tested at all ranges. Calibration curves were produced together with detection and gating probabilities.
2. The radars were placed on the vehicle. The vehicle was equipped with a paint drip and required to move up and down in a straight line of about 200 m. The paint tracks were then surveyed into the beacon map. This provided a coarse absolute measure of straight-line absolute accuracy. Repeat tests with container clutter were performed. Tests on curved tracks proved to be impossible to check absolutely.
3. Repeatability tests were performed in which the vehicle was required to execute a path, coming to a halt at specific places where location and orientation were measured. A typical trial involved the vehicle executing the same trajectory continuously for periods of about 12 hours.

The validation process was undertaken over a 6-month period. It highlighted some major problems in the generation of absolute performance measures and validation procedures for this type of AGV system. In future work, it is considered essential to develop a systematic and comprehensive validation suite for this type of problem. This would ideally involve the use of a second absolute position sensor to obtain independent measurements of the AGV location,³ together with the determination of an appropriate error budget for the different sensors and vehicle states (as described in Maybeck [1979]).

5.2. System Analysis

To explain the operation and performance of the vehicle system, we concentrate on describing a specific example trajectory in detail. This example trajectory was executed on a test site adjacent to the container terminal and

consisting of a terrain of rough tarmac roads and an environment with only a few buildings and other obstacles. The test site provides a generally "cleaner" environment for the radar, with lower false alarm rates and better beacon view characteristics than was found in the container terminal.

The trajectory considered and the layout of beacons in this instance is shown in Figure 10. The path is initially specified by the three straight-line segments $[x, y] = [0, 0]$ to $[40, 0]$, $[40, 0]$ to $[40, 30]$, and $[40, 30]$ to $[0, 30]$. A trajectory is generated that links these line segments by vehicle steer functions. Figure 10 shows the resulting planned path to be taken by the center of the vehicle and by the coordinate system centered on the front axle (the vehicle reference coordinate system). An outline of the vehicle at each hit-box along the path is also shown. The overshoot of the front axle from the plan specification after the two turns is a well-known characteristic of this type of car-like vehicle. The nominal demanded wheel velocity and mean front steer angle are shown in Figure 11. The demanded mean wheel velocity and demanded front steer angle during trajectory execution are shown in Figure 12. The vehicle is controlled to remain on the trajectory by comparing estimated positions with the specified trajectory and using both front and rear axles to steer the vehicle onto track. This gives rise to both the differences in value and timing between nominal and demanded control action. The mean measured angular wheel velocity $\bar{\omega}(k)$ for the trajectory is shown in Figure 13A. The vehicle starts stationary at the origin of the base coordinate system. The navigation system is initialized, and the vehicle first remains stationary for approximately 30 seconds. This allows the navigation filter to approach a steady state. At 32 seconds the vehicle accelerates along the x -axis to 2 m/s. At 47 seconds, the vehicle executes a 90° turn onto the trajectory defined by $x = 40$ m. The mean measured front steer angle $\bar{\gamma}(k)$ is shown in Figure 13B. The vehicle continues at 2 m/s until, at 62 seconds, it executes a second 90° turn onto the new trajectory defined by $y = 30$ m. At 77 seconds the vehicle decelerates to approximately 0.5 m/s, when the position control mode comes into effect. The vehicle then comes to a halt at 86 seconds at $x = 0, y = 30$. The vehicle then remains stationary. The vehicle is velocity controlled along this trajectory except when approaching the terminal position, when the vehicle reverts to a position control mode. The changeover from velocity control to position control can be clearly seen at approximately 82 seconds. There are minor tracking steer corrections while the vehicle is moving in a straight line from 32 to 47 and from 75 to 87 seconds. The steer moves to maximum lock during the two 90° turns, and there is quite a large amount of corrective

3. The essential problem here is that if such a sensor were available, then it could, in principle, be used as the navigation sensor itself and so the original problem would remain. In practice, though, it should be possible to use something like a laser theodolite to provide such information, at least over short distances.

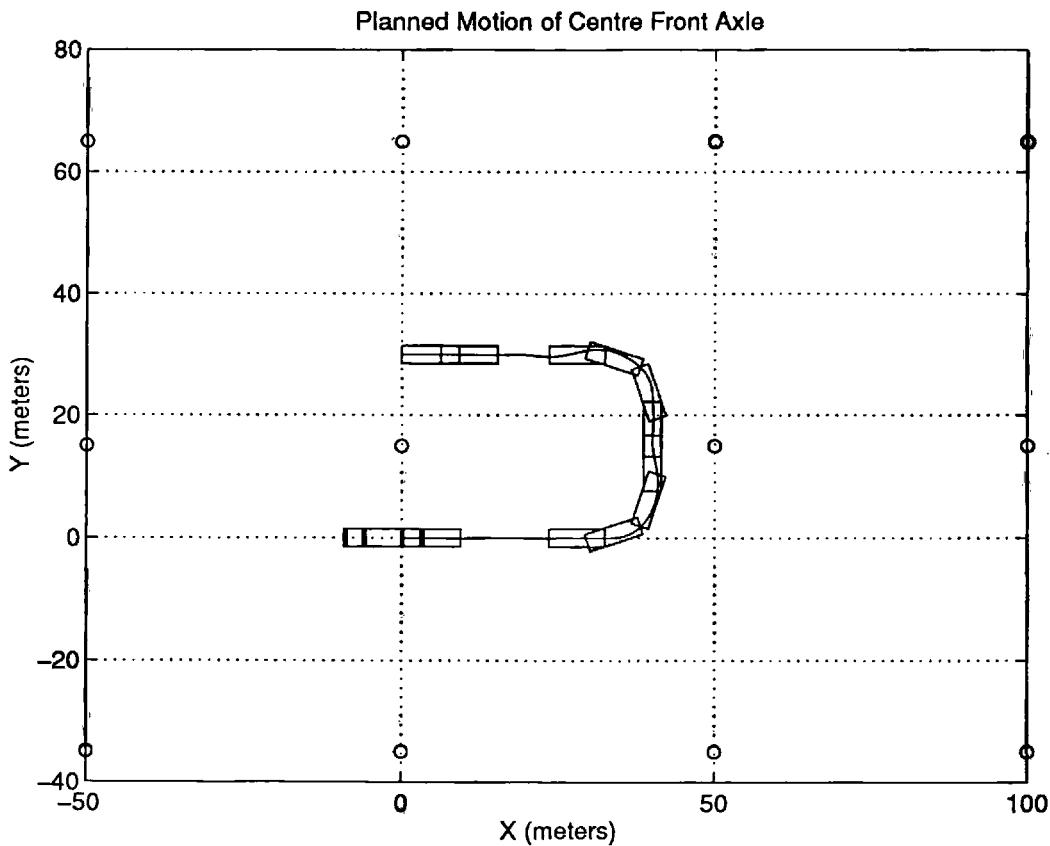


Fig. 10. The example trajectory and beacon layout. The figure shows the path of the vehicle front axle and an outline of the vehicle at each of the hit-boxes generated along the trajectory. The beacons are shown by the symbol “o”.

steer as the vehicle resumes straight-line motion at 58 and 72 seconds.⁴ Some steer correction can also be seen in the second corner at 66 seconds. The rear steer angle is very nearly a mirror image of the front steer angle, as the rear steers negative when the front steers positive to produce a smaller vehicle turn radius. The exception to this is when the vehicle corrects a pure position track error by steering both axles in the same direction.

