Simultaneous Localisation and Map Building

Michael Csorba

Robotics Research Group
Department of Engineering Science
University of Oxford

To my Parents.

Michaelmas Term, 1997
Robotics Research Group
Department of Engineering Science
University of Oxford

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfillment of the requirements for the degree of **Doctor of Philosophy**. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Michael Csorba
Balliol College, Oxford

Copyright ©1997 Michael Csorba
All Rights Reserved

Abstract

Michael Csorba
Balliol College

Doctor of Philosophy
Michaelmas Term 1997

Simultaneous Localisation and Map Building

This thesis examines the problem of localising an Autonomous Guided Vehicle (AGV) travelling in an unknown environment. In this problem, the AGV faces the dual task of modeling the environment *and* simultaneously localising its position within it. The Simultaneous Localisation and Map Building (SLAM) problem is currently one of the most important goals of AGV research. Solving this problem would allow an AGV to be deployed easily, with very little initial preparation. The AGV would also be flexible and able to cope with modifications in the environment. A solution to the SLAM problem would enable an AGV to would be truly "autonomous."

The thesis examines the SLAM problem from an estimation theoretic point of view. The estimation approach provides a rigorous framework for the analysis and has also proven to be successful in actual applications.

The most significant contribution of this thesis is to provide, for the first time, a detailed development of the theory of the SLAM problem. It is shown that correlations arise between errors in the vehicle and the map estimates, and these correlations are identified as fundamentally important to the solution of the SLAM problem. It is demonstrated that ignoring these correlations results in the loss of the fundamental structure of the SLAM problem and leads to inconsistency in map and vehicle estimates. The evolution of the map is also examined. It is shown that the map cannot diverge, in particular it is shown that the estimate of individual features in the map, groups of features and relative distances between features cannot diverge. The thesis further examines the theoretical limit to the accuracy of the map and the localisation of the vehicle. The relative nature of the observations are exploited to identify the existence of a limit, and the determination of the limit is also presented.

Together, these results show that it is theoretically possible to start at an unknown location in an unknown environment and to build a map by which to navigate.

It is identified in the thesis that the full solution to the SLAM problem results in a large storage requirement and heavy computational burden. This leads to the need to find alternative solutions to the SLAM problem that have both a sound theoretical basis and which also permit practical implementations.

Two such solutions to the SLAM problem are developed in this thesis. The first is the Bounded Region (BOR) filter. The BOR filter makes the assumption of bounded errors in the observation and vehicle models, which permits the filter to be formulated without the need to consider correlations. Consequently the storage and the computational requirement of the BOR filter are very low.

The second alternative solution developed in this thesis is the Relative (REL) filter. The REL filter estimates the relative distance between features and the relative angle formed by groups of three features. Such estimates can be formed without the need to consider correlations between estimates. The REL filter therefore has a low storage requirement and computational burden.

The thesis also investigates the performance of these solutions to the SLAM problem in a real implementation. An AGV equipped with a laser rangefinder builds a map of features in a corridor and simultaneously localises its own position using the map. Each approach is investigated in detail, in particular the evolution of the map is examined and important aspects of the localisation are identified.

Acknowledgements

Firstly, I'd like to thank Hugh Durrant-Whyte for his help and enthusiasm during the past three years. I have benefited greatly from his supervision.

I am also very greatful to British Aerospace for funding my research, and for giving me the opportunity to discuss it with them at meetings in Bristol, Orlando and Sydney.

Simon Julier deserves a special thanks for ploughing through previous versions of this thesis, and making many useful corrections and suggestions. Special thanks also go to Teun Hendriks and Tom Murphy who have been great supervisors during my internships at Philips in New York. I have enjoyed working with them very much.

I have been very fortunate to know the old group at Terrapin Hut: Mike and Andrew Stevens, Stephe Borthwick, Jeffrey Uhlmann, Simon Cooper, Ben Quine, Rob Smith, Luke Forster, Arthur Mutambarra, Gordon Kao and Mariano Fernandez. Their mischief made my time in the Robotics Research Group a very happy one.

In Sydney, I would like to thank Peter Dulimov, for sharing four flats with me in two years, and Fiona, for the special time spent together in Cronulla, Pebbly Beach and Noosa. Thanks also to the Mechatronics Group: Steve Scheding, Daniel Pagac, Mohammad Bozorg, Paul Newman, Salah Sukkarieh, Gaius Chan, Steve Clark, Monica Louda, Keith Willis, Tim Bailey, Adrian Bonchis, Nquyen Hong Qang, Phil Avery, Tobias Karlsson, Quang Ha, Tuan Anh Le, Rajmohan Madhovan and Eduardo Nebot.

Sydney would not have been the same without Pedro, Charlie and the dance lessons at La Vina. Thanks to them as well.

Finally, I would like to thank my brother and my parents for their support and for being a loving family.

Contents

List of Tables

Notation

x_v	x coordinate of the vehicle
y_v	y coordinate of the vehicle
ϕ_v	orientation of the vehicle
$\mathbf{x}_v$	vehicle state vector
$\hat{\mathbf{x}}_v$	estimate of the vehicle state vector
$\tilde{\mathbf{x}}_v$	error in the estimate of the vehicle state vector
$\mathbf{P}_{vv}$	covariance matrix of the vehicle state estimate
x_i	x coordinate of feature number i
y_i	y coordinate of feature number i
p_{i_j}	j^{th} dimension of feature number i
$\mathbf{p}_i$	feature number i state vector
$\hat{\mathbf{p}}_i$	estimate of feature number i state vector
$\tilde{\mathbf{p}}_i$	error in the estimate of feature number i state vector
$\mathbf{P}_{ii}$	covariance matrix of state estimate of feature number i
$\mathbf{x}_{mav}$	map augmented vehicle state vector
$\hat{\mathbf{x}}_{mav}$	estimate of the map augmented vehicle state vector
$\tilde{\mathbf{x}}_{mav}$	error in the estimate of the map augmented vehicle state vector
$\mathbf{P}_{mav}$	covariance matrix of the estimate of the map augmeented vehicle state
$\mathbf{P}_{vi}$	covariance between the vehicle state estimate and the estimate of feature number i
$\mathbf{P}_{nm}$	covariance between component n and m of the map augmented vehicle state estimate
σ_{any}	variance of an arbitrary dimension of the map augmeented vehicle state estimate
σ_{map}	variance of an arbitrary dimension of an arbitrary feature within the map augmented vehicle state estimate
$\Delta\mathbf{p}_{ij}$	relative distance vector of the position of feature number i with respect to feature number j
$\Delta_m p_{ij}$	the m^{th} dimension of $\Delta\mathbf{p}_{ij}$
$\Delta\mathbf{P}_{ij}$	covariance of the estimate of $\Delta\mathbf{p}_{ij}$
$\Delta_m P_{ij}$	the variance of the m^{th} dimension of $\Delta\mathbf{p}_{ij}$
$\mathbf{P}_0$	additive factor in $\mathbf{P}_{mav}$
$\mathbf{P}_E$	additive factor in $\mathbf{P}_{mav}$ representing a lower limit to the covariance

$\mathbf{f}_{\mathbf{x}_v}$	vehicle process model
$\nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}$	Jacobian of the vehicle process model with respect to the vehicle state
$\nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}}$	Jacobian of the vehicle process model with respect to the process noise
$\mathbf{f}_{\mathbf{x}_{mav}}$	process model of the map augmented vehicle
$\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{x}_{mav}}$	Jacobian of the map augmented vehicle process model with respect to the map augmened vehicle state
$\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}}$	Jacobian of the map augmented vehicle process model with respect to the process noise
V	velocity input
γ	steer input
v_v	velocity component of the process noise
v_γ	steer angle component of the process noise
$\mathbf{h}_{\mathbf{x}_v}$	observation model for the observation of only the vehcile
$\mathbf{h}_{\mathbf{p}}$	observation model for the observation of a feature
$\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}$	observation model for the observation of a feature relative to the vehicle
$\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the vehicle state
$\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the state of feature number i
$\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the observation noise
$\mathbf{h}_{\mathbf{x}_{mav},i}$	observation model for the observation of feature number i relative to the vehicle in the case of the map augmented vehicle state
$\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_{mav}}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the map augmented vehicle state (simplified to $\mathbf{H}_i$)
$\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_v}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the vehicle state (simplified to $\mathbf{H}_v$)
$\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_i}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the state of feature number i (simplified to $\mathbf{H}_p$)
$\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}}$	Jacobian of the observation model for the observation of feature number i relative to the vehicle with respect to the observation noise
$\mathbf{z}_{\mathbf{x}_v}$	observation of only the vehicle
$\mathbf{z}_{\mathbf{p}_i}$	observation of only feature number i
$\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}$	observation of feature number i relative to the vehicle
$\mathbf{z}_{\mathbf{x}_{mav},i}$	observation of feature number i relative to the vehicle in the map augmented vehicle case
r	range observation
θ	bearing observation
w_r	range component of the observation noise
w_θ	bearing component of the observation noise