The estimated values for the error source terms were found from an evaluation of the initial validation test statistics together with a certain amount of on-line testing. The performance of the vehicle navigation system was found to be relatively insensitive to specific values of process model errors up to a factor of 2 to 4, while being quite sensitive to estimated observation errors. The reason for this becomes clear when the variance history of the filter is analyzed. The values used in the example described are as follows:

Multiplicative slip error σ_q (%/100):	0.02
Additive slip error σ_ω (rads/s):	0.1
Multiplicative skid error σ_s (%/100):	0.02
Additive skid error σ_γ (rads):	0.035
Wheel radius error rate σ_R (m/s):	0.001
Radar range error σ_r (m):	0.3
Radar bearing error σ_θ (rads):	0.035

The navigation system must be accurately initialized and allowed to reach a steady operation at start-up. Initialization is performed once, and then the vehicle is left in continual operation. The vehicle must be initialized at a known location, as there are no unique identities associated with the beacons. To do this, the vehicle is manually driven over one of a set of starting markers placed on the ground at exactly known positions. In this example trajectory, the initial state is given as $\hat{x}(0 | 0) = [0.0, 0.0, 0.0, 0.6]^T$. The initial position errors are taken as

$$\sqrt{P(0 | 0)} = \begin{bmatrix} 0.3 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.3 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.05 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.01 \end{bmatrix}$$

4. It is typically more space efficient to oversteer out of a corner and then to steer back onto track, as otherwise the front of the vehicle can substantially overshoot the initial straight-line plan.

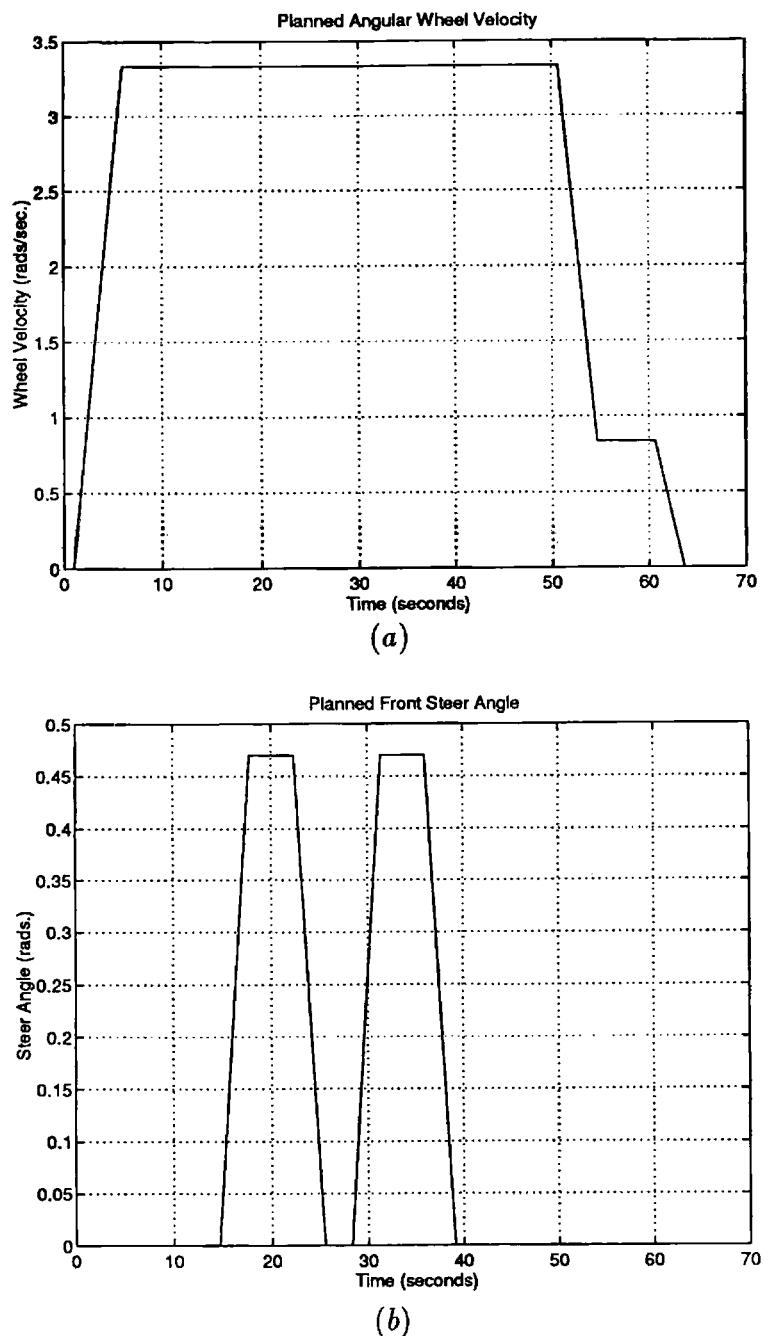


Fig. 11. The nominal control signals generated for the example trajectory. (A), Mean angular velocity of front wheel. (B) Mean front steer angle.