$\hat{\mathbf{x}}_v\|_b$	bounded estimate of the vehicle state
$\hat{x}_v\|_{min}$	minimum bound on the x coordinate of the vehicle
$\hat{x}_v\|_{max}$	maximum bound on the x coordinate of the vehicle
$\hat{y}_v\|_{min}$	mniimum bound on the y coordinate of the vehicle
$\hat{y}_v\|_{max}$	maximum bound on the y coordinate of the vehicle
$\hat{\phi}_v\|_{right}$	'right' bound on the vehicle orientation
$\hat{\phi}_v\|_{left}$	'left' bound on the vehicle orientation
$\hat{\mathbf{p}}_i\|_b$	bounded estimate of the state of feature number i
$\hat{x}_i\|_{min}$	minimum bound on the x coordinate of feature number i
$\hat{x}_i\|_{max}$	maximum bound on the x coordinate of feature number i
$\hat{y}_i\|_{min}$	mniimum bound on the y coordinate of feature number i
$\hat{y}_i\|_{max}$	maximum bound on the y coordinate of feature number i
$\Delta\mathbf{x}_v$	relative motion of the vehicle
Δx_v	relative motion of the vehicle in the direction of the vehicle heading at the start of the motion
Δy_v	relative motion of the vehicle perpendicular to the direction of the vehicle heading at the start of the motion
$\Delta\phi_v$	orientation change of the vehicle with respect to the heading of the vehicle heading at the start of the motion
$\mathbf{f}_{\Delta\mathbf{x}_v}$	restricted vehicle process model for the relative approach

Chapter 1

Introduction

1.1 The Objectives of the Thesis

This thesis examines the problem of localising a vehicle in a previously unexplored environment. The problem is addressed by using the vehicle to build a map of the environment and simultaneously using the map to localise the vehicle, leading to the Simultaneous Localisation and Map Building (SLAM) problem.

The main objectives of the thesis, in the order of importance, are:

- To gain a theoretical understanding of the problem of simultaneous localisation and map building, specifically of how a solution can be formulated.
- To identify any significant characteristics of a theoretical solution, in particular whether divergence will occur and whether there is a limit to the accuracy of the map and the localisation of the vehicle.
- To develop practically feasible solutions to the SLAM problem.
- To evaluate the available solutions in an actual implementation.

The solution to the simultaneous localisation and map building problem is, in many respects, a "Holy Grail" of the mobile robotics research community. The ability to place a mobile robot at an unknown location in an unknown environment and then have it build a map and use this map to navigate would indeed make such a robot "autonomous." This thesis, for the first time, elucidates a rigourous mathematical solution to this problem, describes its structure and properties, and demonstrates how this leads to practical simultaneous localisation and map building algorithms.

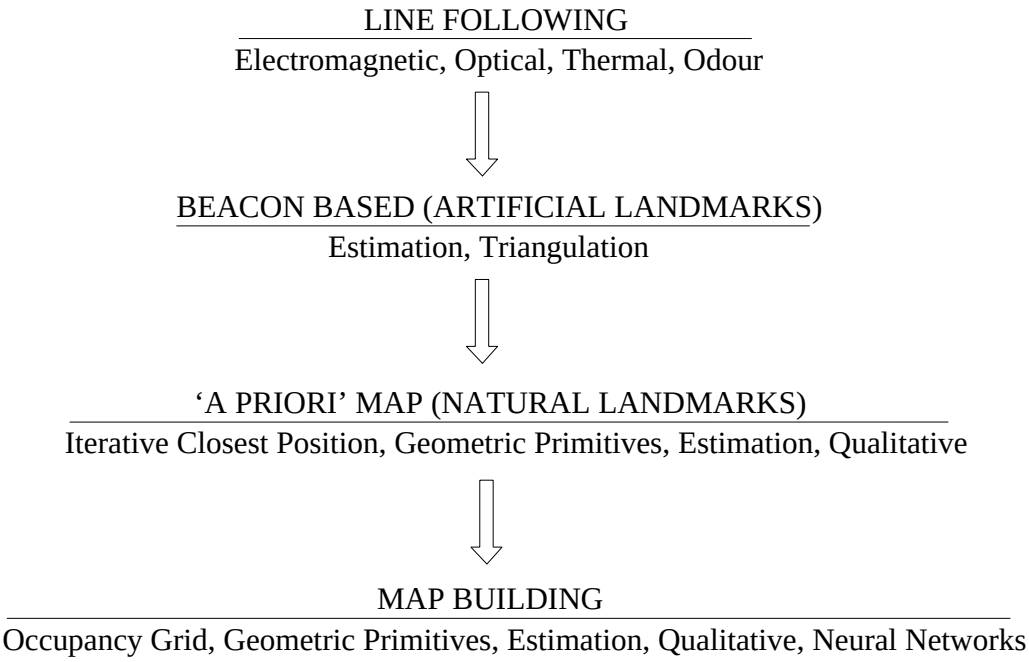

Figure 1.1: The Evolution of Autonomous Navigation.

1.2 The Navigation of Autonomous Vehicles

A fundamental competency required of a mobile robot, or Autonomous Guided Vehicle (AGV), is the ability to know its own position. All actions that an AGV is likely to perform depend on this. For example, planning and executing a path in an environment requires continuous knowledge of the vehicle location. In certain applications of AGVs, such as underwater exploration [58], the failure of this competency may even lead to the loss of the AGV.

The importance of AGV navigation has been recognised for some time, and in the last decade much research effort has been expended to achieve autonomous navigation.

1.2.1 Early Localisation Systems

An evolutionary flow diagram of the navigation of AGVs is given in Figure 1.1. Using successful applications as a guideline, line following robots in the manufacturing industry may be seen as the birth of AGV navigation systems. Line following is the process of guiding an AGV along a fixed path using a track that the AGV can identify and follow. This is a simple navigational method that requires only very simple sensing and processing resources. The method is now so common place that the construction of line following AGVs is described in hobby books [45].

Tsumura [90], for example, presents a survey of line following AGVs in Japanese factories in 1986. According to Tsumura, over 90% of AGVs in factories in 1986 were guided by wires buried in

the floor of factories. The wires carry an alternating current and the AGV follows the wire using an induction coil. Various forms of line following are investigated by researchers to this day. Deveza [27] and Russel [80] describe an odour trail following AGV, and a heat trail following AGV is described in [79].

line following methods have also been implemented over long distances. An example is the Rapidly Adapting Lateral Position Handler (RALPH) of the Carnegie Mellon University [75], which uses curbs and lane markings to keep within the lane of a road. This system has successfully guided a vehicle from New York to Los Angeles.

However, approaches that rely on line following have significant disadvantages. A major disadvantage is that the AGV can only follow a line, and its operation is therefore severely restricted. A further disadvantage is that the AGV will not know where its current position is along the track. A possible solution to this problem is to place bar codes on the track [90], but that allows localisation only at specific points along the track. Generally, the navigational capabilities of a line following AGV are extremely limited.

A further problem of line following strategies is the cost and time needed to lay tracks for the AGV. In some cases a line following strategy may be impossible to implement, for example due to poor ground conditions [28]. In other cases several AGV paths may need to cross each other, confusing the tracking sensor on the AGV.

Line following strategies are also inflexible since considerable new investment in capital and time is needed to modify the path of the vehicle. They are also very fragile. If the vehicle loses contact with the track, it can no longer navigate. This condition may arise due to a damaged or obscured track, but could also result if the AGV approaches a turn too quickly and skids as a result.

An alternative navigation method that does not suffer the problems outlined above relies on the localisation of the AGV. Localisation differs from line following in that it attempts to specify the position and orientation, the pose, of the vehicle within the environment rather than make small corrections on its motion. The AGV is not limited to moving along a fixed path, instead any position within the environment is permitted.

A limited form of localisation is possible using odometry or an Inertial Navigation System (INS) [55]. Odometry, which counts the revolution of the wheels, and INS, which measures the forces experienced during accelerations, allow the vehicle to localise by deducing (through the integration of the measurements) its pose from its movements. Odometry was already known to the Romans and drawings of odometers were made by Leonardo da Vinci [21]. However, both odometry and INS suffer from the accumulation of errors during their operation, and therefore lead to unbounded localisation errors [12].

1.2.2 Beacon Based Localisation

The first successful localisation systems employed beacons to determine the position of the vehicle. Beacon based localisation systems rely on a set of beacons placed at known positions in the environment. The AGV is equipped with a sensor that can observe the beacons, and the navigational system uses these observations and knowledge of the beacon positions to localise the position of the vehicle. Unlike odometry and INS, the accuracy of the localisation does not deteriorate with time in a beacon based system. This is due to the fact that the sensor on the AGV provides observations of position, rather than observations of motion, and therefore is not subject to the accumulation of "integration errors."

Beacon based localisation systems can be categorised according to the type of signal by the sensor, which relates to the beacon characteristics, and according to the type of information processing employed by the system.

The main types of signals used by beacon based localisation systems are infrared laser, ultrasound and millimeter wave radar. Infrared laser is used, for example, by commercially-available General Electric Company - Caterpillar (GEC-CAT) AGV [14] and the commercially-available Netzler & Dahlgren Company (NDC) AGV [73]. In both systems, the AGV is equipped with a rotating infrared laser emitter/receiver, and the beacons are reflective strips placed at known positions in the environment. The angle of the sensor is registered when a laser reflection is observed, thus giving the bearing of the observation.

Laser is by far the most widespread signal used in beacon based localisation systems. Some of the companies currently using laser in a beacon based localisation system include GTC Castrovillari in Italy, Tetra Pak in Singapore, Morris Automation in England and Costerplast in Italy [73].

The second common signal type is ultrasound. This system usually relies on active (emitting) rather than passive (reflecting) beacons. Ultrasound is easily reflected by most objects in typical AGV environments. Consequently, passive beacons would be difficult to identify using ultrasound. Another advantage of active beacons is that the transmitted signal may contain a specific code to identify the emitting beacon [61], or the beacons may identify themselves by transmitting in a fixed sequence with the first beacon emitting a modified signal to the others [50].

The third type of signal used in beacon based localisation systems is the millimeter wave radar (MMWR). Possibly the first example of its use in a beacon based localisation system was developed by Fire Fly Ltd on the FRAIT AGV [28]. The MMWR signal is reflected by metallic beacons placed at known locations in the environment, and the radar emitter/receiver provides both range and bearing to the beacons. The determination of range using MMWR signal is described, for example, by Skolnik [84]. The FRAIT AGV was developed in an effort to automate a port, and had the task

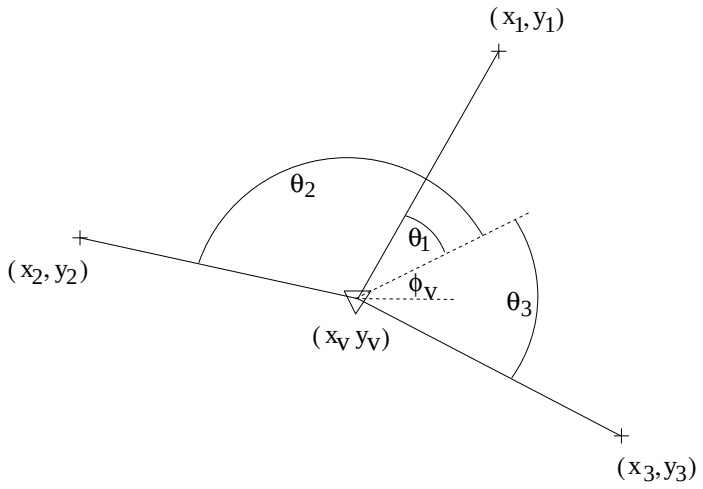

Figure 1.2: Triangulating on three observations.

of transporting containers between the quay area and the loading yards.

Two main categories of beacon based localisation systems can be identified according to the type of information processing employed by the system. These are triangulation and estimation.

Triangulation involves combining several bearing observations to deduce the position of the AGV, using the assumption that the position and orientation, the pose, of the vehicle does not change between the observations. The observations supply constraints on the vehicle pose. The number of constraints needed to uniquely determine the pose of the vehicle depends on the number of degrees of freedom of the pose. For a vehicle pose defined in terms of a position (x_v, y_v) and an orientation ϕ_v, three observations θ_1, θ_2 and θ_3 are needed to uniquely define the pose, as shown in Figure 1.2. Triangulation is the most widespread method used to localise an AGV. The NDC AGV uses triangulation in the applications listed in [73].

An alternative approach to triangulation is to form an estimate of the vehicle pose. Estimation relies on a model of the vehicle and the sensor. The estimate is updated each time a new observation is made, and the vehicle is localised from observation to observation using a prediction of the vehicle pose based on the vehicle control inputs, as shown in Figure 1.3.

The Kalman filter [47, 48, 66] is widely used to form a probabilistic estimate of the AGV pose, according to Bayesian Estimation Theory [5], in terms of a mean estimate and the covariance of the estimate.

The Kalman filter is used in the GEC-CAT AGV to localise the vehicle based on bearing-only observations [14], and is used by the FRAIT vehicle to localise based on range-bearing observations [28].

A non-probabilistic estimate of the vehicle pose can also be formed, for example by expressing

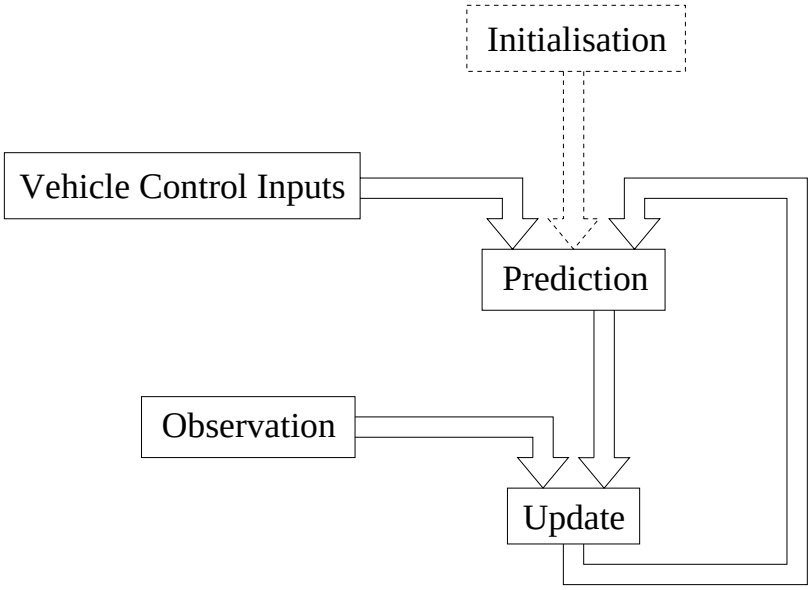

Figure 1.3: The estimation process.

the estimate in terms of bounded regions [83]. The true vehicle pose is guaranteed to be captured within the bounds of the estimate. This is examined, for example, by Hanebeck in [37], where a bounded pose estimate of an AGV is formed based on bearing-only measurements.

A common problem of beacon based localisation systems is that beacons have to be installed. This may be expensive and may also increase the deployment time of the AGV. A further problem of using beacons is that the beacons may get damaged or obscured, compromising the performance of the AGV.

The Global Positioning System (GPS) [22, 30] is a form of beacon based localisation using active beacons. GPS based localisation relies on the reception of signals emitted by several GPS satellites. The signals contain information to identify the satellite emitting the signal and also contain a time stamp. Knowing the position of the satellites and using the time-of-flight of the signal obtained from the time stamps, the position of the GPS receiver on the AGV, and hence the AGV position, is determined. The principle advantage of using GPS over land based beacons is that the GPS signal is already available and therefore it is not necessary to install beacons. This reduces the deployment cost and deployment time of the system. Further, a GPS system is less susceptible to damage since the satellites, the beacons of the GPS, are not easily accessible.

However, the signal from at least four satellites are needed to resolve the position completely, but often this is not available due to mountains, buildings or dense foliage [77]. A further problem is the accuracy and the frequency of GPS fixtures. Crow [22] reports an accuracy of 0.5m and a lag of 1.5s of GPS fixes. Durrant-Whyte also reports problems with GPS [28], specifically with its

reliability and accuracy.

1.2.3 Localising With An 'A Priori' Map

A significant improvement to the cost effectiveness and flexibility of beacon based localisation methods is achieved when naturally occurring landmarks are used as beacons [87, 10]. Using natural landmarks eliminates the cost of installing beacons, and landmarks are usually also more robust to damage. However, it is then often necessary to identifying landmarks in the received signals. The identification, referred to as feature extraction, has been the topic of considerable research [70, 92, 76].