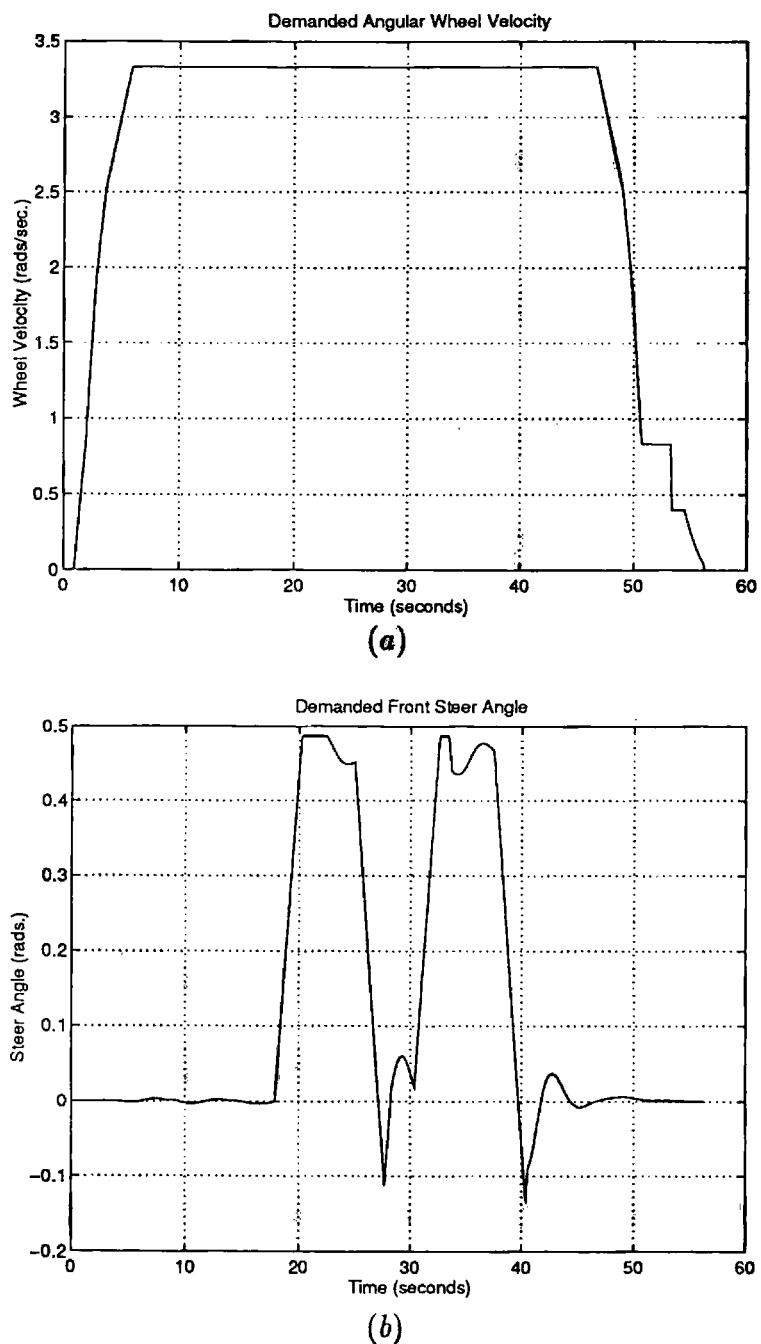


Fig. 12. The demanded control signals generated during execution of the example trajectory. (A) Demanded angular wheel velocity. (B) Demanded front steer angle.

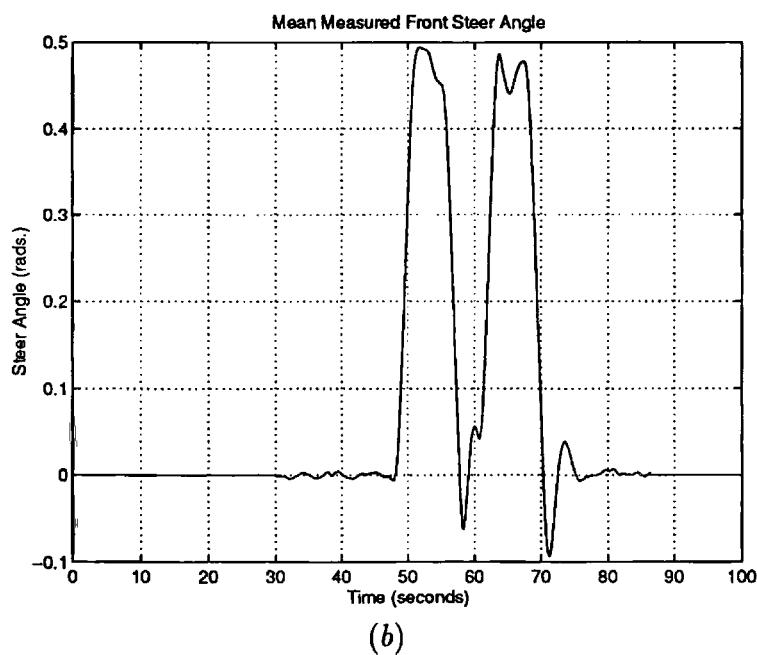
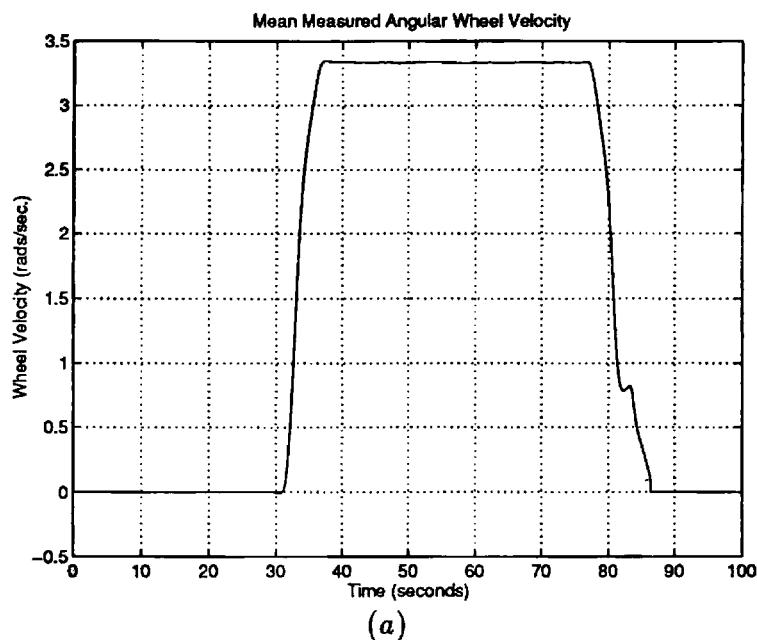


Fig. 13. The measured control signals generated during execution of the example trajectory. (A) Mean measured angular wheel velocity. (B) Measured front steer angle.

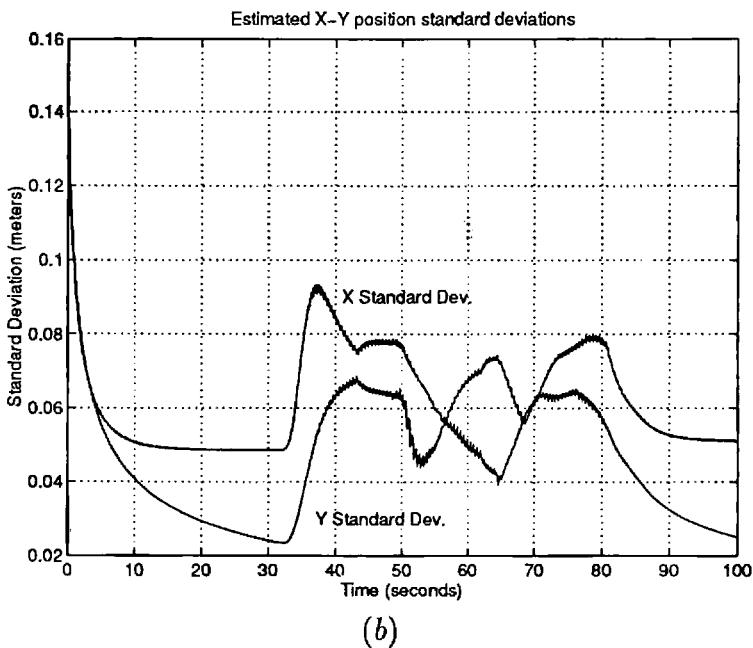
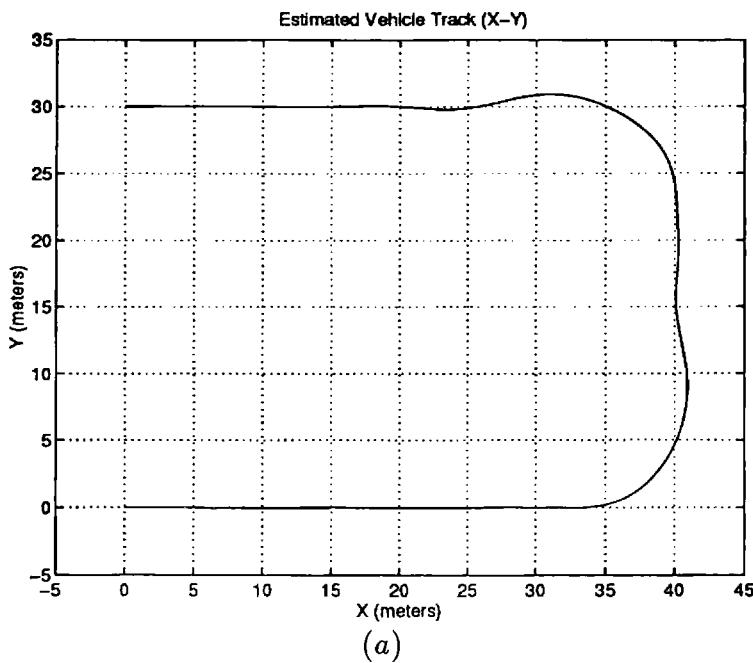