Navigating using natural landmarks has also found applications in missile guidance [42]. The American Cruise Missile uses images of the terrain acquired prior to the mission to guide itself to the target. Many navies have terrain maps of the sea bed that submarines use to navigate while submerged, and cartographic databases are available of many areas of the world [40]. More recently, NASA's Microrover Pathfinder that landed on Mars in July 1997 used terrain navigation, though not autonomously. A video image of Pathfinder taken by a camera on Sojourner, the landing craft, was sent back to mission control on Earth. A graphical model of Pathfinder was then superimposed on the image to localise Pathfinder on the Martian terrain [64].

Parts of existing structures can also serve as natural landmarks. Ebihara [49] describes a mobile robot which navigates inside a nuclear reactor and uses the naturally occurring features within the reactor as beacons. Leonard [56] and Cox [20] describe an AGV using features in a room detectable by a sonar and an optical rangefinder respectively.

Features can sometimes provide more information than just a range and a bearing. Baeg [3] measures the orientation of features and uses the measurement to improve the localisation of the AGV.

Researchers have also examined how to select landmarks to increase the accuracy of localisation [89, 10]. Sutherland and Thompson argue in [89] that the vehicle path should follow routes where better localisation is possible due to more favourable landmarks.

Several approaches have been used to obtain the vehicle pose based on the data supplied by the sensors. The Iterative Closest Position (ICP) [8] method considers the localisation problem as a correspondence problem. The observation made by the sensor is matched with the 'a priori' map, with each iteration producing a better match. Zhang [94] identifies two ICP methods. One method, called the 'primitive-based approach,' extracts a set of primitives, such as lines and corners, and then matches the pose of the primitives extracted from the observation to the pose of the primitives in the 'a priori' map. The second method identified by Zhang, called the 'surface-based

approach,' considers the observation as a whole and attempts to match it with the map according to a correlation measure. This approach is also found in [69].

Estimating the vehicle pose using the Kalman filter has also been used in natural landmark based approaches. If landmarks can be identified, they can be treated as beacons. The landmarks are usually selected and identified via geometric primitives. This approach is very popular and can be found in [57, 26, 91].

A third approach to localise an AGV using an 'a priori' map involves qualitative arguments. The maps used in qualitative methods are topological representations of the environment. Bertin in [6] argues that qualitative approaches are generally more robust than quantitative approaches.

All of the examples of localisation given above have an important common element: the environment is known to the AGV in terms of the 'a priori' map. As a consequence, the AGV is limited to operating within the known environment. Added to this restriction is the disadvantage that the environment has to be surveyed, or mapped, before the AGV can operate. If it is desired to extend the environment of operation, new areas have to be mapped. If the area of operation of the AGV is fixed, it may still be necessary to perform new surveys since features in the environment might change over time.

To overcome these problems, researchers have investigated the possibility of the AGV simultaneously building a map of the environment and using it to localise.

1.2.4 Building Your Own Map

The Simultaneous Localisation and Map Building (SLAM) problem is currently one of the most important goals of AGV research. Solving this problem would allow an AGV to be deployed easily, with very little initial preparation necessary. The AGV would also be flexible, able to cope with modifications in the environment. An AGV that has solved the SLAM problem would be truly "autonomous."

Since the aim of the map is to allow the AGV to localise, building a map of the environment is only part of the overall problem. Moreover, map building and localisation, as will be argued in this thesis, cannot be viewed as two separable tasks. However, most researchers in the past have considered map building a separate problem to localisation. This led to the development of several inconsistent solutions to the SLAM problem.

Elfes [29] describes an approach to map building based on occupancy grids. These grids divide the environment into individual cells. Initially each cell is labeled 'unknown.' As the AGV explores the environment, it assigns new labels, 'empty' or 'occupied', to the cells according to its findings. This approach is still popular today [33, 60, 93]. However, Kriegman argues in [51] that the occupancy

grid approach cannot cope with uncertainties inherent to the information available to the AGV. A further problem of occupancy grids is that the larger the environment, the more grids are needed.

Geometrical primitives have also been used to model the environment, for example by Ayache [2], where line and planar features are extracted from a video image and used to model the environment. Ayache uses the Kalman filter to estimate the variance of the primitives based on the variance of the observations and the variance of the displacement between the origins of the observations. Updating the motion of the AGV is also considered, but SLAM effects are not examined.

Qualitative methods have been used to build maps as well. Kuipers [52, 53] describes an algorithm that defines distinctive places and interconnecting paths and constructs a topological map. However, no examination into SLAM effects are given.

A topological map of the environment may also be constructed using neural networks [15, 38]. This is examined in Morellas [71], but again SLAM effects are ignored.

The first consistent solution to the SLAM Problem was published by Smith, Self and Cheeseman [85]. Although they do not examine the SLAM effects, their solution provides a consistent treatment of them. Smith *et al* employ estimation and the Kalman filter. The map building problem is considered an extension to the task of localising the AGV. The approach of Smith *et al* is important, because it applies rigorous estimation methods to the SLAM problem. This is a particularly promising approach due the successful history of applications of estimation [86]. Modifications of the approach have been also published [9] and the approach has also been implemented, for example by Moutarlier [72], who also addresses the problem of bias in the filter.

However, there is no development in the literature aimed at understanding the fundamentals of the SLAM problem. In particular, the effects arising due to simultaneously localisation and map building are not examined. This understanding is essential in providing an exact formulation of the problem, and developing consistent solutions to it. The fundamental effect of simultaneous localisation and map building is, as will be shown in this thesis, the introduction of correlations between the errors in the vehicle and the feature estimates. These correlations capture the essence of the SLAM problem.

The solution presented by Smith *et al* leads to a considerable demand on computation and storage, as identified in several publications [57, 78, 39]. Several attempts have been made to overcome this. In [57] thresholds are used to consider estimates with small variances as perfect information; in [65] it is assumed that the error in the AGV estimate is not correlated with the error in the landmark estimates; correlations are also ignored in [78].

All the modifications render the original solution inconsistent, and demonstrate the importance of understanding the SLAM problem. The understanding of the SLAM problem is necessary both from a theoretic and a implementation point of view.

1.3 Main Contributions of the Thesis

This thesis presents, for the first time, a theoretical investigation into the Simultaneous Localisation and Map Building (SLAM) problem. The thesis develops and analyses a rigourous mathematical solution to the SLAM problem, identifying intrinsic properties, and using the insights to develop practical consistent simultaneous localisation and map building algorithms.

The main contributions of this thesis, in order of importance, are:

- A detailed theoretical treatment of the estimation based solution to the SLAM problem, in particular identifying the importance of the correlations between the estimate errors.
- A derivation showing the convergence of the solution. It is shown that the map created by the estimation based solution is guaranteed not to diverge.
- Constructing the map based on observations made by the vehicle is shown to lead to a theoretical lower bound on map accuracy, which can only be surpassed by incorporating external terrain or vehicle data.
- Development of a non Bayesian, set theoretic, approach to the SLAM problem, which avoids many of the computational problems of the probabilistic estimation based approach, and provides a tractable solution to the SLAM problem.
- A new approach to the SLAM problem that uses the Kalman filter to build a relative map of the environment. This approach reformulates the SLAM problem to avoid the problem of estimate error correlations and thus provides a computationally tractable solution to the SLAM problem.
- The three approaches to the SLAM problem developed in the thesis are examined in an experimental implementation, verifying the theoretical developments and identifying practical considerations.

1.4 The Structure of the Thesis

Chapter 2 introduces the mathematical formulation of the localisation problem; the problem of determining the pose of a vehicle within an environment based on observations of the environment. The chapter justifies and develops the estimation theoretic approach to the problem, with the pose of the vehicle identified as a state that needs to be estimated. It further develops models for the vehicle and the sensor, and states the assumptions made about the errors in the models. The Kalman filter is introduced as an optimal statistical algorithm to be used in the estimation process. The

chapter then considers three special cases. The first case concerns the problem of estimating the state of a vehicle in a perfectly known environment. It is shown that the observations of a perfectly known environment are, in effect, indirect observations of the vehicle state itself. The second case considers a vehicle with perfectly known position traveling in an unknown environment. The problem addressed is the task of constructing a map of the environment. This is not a localisation problem, but its examination leads to essential insights into the SLAM problem. The third and last case combines the previous two cases. It considers the problem of simultaneously localising the vehicle and building a map of the environment. Thus the Simultaneous Localisation and Map Building (SLAM) problem is introduced.