Fig. 14. The estimated trajectory of the center front axle (A), and the example standard deviations in x and y coordinates of the estimated front axle location (B) as computed by the navigation system.

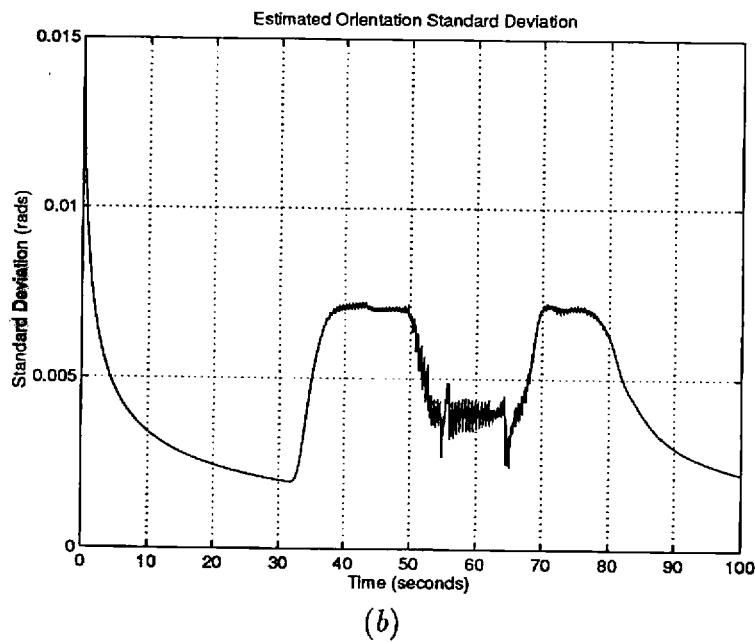
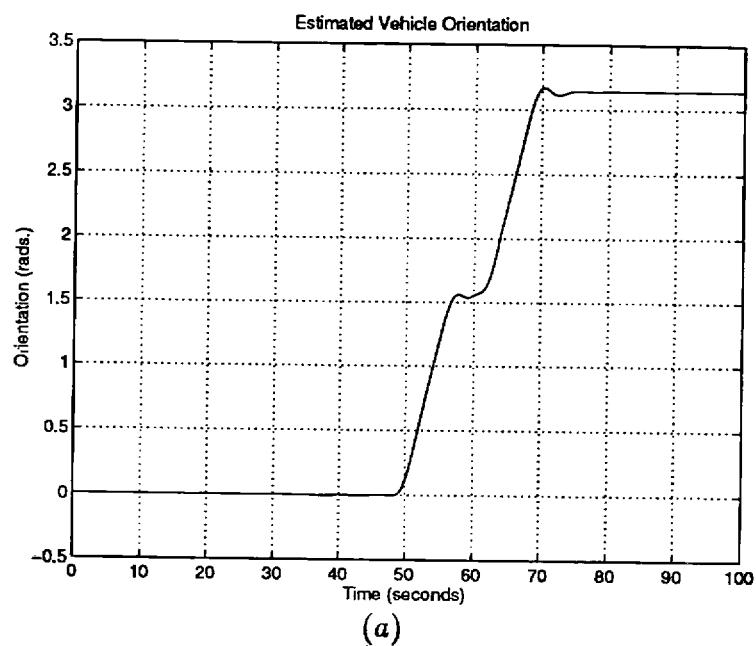


Fig. 15. The estimated orientation of the vehicle (A) and the estimated standard deviation in estimated vehicle orientation (B) as computed by the navigation system.

Figure 14A shows the trajectory estimated by the navigation filter. It should be emphasized again that only the start and end locations of the trajectory can be independently validated (by theodolite), and that the trajectory shown is an estimated path and not an actual path. The estimated trajectory shows the path taken by the center front axle and may be compared to that computed from the nominal control signals (Fig. 10) as a comparison of control system performance. The trajectory was verified (very approximately) by visually aligning ground markers with maximum outswings of the vehicle front and rear. The terminal position was estimated by the navigation system at (0.012, 30.005) m and independently measured by theodolite at (0.028, 29.993) m. The standard deviations in x and y location estimates are shown in Figure 14B. From 0 to 32 seconds the standard deviation estimates converge from initial to steady-state values. The steady-state values depend on the layout of the observed beacons in the neighborhood of the vehicle. The y standard deviation is lower than the x value at this start location, because the vehicle is oriented along the x axis: The along-axis process model noise source $\delta\Omega(k)$ (in the direction of vehicle travel) is larger than the cross-axis process noise source $\delta\Gamma(k)$, as might be expected, given that the vehicle moves primarily by driving in the heading direction rather than by slipping sideways. At 32 seconds, when the vehicle starts moving, both x and y standard deviations rise rapidly. This is for two reasons: first, the additional injection of multiplicative process noise into (primarily) along-axis motion estimates; and second, due to the effect of large wheel radius uncertainty (described below), affecting motion estimates. After an initial transient, the in-motion standard deviation estimates reach a steady state at approximately 45 seconds. The increase in standard deviation at this time as compared to stationary values is almost all due to the injection of multiplicative noise. Note that this increase in standard deviation increases linearly with vehicle speed. As the vehicle enters the first curve, both x and y standard deviations *reduce*: Initially, because of the multiplicative skid error, one might expect vehicle position estimate uncertainty to increase in the corner; however, this is offset in both x and y directions for two independent reasons. First, the x standard deviation reduces because as the vehicle steers into the corner, the vehicle reorients so that the (relatively large) along-axis errors are now generated in the y direction. Second, as the steer angle $\bar{\gamma}_f(k)$ increases, the rate of change of orientation also increases, making the orientation "more observable" (described below). This in turn reduces orientation uncertainty, with a consequent increase in the observability and reduction in standard deviation of estimate of y . Once the steer signal reduces close to zero (at approximately 65 seconds), the x and y standard deviations return to a straight-line motion value

in which x and y values are reversed from their values prior to the turn. This reversal is again due to the change in vehicle heading, and along-axis process noise injection, from x to y directions. After the second turn, the x and y standard deviations again reverse values and gradually reduce as the vehicle comes to a halt. The slight high-frequency variations observed in the standard deviation estimates are due to updates occurring from different beacons at different ranges with consequently different Cartesian observation errors. This plot of x and y standard deviation demonstrates many of the most useful and important features of the vehicle model proposed.