In **Chapter 3** the SLAM Problem is examined in detail. The estimation approach to the problem is developed and used to show why so many previous attempts at solving the problem were unsuccessful. It is shown that the errors in all of the state estimates are correlated, and it is further shown that the correlations are fundamental to the SLAM problem. It is shown that neglecting to take account of the correlations leads to inconsistency. A detailed examination of the estimation approach is given. In particular, the evolution of the map of the environment is considered, leading to important conclusions regarding its convergence. The convergence is examined in terms of the whole map, individual features in the map, groups of features and relative distances between features. Further examinations are used to reveal a theoretical limit to the accuracy of the constructed map and the accuracy of the vehicle localisation. It is demonstrated that only additional information, such as independent knowledge of the terrain or the location of the vehicle, allows the accuracy of the map to improve past the identified limit. Finally, it is also pointed out in this chapter that having to take account of all estimate error correlations makes the solution computationally impractical in most real cases.

Chapter 4 addresses the impracticality of the estimation solution to the SLAM problem. In particular, two alternative solutions to the SLAM problem are developed. The first solution, referred to as the Bounded Region (BOR) filter, assumes that the errors in the vehicle and the sensor models are bounded. This allows the BOR filter to formulate a bounded estimate of the vehicle pose and a bounded estimate for each of the features in the environment. Consequently, the problem of correlated errors, associated with the estimation approach is not encountered. The BOR filter provides a practical solution to the SLAM problem.

The second alternative solution is referred to as the Relative (REL) filter. This filter makes the same assumptions about the errors in the vehicle and the sensor models as are made in the estimation approach. However, the filter constructs relative rather than absolute estimates. In particular, estimates of the relative distance between two features and estimates of the relative angles formed by three features. The approach is shown not to suffer from the correlation problem.

In **Chapter 5** the three primary map building algorithms, the estimation approach, the BOR and the REL filter are compared in an actual implementation. The chapter describes how the experimental scenario was constructed and how the operation of the robot and the sensors were verified before the three solutions to the SLAM Problem were tested.

Detailed examinations of the behaviour of the algorithms are given and discussed. In particular, the accuracy of the vehicle localisation and the accuracy of the constructed maps are examined. Individual characteristics of each algorithm are identified and, based upon the characteristics, conclusions about the applicability of the algorithms are reached.

Chapter 6 summarises the work of the thesis and provides final conclusions. In particular it identifies inadequacies of the implementations developed and suggests further areas of research in the SLAM problem.

Chapter 2

Background

2.1 Introduction

This chapter presents a mathematical formulation of the localisation problem. The localisation problem, as presented in this thesis, is defined as the problem of determining the position and orientation, the pose, of a vehicle traveling within an environment, based upon observations of the environment made relative to the vehicle. The particular approach taken to solve this problem is to form a probabilistic estimate of the vehicle pose.

The estimate of the vehicle pose is formed using the well-known Kalman filter algorithm. In Section 2.2 the Kalman filter is introduced. In Section 2.3 mathematical models for the vehicle, the environment and the sensor are developed. Section 2.4 examines the effect of using relative observations in the Kalman filter. In particular, Section 2.4.1 examines the operation of the Kalman filter in the context of vehicle localisation in a known environment. The operation of the Kalman filter is then examined in Section 2.4.2 for the reverse task: the construction of a map of the environment, based on observations made relative to the vehicle, when the pose of the vehicle is always known perfectly. Although this is not a solution to the localisation problem, it is investigated because it is an essential precursor to the more difficult problem of simultaneous localisation and map building, which is the main subject of this thesis. Simultaneous localisation and map building is introduced in Section 2.4.3 and some of the associated problems are described which have to be solved in order to solve the localisation problem in a general way. Section 2.5 summarises the results of this chapter.

2.2 The Kalman Filter

Estimation using the Kalman filter has been adopted in this thesis to the solve the localisation problem. The Kalman filter is a standard estimation tool used widely in the construction of navigational systems. A brief discussion of estimation using the Kalman filter is now given.

The general estimation process is formulated. The state to be estimated will be denoted by a state vector $\mathbf{x}$, but it is not immediately assumed to be the state of the vehicle. States other than the state of the vehicle, in particular the state of features in the environment, will need to be considered at a later stage.

2.2.1 The Process Model

In order to develop the Kalman filter algorithm, a process model is needed. The process model describes the evolution of the state $\mathbf{x}$ in time. It is usually derived in continuous-time form as

$$\dot{\mathbf{x}}(t) = \mathbf{f}^c \left[\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t), t\right], \tag{2.1}$$

where t denotes time, $\mathbf{u}(t)$ is the control input to the system and $\mathbf{v}(t)$ is the process noise. The process noise accounts for the perturbations to the system that have not been modelled explicitly.

Although the behaviour of the majority of real systems are most accurately modeled in continuous-time, a discrete-time formulation will be necessary in view of an implementation based on digital computers.

In discrete-time, only discrete steps $t_0, t_1, \ldots$ will be considered. To simplify notation, a time step t_k will be referred to only by the time index k. The discrete-time process model is derived by integrating the continuous-time process model between two consecutive time steps. The process noise and control input vectors are redefined to reflect the integration. The discrete-time process model is

$$\mathbf{x}(k+1) = \mathbf{f}\left[\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k), k\right]. \tag{2.2}$$

The process noise is assumed to be a Gaussian-distributed random variable which is zero mean and has covariance $\mathbf{Q}(k)$,

$$\mathbf{v}(k) = \mathcal{N}(\mathbf{0}, \mathbf{Q}(k)).$$

The assumption of zero mean process noise is justified by the fact that a mean other than zero indicates a potential to improve the process model and should be exploited [1]. The assumption of a Gaussian distribution is justified by the argument that the perturbations that have not been modelled explicitly rise as the combination of a large number of small perturbations due to many independent sources. Applying the central limit theorem [74] justifies the assumption that the combined perturbation has a Gaussian distribution.

The process model is assumed to model any deterministic behaviour of the state, thus the process noise at any time is assumed to be independent of the state of the system and the process noise at any other time,

$$\begin{aligned} \mathrm{E}\left[\mathbf{v}(i)\mathbf{x}^T(j)\right] &= \mathbf{0} \ \forall \ i,j \\ \mathrm{E}\left[\mathbf{v}(i)\mathbf{v}^T(j)\right] &= \mathbf{Q}(i)\delta_{ij}, \end{aligned}$$

where δ_{ij} is the Dirac delta function and $\mathrm{E}[\cdot]$ is the expectation operator.

2.2.2 The Observation Model

Observations of the state $\mathbf{x}(k)$ are made according to the observation model,

$$\mathbf{z}(k) = \mathbf{h}\left[\mathbf{x}(k), \mathbf{w}(k), k\right].$$

where $\mathbf{w}(k)$ is the observation noise, and plays a similar role in the observation model to the role played by the process noise in the process model. It accounts for effects that are not modelled explicitly, and is assumed again to be a random variable, with zero mean, Gaussian distribution and with a variance of $\mathbf{R}(k)$,

$$\mathbf{w}(k) = \mathcal{N}(\mathbf{0}, \mathbf{R}(k)).$$

The assumption of zero mean is again justified by the use of an observation model that exploits all deterministic properties of the observation. The observation noise in the observation model is, as with the process model, assumed to arise as the combination of noise components due to many independent sources. The central limit theorem is again used to justify the assumption that the combined noise will have a Gaussian distribution.

The observation model is assumed to model any deterministic behaviour of the observation, this is formally stated as the assumption that the observation noise at any time is independent of the state of the system and the observation noise at any other time. A further assumption is that the process and observation noise are also independent [1] [2],

$$\begin{aligned} \mathrm{E}\left[\mathbf{w}(i)\mathbf{x}^T(j)\right] &= \mathbf{0} \ \forall \ i,j \\ \mathrm{E}\left[\mathbf{w}(i)\mathbf{w}^T(j)\right] &= \mathbf{R}(k)\delta_{ij} \\ \mathrm{E}\left[\mathbf{w}(i)\mathbf{v}^T(j)\right] &= \mathbf{0} \ \forall \ i,j. \end{aligned}$$

2.2.3 The Estimation Process

The Kalman filter estimates a state using the process and observation models, together with the assumptions made about the process and the observation noise.

The estimate of the true state $\mathbf{x}(i)$ conditioned on the sequence of observations $\mathbf{Z}^j = \{\mathbf{z}(1), \ldots, \mathbf{z}(j)\}$ is denoted by $\hat{\mathbf{x}}(i|j)$. The Kalman filter finds the estimate that minimises the mean squared error in the estimate, and is therefore equal to the expected value of the state conditioned on the sequence of observations [66],

$$\hat{\mathbf{x}}(i|j) = \mathrm{E}\left[\mathbf{x}|\mathbf{Z}^j\right]. \tag{2.3}$$

The error between the true state $\mathbf{x}(i)$ and the estimated state $\hat{\mathbf{x}}(i|j)$ is denoted by $\tilde{\mathbf{x}}(i|j)$, and the covariance of $\hat{\mathbf{x}}(i|j)$ is denoted by $\mathbf{P}(i|j)$ [3],

$$\begin{aligned} \tilde{\mathbf{x}}(i|j) &\triangleq \mathbf{x}(i) - \hat{\mathbf{x}}(i|j), \\ \mathbf{P}(i|j) &= \mathrm{E}\left[\tilde{\mathbf{x}}(i|j)\tilde{\mathbf{x}}^T(i|j)\right]. \end{aligned} \tag{2.4}$$

The Kalman filter computes the estimate $\hat{\mathbf{x}}(i|j)$ and also computes the associated covariance $\mathbf{P}(i|j)$. However, the conditions given in Equations 2.3 and 2.4 may fail, due to, for example, an incorrect

[1] The assumptions of independence associated with any noise is, according to Langevin [32], page 6, justified by the "irregularity" of the noise.

[2] The assumptions of independence and zero-mean associated with the process noise and the observation noise are made for convenience. Non-zero means and correlations between the state and the process and the observation noise can also be incorporated [44].

[3] Strictly speaking, this is a mean-squared error, but, following the literature, it will be referred to as a covariance.

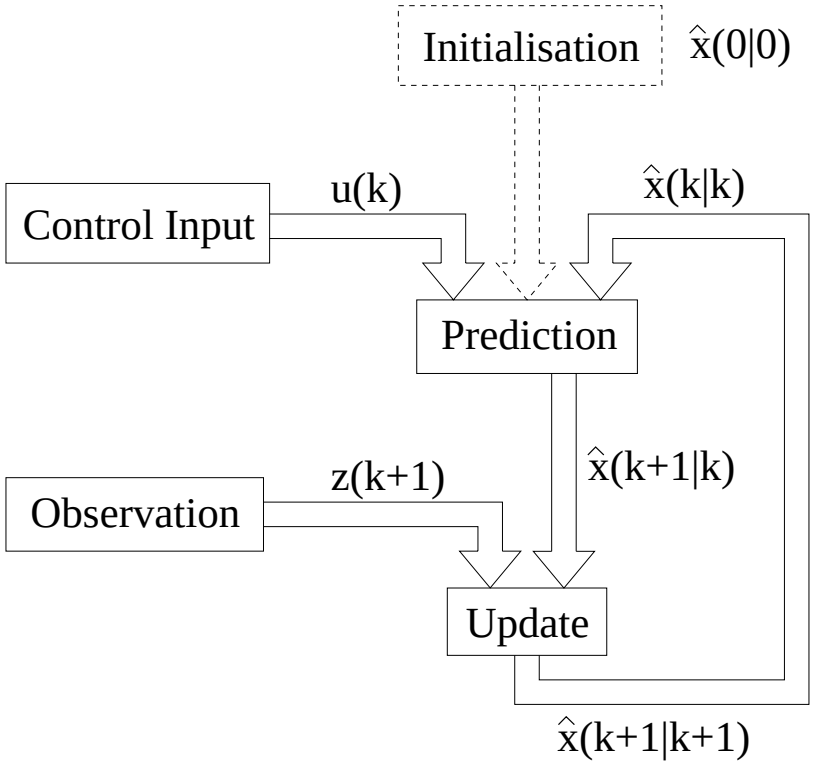

Figure 2.1: The estimation process.

implementation of the Kalman filter. A particular type of failure, called inconsistency [4] [46], will be defined as follows.

> In this thesis, an estimate $\hat{\mathbf{x}}(i|j)$ is said to be consistent with its covariance matrix $\mathbf{P}(i|j)$ if and only if $\mathbf{P}(i|j) - \mathrm{E}\left[\tilde{\mathbf{x}}(i|j)\tilde{\mathbf{x}}^T(i|j)\right]$ is positive semidefinite [5], and is inconsistent otherwise.

An iteration in the Kalman filter estimation process is now considered. The aim is to formulate an estimate $\hat{\mathbf{x}}$ of the state $\mathbf{x}$ at time step $k+1$, based only on the estimate at time step k and an observation made at time step $k+1$.

The iteration can be broken down into two stages as shown in Figure 2.1. A prediction stage, which forms an estimate of the state at time step $k+1$ based only on the estimate at time step k, and an update stage, which combines the predicted estimate at time step $k+1$ with the observation at time step $k+1$ to form an estimate at time step $k+1$. The iteration then repeats recursively.

The estimation process is examined in detail in Sections 2.2.4 – 2.2.6. In Section 2.2.4 the process and the observation models are assumed to be linear. This leads to a very simple prediction and update stage. In Section 2.2.5 the linearity restriction is removed and general non-linear process

[4] Inconsistency in an estimate can be interpreted as an unjustifiably high confidence in the estimate.
[5] A positive semidefinite matrix is defined in Section A.1.

and observation models are considered. Section 2.2.6 examines an important condition that must be satisfied in order to produce consistent estimates.

2.2.4 The Linear Case

In the linear case the process and the observation models are defined as

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \mathbf{v}(k), \tag{2.5}$$

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k). \tag{2.6}$$

The state transition is described by a transition matrix $\mathbf{F}(k)$, and the control inputs are mapped into the state space by the matrix $\mathbf{G}(k)$. The observation is described by the observation matrix $\mathbf{H}(k)$. The properties of the process and the observation noise are as defined in Sections 2.2.1 and 2.2.2 respectively.

An estimate at time step $k+1$ may be obtained by taking expected values of Equation 2.5 conditioned on the first k observations,

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{F}(k)\hat{\mathbf{x}}(k|k) + \mathbf{G}(k)\mathbf{u}(k).$$

Since the process model is linear, the prediction of the covariance matrix can be expressed in closed form,

$$\begin{aligned}
\tilde{\mathbf{x}}(k+1|k) &= \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1|k) \\
&= \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \mathbf{v}(k) - [\mathbf{F}(k)\hat{\mathbf{x}}(k|k) + \mathbf{G}(k)\mathbf{u}(k)] \\
&= \mathbf{F}(k)\tilde{\mathbf{x}}(k|k) + \mathbf{v}(k)
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^T(k+1|k)\right] \\
&= \mathrm{E}\left[\mathbf{F}(k)\tilde{\mathbf{x}}(k|k)\tilde{\mathbf{x}}^T(k|k)\mathbf{F}^T(k)\right] + \mathrm{E}\left[\mathbf{v}(k)\mathbf{v}^T(k)\right] \\
&\quad + \mathrm{E}\left[\mathbf{F}(k)\tilde{\mathbf{x}}(k|k)\mathbf{v}^T(k)\right] + \mathrm{E}\left[\mathbf{v}(k)\tilde{\mathbf{x}}^T(k|k)\mathbf{F}^T(k)\right] \\
&= \mathbf{F}(k)\mathbf{P}(k|k)\mathbf{F}(k)^T + \mathbf{Q}(k),
\end{aligned}$$

as $\tilde{\mathbf{x}}(k|k)$ and $\mathbf{v}(k)$ are assumed uncorrelated. The predicted state allows the observation at time

step $k+1$ to be predicted. The innovation vector $\nu(k+1)$ is defined as the difference between the actual observation at time step $k+1$ and the predicted observation,

$$\begin{aligned}
\hat{\mathbf{z}}(k+1|k) &= \mathrm{E}\left[\mathbf{z}(k+1)|\mathbf{Z}^k\right] \\
&= \mathrm{E}\left[\mathbf{H}(k+1)\mathbf{x}(k+1)|\mathbf{Z}^k\right] \\
&= \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k),
\end{aligned}$$

$$\begin{aligned}
\nu(k+1) &\triangleq \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) \\
&= \mathbf{z}(k+1) - \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k).
\end{aligned} \tag{2.7}$$

The update of the estimate is equal to the weighted sum of the observation and the prediction,