Figure 15A shows the estimated vehicle orientation produced by the navigation system during this example trajectory. The orientation estimate would appear reasonable and consistent, although there is essentially no practical way of validating the estimated values while the vehicle is in motion. The terminal orientation as estimated by the navigation system was 3.140 rad, and the independent theodolite measurement (from observation of two points on the vehicle) gave actual orientation at 3.142 rad. Figure 15B shows the estimated standard deviation in orientation estimates. The orientation standard deviation tends to a steady-state value while the vehicle is stationary. At 32 seconds, when the vehicle begins moving, estimated orientation uncertainty rises to an in-motion steady level. As the vehicle enters a turning motion, the value reduces. This is because as the steer signal $\gamma(k)$ increases, the terms $\sin \gamma(k)$ governing the orientation rate also increase. This in turn more closely correlates the orientation error with the vehicle position error. The vehicle position error is the state most closely correlated with the error in beacon observation and so benefits most when an observation is made. As the orientation itself becomes more closely correlated with the vehicle location, it consequently benefits from also being more correlated with observation error. This effect dominates any increase in uncertainty introduced by multiplicative skid error sources. While turning, the orientation error remains low, then increases again when the vehicle returns to straight-line motion, finally reducing again as the vehicle slows to a halt.

The estimate of mean wheel radius is shown in Figure 16A. The estimate remains at its initial value until the vehicle starts to move. This is because the wheel radius is unobservable when the vehicle is stationary. The estimated mean radius rises quite quickly once the vehicle is in motion to stabilize at a value close to 0.66 m. This value does not necessarily correspond to any physical wheel radius, but rather constitutes all effects that relate the angular velocity of the axle to the estimated linear vehicle velocity. Figure 16B shows the estimated standard deviation in mean wheel radius estimate. The growth in standard deviation from 0 to 35 and from 83

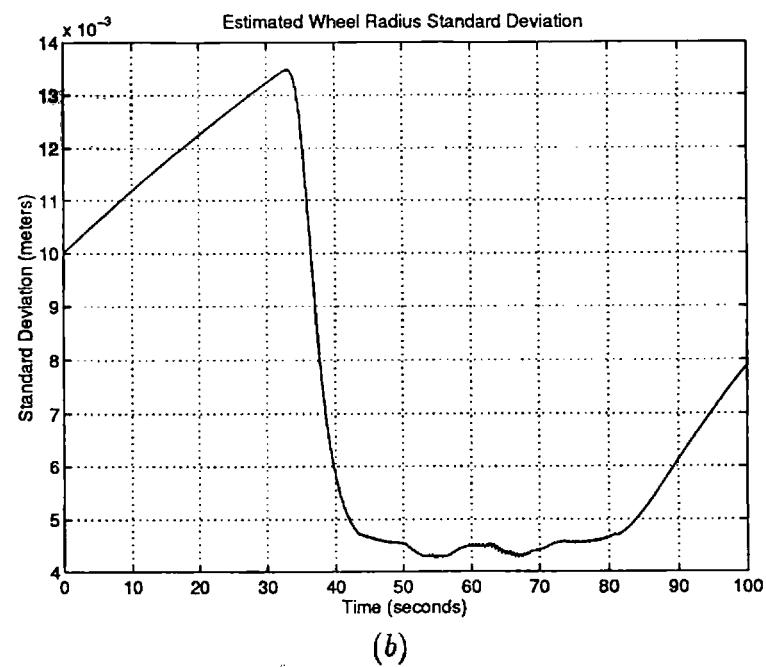
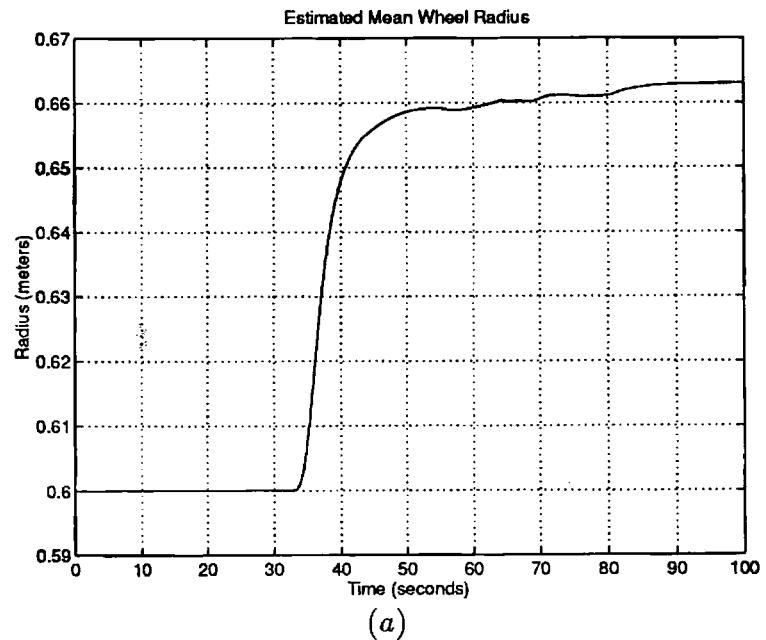


Fig. 16. The estimated average vehicle wheel radius (A) and the estimated standard deviations in estimated average wheel radius (B) as computed by the navigation system.

to 100 seconds reflects the fact that the wheel radius is unobservable when the vehicle is stationary. The standard deviation reduces substantially once the vehicle is in motion (35 to 40 seconds) and as the radius estimate becomes correlate with the position estimate. Once in motion, the standard deviation is approximately constant. Practically, the unobservability in wheel radius, when the vehicle is stationary, is limited by setting $\sigma_R = 0$ once the standard deviation rises above a certain point. This stops an unbounded growth in radius uncertainty. However, a high level of wheel radius uncertainty when the vehicle is stationary is often desirable, as this is when large changes in vehicle load (and thus wheel deflection) occur. This model therefore copes well with axle and wheel deflections caused by change in vehicle load. Slow changes in effective wheel radius due to terrain variations (slipping appears as a wheel radius change) and to tire warmup, are adequately modeled by the near constant standard deviation found during motion.

The value and effectiveness of the process model is qualitatively well demonstrated by its accounting for all major effects of vehicle motion, terrain, and wheel variations. A quantitative measure of the effectiveness of the model can be obtained by considering the rate at which the process model injects errors into the state estimates once the filter is considered to be well tuned. As an example of this, Figure 17 shows the rate of process noise injected into the estimate of x position. This describes how errors in this estimate would grow in the absence of any observation information and so indicates how good the model is at any stage in the trajectory at predicting position. When the vehicle is in stationary position, uncertainty grows at the rate of approximately 2 cm/s. There is a sharp rise in uncertainty growth when the vehicle begins moving due to initial uncertainty in wheel radius, but this then reduces to an in-motion injection rate of approximately 6 cm/s. Generally, the lower the noise injection rate, the better the vehicle model. The better the vehicle model, the better use made of the observation information. Equivalently, a good vehicle model both improves position estimation accuracy and reduces the requirement for accurate external sensing.