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1), \tag{2.8}$$

where $\mathbf{W}(k)$ is the Kalman weighting matrix which determines the influence of the innovation on the updated estimate. The error in the updated estimate is therefore

$$\begin{aligned}
\tilde{\mathbf{x}}(k+1|k+1) &= \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1|k+1) \\
&= \mathbf{x}(k+1) - [\hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1)] \\
&= \tilde{\mathbf{x}}(k+1|k) - \mathbf{W}(k+1)\nu(k+1)
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}(k+1|k+1) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k+1)\tilde{\mathbf{x}}^T(k+1|k+1)\right] \\
&= \mathrm{E}\left[[\tilde{\mathbf{x}}(k+1|k) - \mathbf{W}(k+1)\nu(k+1)]\,[\tilde{\mathbf{x}}(k+1|k) - \mathbf{W}(k+1)\nu(k+1)]^T\right] \\
&= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^T(k+1|k)\right] + \mathrm{E}\left[\mathbf{W}(k+1)\nu(k+1)\nu^T(k+1)\mathbf{W}^T(k+1)\right] \\
&\quad -\mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\nu^T(k+1)\mathbf{W}^T(k+1)\right] - \mathrm{E}\left[\mathbf{W}(k+1)\nu(k+1)\tilde{\mathbf{x}}^T(k+1|k)\right] \\
&= \mathbf{P}(k+1|k) + \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1) \\
&\quad -\mathbf{S}_{\mathbf{x}\nu}(k+1|k)\mathbf{W}^T(k+1) - \mathbf{W}(k+1)\mathbf{S}_{\nu\mathbf{x}}(k+1|k),
\end{aligned} \tag{2.9}$$

and

$$\mathbf{S}_{\nu\nu}(k+1|k) = \mathrm{E}\left[\nu(k+1)\nu^T(k+1)\right] \tag{2.10}$$

$$\begin{aligned}
\mathbf{S}_{\mathbf{x}\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\nu^T(k+1)\right] \\
&= \mathrm{E}\left[\nu(k+1)\tilde{\mathbf{x}}^T(k+1|k)\right]^T \\
&= \mathbf{S}_{\mathbf{x}\nu}^T(k+1|k),
\end{aligned} \tag{2.11}$$

where $\mathbf{S}_{\nu\nu}(k+1|k)$ is the covariance of the innovation, and $\mathbf{S}_{\mathbf{x}\nu}(k+1|k)$ is the covariance of the prediction error and the innovation. The weighting matrix is chosen to minimise the expected mean squared error, $\mathrm{E}\left[\tilde{\mathbf{x}}^T(k+1|k+1)\tilde{\mathbf{x}}(k+1|k+1)\right]$, which is equivalent to the trace of $\mathbf{P}(k+1|k+1)$. The solution for $\mathbf{W}(k+1)$ is [66]

$$\mathbf{W}(k+1) = \mathbf{S}_{\mathbf{x}\nu}(k+1|k)\mathbf{S}_{\nu\nu}^{-1}(k+1|k). \tag{2.12}$$

The Kalman filter computes the optimal update, in the least squares sense, for a state estimate based on the covariance of the state estimate error and the innovation and the covariance of the innovation.

The Kalman update equation for the state estimate and the covariance of the estimate are therefore

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \quad (2.13)$$

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1). \quad (2.14)$$

Equations 2.8–2.14 indicate that the effect of the observation model on the Kalman filter is via the innovation and the covariances of the innovation, rather than the observation itself or the way it is formed.

In the linear case, the update can be expressed in closed form since

$$\begin{aligned}\nu(k+1) &= \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) \\ &= \mathbf{H}(k+1)\mathbf{x}(k+1) + \mathbf{w}(k+1) - \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k) \\ &= \mathbf{H}(k+1)\tilde{\mathbf{x}}(k+1|k) + \mathbf{w}(k+1) \quad (2.15)\end{aligned}$$

$$\begin{aligned}\mathbf{S}_{\nu\nu}(k+1|k) &= \mathrm{E}\left[\nu(k+1)\nu^T(k+1)\right] \\ &= \mathbf{H}(k+1)(k)\mathbf{P}(k+1|k)\mathbf{H}^T(k+1) + \mathbf{R}(k+1)\end{aligned}$$

$$\begin{aligned}\mathbf{S}_{\mathbf{x}\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\nu^T(k+1)\right] \\ &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\left[\mathbf{H}(k+1)\tilde{\mathbf{x}}(k+1|k) + \mathbf{w}(k+1)\right]^T\right] \\ &= \mathbf{P}(k+1|k)\mathbf{H}^T(k+1),\end{aligned}$$

as $\tilde{\mathbf{x}}(k+1|k)$ and $\mathbf{w}(k+1)$ are assumed uncorrelated.

The innovation $\nu(k+1)$ in Equation 2.15 has a special form. It is dependent only on two unknown quantities: the observation noise $\mathbf{w}(k+1)$ and the estimate error $\tilde{\mathbf{x}}(k+1|k)$. These unknown quantities have a special relationship: they are not correlated, $\mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\mathbf{w}^T(k+1)\right] = \mathbf{0}$.

> An observation that leads to an innovation constrained to an estimate error and an observation noise, uncorrelated with each other, shall be referred to as a Constrained Innovation (COIN) observation.

The important result of a COIN observation is that knowledge of only two covariances are needed to form the update: the covariance of the estimate, $\mathbf{P}(k+1|k)$, and the covariance of the observation noise, $\mathbf{R}(k+1)$.

The following box summarises the results for the Kalman filter in the linear case.

The Kalman Filter

The Linear System

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \mathbf{v}(k) \\ \mathbf{z}(k) &= \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k)(k) \end{aligned}$$

The Prediction

$$\begin{aligned} \hat{\mathbf{x}}(k+1|k) &= \mathbf{F}(k)\hat{\mathbf{x}}(k|k) + \mathbf{G}(k)\mathbf{u}(k) \\ \mathbf{P}(k+1|k) &= \mathbf{F}(k)\mathbf{P}(k|k)\mathbf{F}^T(k) + \mathbf{Q}(k) \end{aligned}$$

The Update

$$\begin{aligned} \hat{\mathbf{x}}(k+1|k+1) &= \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \\ \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1) \end{aligned}$$

where

$$\begin{aligned} \nu(k+1) &= \mathbf{z}(k+1) - \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k) \\ \mathbf{S}_{\nu\nu}(k+1|k) &= \mathbf{H}(k+1)\mathbf{P}(k+1|k)\mathbf{H}^T(k+1) + \mathbf{R}(k+1) \\ \mathbf{W}(k+1) &= \mathbf{P}(k+1|k)\mathbf{H}^T(k+1)\mathbf{S}_{\nu\nu}^{-1}(k+1|k) \end{aligned}$$

2.2.5 The Non-Linear Case

In the non-linear case, the system is described by process and observation models in the form

$$\mathbf{x}(k+1) = \mathbf{f}\left[\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k), k\right] \tag{2.16}$$

$$\mathbf{z}(k) = \mathbf{h}\left[\mathbf{x}(k), \mathbf{w}(k), k\right]. \tag{2.17}$$

To estimate the state of a non linear system, a modified form of the Kalman filter, the Extended Kalman Filter (EKF) is widely used [44, 4]. The EKF assumes that the process and observation models are locally linear, that $\mathbf{v}(k)$ and $\mathbf{w}(k)$ are small, and that

$$\hat{\mathbf{x}}(k|k) \approx \mathrm{E}\left[\mathbf{x}(k)|\mathbf{Z}^k\right].$$

The process model is linearised as a Taylor series expansion about $\hat{\mathbf{x}}(k|k)$,

$$\mathbf{x}(k+1) = \mathbf{f}\left[\hat{\mathbf{x}}(k|k), \mathbf{u}(k), \mathbf{0}, k\right] + \nabla\mathbf{f}_{\mathbf{x}}\tilde{\mathbf{x}}(k|k) + \nabla\mathbf{f}_{\mathbf{v}}\mathbf{v}(k) + \text{ higher order terms},$$

where $\nabla \mathbf{f}_{\mathbf{x}}$ is the Jacobian [6] of $\mathbf{f}$ with respect to $\mathbf{x}$ evaluated at $\hat{\mathbf{x}}(k|k)$ and $\nabla \mathbf{f}_{\mathbf{v}}$ is the Jacobian with respect to $\mathbf{v}$. Using the assumptions that the higher order terms in the Taylor series expansion for $\mathbf{x}(k+1)$ are negligible, and that $\tilde{\mathbf{x}}(k|k)$ and $\mathbf{v}(k)$ are small and zero mean random variables, leads to the following prediction equations,

$$\begin{aligned}
\hat{\mathbf{x}}(k+1|k) &= \mathrm{E}\left[\mathbf{x}(k+1)|\mathbf{Z}^k\right] \\
&\approx \mathrm{E}\left[\mathbf{f}\left[\hat{\mathbf{x}}(k|k), \mathbf{u}(k), \mathbf{0}, k\right] + \nabla \mathbf{f}_{\mathbf{x}} \tilde{\mathbf{x}}(k|k) + \nabla \mathbf{f}_{\mathbf{v}} \mathbf{v}(k)\right] \\
&= \mathbf{f}\left[\hat{\mathbf{x}}(k|k), \mathbf{u}(k), \mathbf{0}, k\right]
\end{aligned}$$

$$\begin{aligned}
\tilde{\mathbf{x}}(k+1|k) &= \mathbf{x}(k+1) - \hat{\mathbf{x}}(k+1|k) \\
&\approx \nabla \mathbf{f}_{\mathbf{x}} \tilde{\mathbf{x}}(k|k) + \nabla \mathbf{f}_{\mathbf{v}} \mathbf{v}(k)
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^T(k+1|k)\right] \\
&\approx \nabla \mathbf{f}_{\mathbf{x}} \mathbf{P}(k|k) \nabla \mathbf{f}_{\mathbf{x}}^T + \nabla \mathbf{f}_{\mathbf{v}} \mathbf{Q}(k) \nabla \mathbf{f}_{\mathbf{v}}^T.
\end{aligned}$$