As previously stated, it is difficult to validate vehicle state estimates because of the simple lack of knowledge of the true vehicle state. However, it is sometimes possible to get a good indication of how well the filter matches the true state errors through an analysis of the innovation sequence. In principle, if the assumptions necessary in the Kalman filter are satisfied, the innovation, or error between true and predicted observations, will be zero mean and white with known variance. Figure 18 shows a segment of the x -axis Cartesian innovation sequence and associated 1-sigma gate from the trajectory under study. It can be seen that in this section the innovation

levels are generally comparable to the gate width. This implies that the filter is operating correctly. More precise tests for whiteness are difficult to perform on this type of data set because of observations switching between beacons and the generally high false alarm rate. The switching between different beacons can clearly be seen in the variations in gate size. Sharp changes correspond to the alternate viewing of short-range and long-range beacons. The gradual growth of the gate across the segment reflects the change in range of these beacons.

One of the most serious problems in operating the radar sensor in environments such as container terminals is the elimination of false-alarm or clutter signals due to large objects such as containers, cranes, and buildings. The problem is minimized by providing the beacons with a "unique" polarization pattern unlikely to be found in other environment features. However, this does not completely eliminate the problem, as some error must always be tolerated in matching polarization patterns, and this consequently always leads to false targets being matched. As previously described, remaining false targets are eliminated using a conventional normalized innovation validation gate. The size of the gate is a compromise between having a small enough gate to reject false alarms and having a large enough gate to ensure a high matching rate to correct beacon detections. In this application, the rejection of false alarms is all-important, as a beacon mismatch can have potentially catastrophic effects on position estimation accuracy. This requires the gate to be as small as possible. However, the gate cannot be made too small, as it must allow for sudden errors in vehicle location, which may not be tracked if subsequent correct detections are rejected. A gate size of 0.5 was chosen in the test trajectory described here. This is based on measurements of false alarm and detection frequencies observed during validation. Figure 19 shows a 10-second moving average of the rate of beacon detection and the rate of beacon validation with this gate size. The beacon detection rate corresponds to a detection probability and indicates the fraction of correct beacon observations correctly validated. The beacon validation rate corresponds to 1 minus the false alarm probability and shows the overall fraction of observations validated. The beacon detection rate indicates that approximately 80% of correct observations are used by the navigation filter. The beacon validation rate shows that approximately 80% of all measurements are considered to be false alarms.

5.3. Application Trials

The vehicle system described in this article was installed at an operating container port, Thamesport, on the Medway estuary in southern England. The vehicle

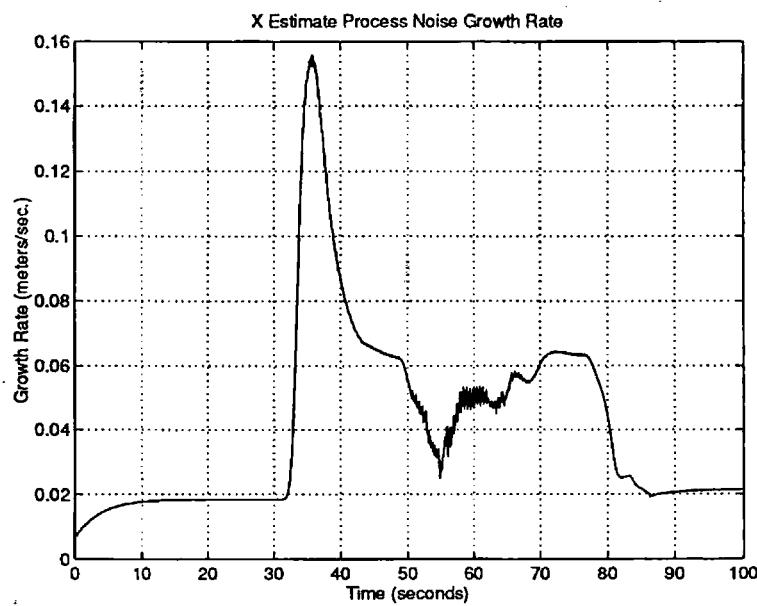


Fig. 17. Rate of process noise injected into the estimated x position coordinate.

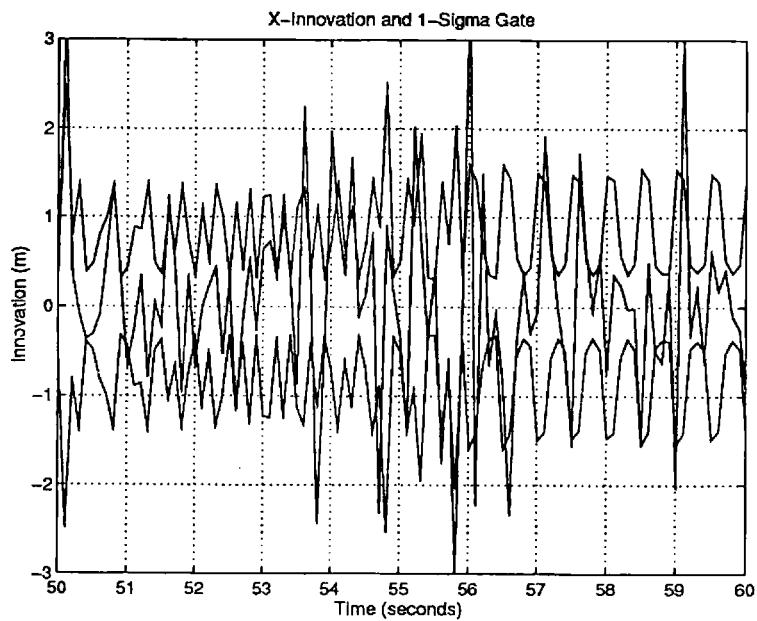


Fig. 18. A portion of the innovation sequence and 1-sigma gate for the x observation coordinate.

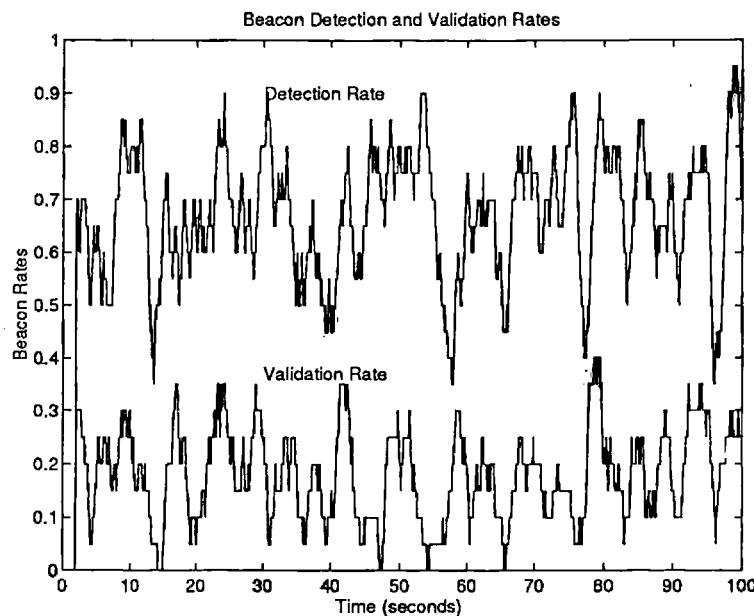


Fig. 19. Ten-second moving average of beacon detection and beacon validation rates.

was logically integrated with the manned vehicle fleet and interfaced with central logistics and with both gantry and quay crane systems. Installation took place over a 6-month period and was completed in October 1993. We briefly describe here some of the main points of the installation and the main problems experienced during operations so far.

The installation area consisted of a main stacking area, with six rail-mounted gantry cranes covering three stacking yards, and a quay area with three post-panamax (90 m) quay cranes. The total installation area is approximately 2 square kilometers, with a quay length of about 600 m (an outline map of the port is shown in Figure 9). The AGV was operated in one allocated stack (with other manned vehicles), over a bridge connecting the yards with the quay, and along the eastern end of the quay itself. Typical cycles from yard to ship are about 1 to 2 Km. A total of 150 beacons were set up to cover the operation area. In the yard area, these were placed along roadways between container stacks at intervals of approximately 40 m. Elsewhere, the beacons were placed so as to tessellate the ground area with a triangle grid of base dimension 50 m (the triangular grid maximizes location accuracy). Typically, three beacons are available for viewing at any location.

The most immediate problem faced was the very high level of clutter experienced by the radar in a container environment. The AGV had to move accurately through roadways between stacks of metal containers 18 m tall. A great deal of time was spent adjusting radar sensitivity and improving gating procedures to overcome this

problem. Major changes in reflectivity of the environment caused by rain and snow also caused problems. Some of these were overcome by employing probabilistic gating and association algorithms. The use of very tight gating procedures, at the expense of low matching rates, also substantially improved system reliability.

A second important problem was the detection of navigation faults due to mismatching and drifting of system components and estimates. The difference in estimates before and after the update were used to identify abrupt faults due to mismatching and sudden load changes. However, low-frequency drifts in component values (radar calibration, steer slippage, geometry change with temperature, etc.) proved very difficult to isolate. Augmented estimators were developed to accommodate this type of condition and to monitor changes in slowly varying parameters. These worked well enough to overcome immediate problems.

A final issue is that of safety. The vehicle system has proved to be reliable and has operated safely during fault conditions.⁵ However, a continual worry has been the need to provide some guarantee of vehicle safety, something that ultimately is very difficult to measure. This is particularly true of the collision detection system, which must operate with absolute reliability regardless of environment conditions. In particular, for commercial reasons, the vehicle was run alongside the existing manned fleet. Although no rules (in Europe) have yet been developed

5. We have had one minor accident, a side crash into a car, resulting from incorrect initialization and from the car being hidden in the radar blind spot at the side of the vehicle.

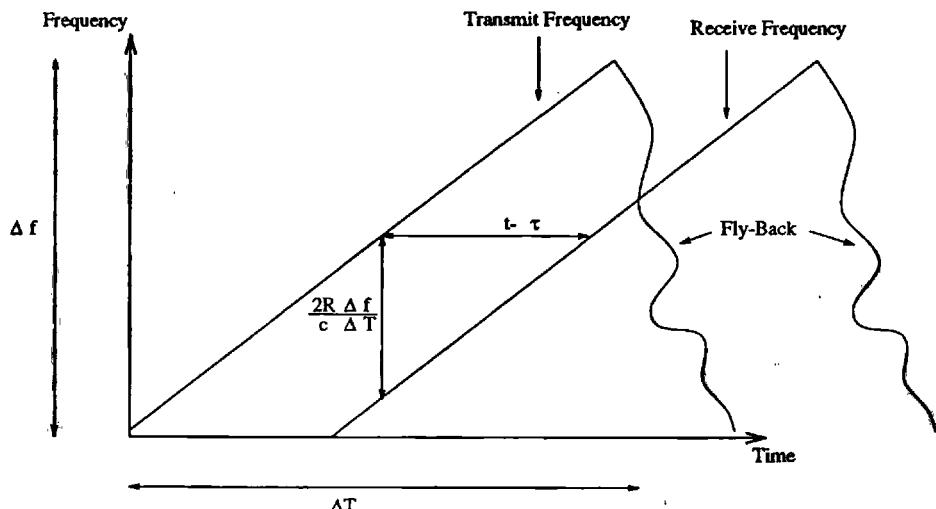


Fig. 20. Graph of transmittal and received frequencies.

to govern AGV systems of this type, it is to be expected that regulatory authorities will require some quantifiable means of specifying safety and reliability levels for such autonomous systems when operating in manned environments. The ability to operate mixed-mode automated and manned systems and the development of quantifiably reliable collision detection systems is likely to be crucial in the commercial development of other application areas.

Some anticipated problems turned out to be much less important than expected. In particular, the navigation system processing was rarely a source of system failures (this tended to be dominated by faults in other vehicle components such as hydraulic system sensors, controllers, etc.). The loss of beacons through obscuration or from deliberate vandalism (by dockers) was also not a problem and rarely resulted in navigation performance below specification.

6. Conclusions

The AGV system described in this article represents an important step in developing the technology needed to commercialize advanced mobile robot systems. The system exploits what is seen as the "state of the art" in sensing and navigation technology.

The development of the AGV has highlighted a number of important issues that will need to be addressed by the robotics community as techniques developed in the laboratory mature to become commercial systems. In particular, the need to develop and build reliability in the context of autonomous systems is seen as crucial. This requires a careful rethink of the way in which techniques and systems are implemented, and the development of quantifiable measures of performance and reliability at both the subsystem level and as part of an overall vehicle.

A number of specific points of further work have also been identified as being important to future AGV systems. The development of improved navigation system vehicle process models is seen to be of considerable future importance. Improved models will provide better prediction information, which will both reduce sensing requirements and improve position estimation capability at high speeds. Further, such models may allow the estimation of additional vehicle parameters such as tire forces, which will allow vehicles to operate effectively in a wider range of rough terrain environments. The development of improved beacon to observation association algorithms could also be of long-term benefit. This would help in improving beacon detection rates and in reducing the possibility of incorrect association. Ultimately, it would be possible to apply such methods to the problem of high-speed radar-based navigation using terrain data rather than a beacon system.

Appendix: Radar Operation Principle

The principle of operation of an FMCW (chirp) radar is summarized in Figure 20 (see Skolnik (1980) for further details). A pulse of radiation is transmitted from the radar. The frequency $f_t(t)$ of radiation in this transmitted pulse is varied in a linear fashion as

$$f_t(t) = f_0 + t \frac{\Delta f}{\Delta T},$$

where f_0 is the base frequency (77 GHz), Δf the swept bandwidth, and ΔT the pulse width. Consider a target at a range R . The time of flight τ out to this target and back to the radar is simply $2R/c$, where c is the propagation

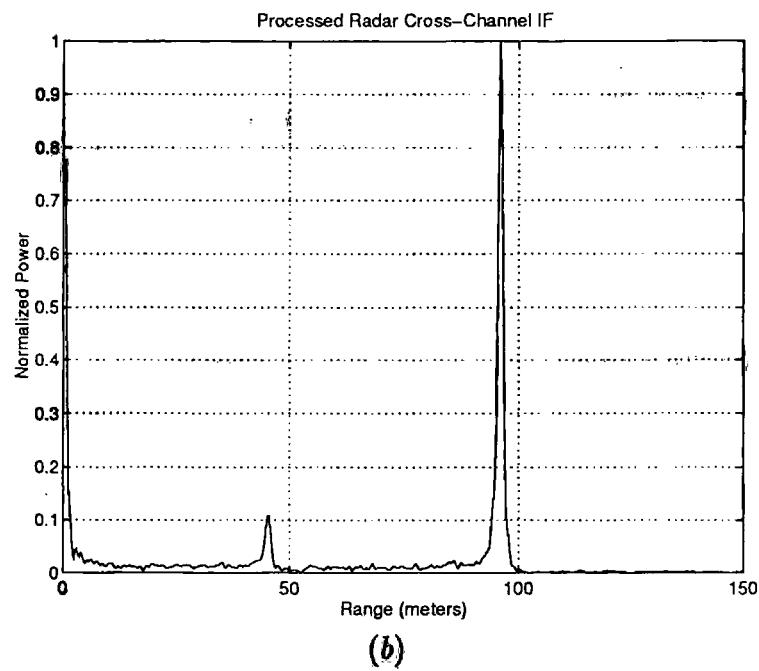
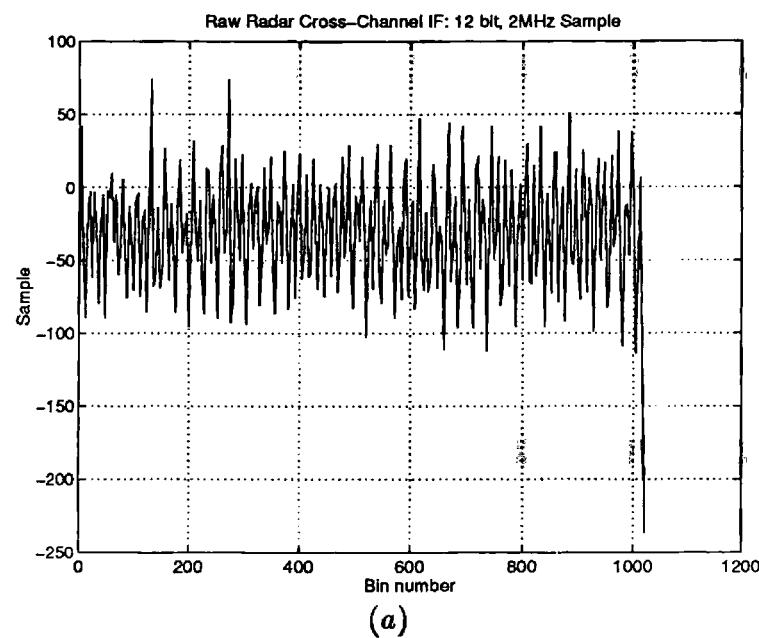


Fig. 21. (A), Raw (IF) radar signal return. (B), Frequency content of return signal. The small return is a person, the larger return is a 100-m² cross section trihedral radar reflector (the physical dimensions of a 100-m² reflector is only 30 cm on base).

speed of the radiation. Thus, the frequency $f_r(t)$ of the reflected radiation detected by the receiver will be given by

$$\begin{aligned} f_r(t) &= f_0 + (t - \tau) \frac{\Delta f}{\Delta T} \\ &= f_0 + \left(t - \frac{2R}{c} \right) \frac{\Delta f}{\Delta T}. \end{aligned}$$

The difference between transmitted and received frequencies, called the *intermediate frequency* (IF), is directly proportional to target range as

$$f_t(t) - f_r(t) = \frac{2R}{c} \frac{\Delta f}{\Delta T},$$

or

$$R = [f_t(t) - f_r(t)] \frac{c \Delta T}{2 \Delta f}.$$

Practically, the IF frequency is obtained by simply multiplying the transmitted and received signals together and filtering out the upper harmonic ($f_t(t) + f_r(t)$).

The smallest frequency δf that can be resolved by the radar unit is limited by the pulse duration as $\delta f = 1/\Delta T$. Thus, the range resolution δR of the radar is proportional only to the swept bandwidth

$$\delta R = \frac{c}{2 \Delta f}.$$

Figure 21A shows a typical IF signal consisting of 1024 discrete samples. To obtain frequency information from this signal, a discrete (fast) Fourier transform (FFT) is performed yielding the result in Figure 21B. The figure shows a complete range plot out to 150 m obtained from one pulse. The signal processing unit for the radar needs to compute these range plots in the pulse time (500 μs). To compute a valid range plot, two 1-K FFTs (one for each polarization channel), associated widowing, magnitude-squared, and thresholding operations must be performed. This requires substantial processing hardware and adds to the cost of this type of radar unit.

References

- Alexander, J. C., and Maddocks, J. H. 1989. On the kinematics of wheeled mobile robots. *Int. J. Robot. Res.* 8(5):15–27.
- Bar-Shalom, Y., and Fortmann, T. E. 1988. *Tracking and Data Association*. New York: Academic Press.
- Chatila, R., and Laumond, J. P. 1985. Position referencing and consistent world modeling for mobile robots. *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 138–143.
- Crowley, J. 1989. World modeling and position estimation for a mobile robot using ultra-sonic ranging. *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 674–681.
- Gelb, M. 1974. *Applied Optimal Estimation*. Cambridge, MA: MIT Press.
- Halme, A., and Koskinen, K. 1993. An approach to automate the mobility of working machines in outdoor environments. In *Int. Conf. Intelligent Robots and Systems (IROS)*.
- Julier, S., and Durrant-Whyte, H. 1995. Navigation and parameter estimation of high-speed road vehicles. *Proc. IEEE Int. Conf. Robotics and Automation*.
- Kane, T. R., and Levinson, D. A. 1990. *Dynamics: Theory and Application*. New York: McGraw-Hill.
- Koskinen, K., Makela, H. M., Rintanen, K., and Halme, A. 1993. An experimental autonomous land vehicle for off-road piloting and navigation research. *IEEE Int. Conf. on Systems, Man and Cybernetics*.
- Leonard, J. J., and Durrant-Whyte, H. F. 1992. *Directed Sonar Navigation*. Boston: Kluwer Academic Press.
- Maybeck, P. S. 1979. *Stochastic Models, Estimation and Control*, Vol. I. New York: Academic Press.
- Skolnik, M. I., 1980. *Introduction to Radar Systems*. New York: McGraw-Hill.
- Thorpe, C., Hebert, M., Kanade, T., and Shafer, S. 1988. Vision and navigation for the Carnegie-Mellon Navlab. *IEEE Trans. Pattern Anal. Machine Intell.* 10(3):362–373.
- Thorpe, C. 1989. *Vision and Navigation: The CMU NAVLAB*. Boston: Kluwer Academic Publishers.
- Viersma, T. J. 1990. *Analysis, Synthesis and Design of Hydraulic Servosystems and Pipelines*. Delft University Press.