The update equations are formed by linearising the observation model. The observation model is expressed as a Taylor series expansion about $\hat{\mathbf{x}}(k+1|k)$,

$$\mathbf{z}(k+1) = \mathbf{h}\left[\hat{\mathbf{x}}(k+1|k), \mathbf{0}, k\right] + \nabla \mathbf{h}_{\mathbf{x}} \tilde{\mathbf{x}}(k+1|k) + \nabla \mathbf{h}_{\mathbf{w}} \mathbf{w}(k+1) + \text{ higher order terms},$$

where $\nabla \mathbf{h}_{\mathbf{x}}$ is the Jacobian of $\mathbf{h}$ with respect to $\mathbf{x}$ evaluated at $\hat{\mathbf{x}}(k+1|k)$ and $\nabla \mathbf{h}_{\mathbf{w}}$ is the Jacobian with respect to $\mathbf{w}$. Using the assumption

$$\hat{\mathbf{x}}(k+1|k) \approx \mathrm{E}\left[\mathbf{x}(k+1)|\mathbf{Z}^k\right],$$

and the assumptions that $\tilde{\mathbf{x}}(k+1|k)$ and $\mathbf{w}(k+1)$ are small and zero mean random variables, and that the higher order terms in the Taylor series are negligible, leads to the following expressions,

[6] All Jacobians in this thesis are assumed to be evaluated with respect to the state indicated in the subscript and evaluated at the current estimate of that state.

$$\begin{aligned}
\hat{\mathbf{z}}(k+1|k) &\approx \mathrm{E}\left[\mathbf{h}\left[\hat{\mathbf{x}}(k+1|k), \mathbf{0}, k\right] + \nabla\mathbf{h}_{\mathbf{x}}\tilde{\mathbf{x}}(k+1|k) + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{w}(k+1)\right] \\
&= \mathbf{h}\left[\hat{\mathbf{x}}(k+1|k), \mathbf{0}, k\right]
\end{aligned}$$

$$\begin{aligned}
\nu(k+1) &= \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1|k) \\
&\approx \nabla\mathbf{h}_{\mathbf{x}}\tilde{\mathbf{x}}(k+1|k) + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{w}(k+1) \qquad (2.18)
\end{aligned}$$

$$\begin{aligned}
\mathbf{S}_{\nu\nu}(k+1|k) &= \mathrm{E}\left[\nu(k+1)\nu^T(k+1)\right] \\
&\approx \nabla\mathbf{h}_{\mathbf{x}}\mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{R}(k+1)\nabla\mathbf{h}_{\mathbf{w}}^T \qquad (2.19)
\end{aligned}$$

$$\begin{aligned}
\mathbf{S}_{\mathbf{x}\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\nu^T(k+1)\right] \\
&\approx \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\left[\nabla\mathbf{h}_{\mathbf{x}}\tilde{\mathbf{x}}(k+1|k) + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{w}(k+1)\right]^T\right] \\
&= \mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T, \qquad (2.20)
\end{aligned}$$

since $\tilde{\mathbf{x}}(k+1|k)$ and $\mathbf{w}(k+1)$ are again assumed uncorrelated.

Equations 2.8–2.14 still apply in the non-linear case, indicating that the effect of the observation model on the EKF is, as in the linear case, via the innovation and the covariances of the innovation.

The innovation in Equation 2.18 indicates that the non-linear observation model defined in Equation 2.17 leads to COIN observations, as defined in Section 2.2.4. As in the linear case, knowledge of only two covariances is needed to form the update: $\mathbf{P}(k+1|k)$ and $\mathbf{R}(k+1)$.

The following box summarises the results for the EKF, where approximate equality due to neglecting higher order terms is understood but no longer indicated.

The Extended Kalman Filter

The Non-Linear System

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}\left[\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k), k\right] \\ \mathbf{z}(k) &= \mathbf{h}\left[\mathbf{x}(k), \mathbf{w}(k), k\right] \end{aligned}$$

The Prediction

$$\begin{aligned} \hat{\mathbf{x}}(k+1|k) &= \mathbf{f}\left[\hat{\mathbf{x}}(k|k), \mathbf{u}(k), \mathbf{0}, k\right] \\ \mathbf{P}(k+1|k) &= \nabla\mathbf{f}_{\mathbf{x}}\mathbf{P}(k|k)\nabla\mathbf{f}_{\mathbf{x}}^T + \nabla\mathbf{f}_{\mathbf{v}}\mathbf{Q}(k)\nabla\mathbf{f}_{\mathbf{v}}^T \end{aligned}$$

The Update

$$\begin{aligned} \hat{\mathbf{x}}(k+1|k+1) &= \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \\ \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1) \end{aligned}$$

where

$$\begin{aligned} \nu(k+1) &= \mathbf{z}(k+1) - \mathbf{h}\left[\hat{\mathbf{x}}(k+1|k), \mathbf{0}, k\right] \\ \mathbf{S}_{\nu\nu}(k+1|k) &= \nabla\mathbf{h}_{\mathbf{x}}\mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{R}(k+1)\nabla\mathbf{h}_{\mathbf{w}}^T \\ \mathbf{W}(k+1) &= \mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T\mathbf{S}_{\nu\nu}^{-1}(k+1|k) \end{aligned}$$

2.2.6 Consistency of the Kalman Filter

The consistency of the Kalman filter is an important issue in this thesis. The Kalman filter will be said to operate consistently if the estimates it computes are consistent according to the definition given in Section 2.2.3.

For a given predicted state estimate $\hat{\mathbf{x}}(k+1|k)$, initially consistent such that

$$\mathbf{P}(k+1|k) = \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^T(k+1|k)\right],$$

the update

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \tag{2.21}$$

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1), \tag{2.22}$$

is shown to be optimal and consistent in Section 2.2.4 only for the case

$$\mathbf{W}(k+1) = \mathbf{S}_{\mathbf{x}\nu}(k+1|k)\mathbf{S}_{\nu\nu}^{-1}(k+1|k). \tag{2.23}$$

The same update equations are also deduced in Section 2.2.5 for the non-linear case, and again optimality and consistency are shown only for the condition given in Equation 2.23. Although the linearisations used in Section 2.2.5 lead to errors not accounted for in the deduction, these errors are assumed sufficiently small if the non-linear model is differentiable and the linearisation is over a sufficiently small range. [7]

It is possible to form an update $\hat{\mathbf{x}}(k+1|k+1)$ using a $\mathbf{W}(k+1)$ that does not satisfy Equation 2.23, and is therefore sub-optimal. However, the covariance update $\mathbf{P}(k+1|k+1)$ must then satisfy Equation 2.9,

$$\begin{aligned} \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) + \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1) \\ &\quad -\mathbf{S}_{\mathbf{x}\nu}(k+1|k)\mathbf{W}^T(k+1) - \mathbf{W}(k+1)\mathbf{S}_{\nu\mathbf{x}}(k+1|k), \end{aligned} \tag{2.24}$$

to ensure that the covariance update $\mathbf{P}(k+1|k+1)$ satisfies

$$\mathbf{P}(k+1|k+1) = \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k+1)\tilde{\mathbf{x}}^T(k+1|k+1)\right].$$

For any $\mathbf{W}(k+1)$, care must be taken to ensure that the covariance matrix $\mathbf{P}(k+1|k)$ is updated consistently with the update of $\hat{\mathbf{x}}(k+1|k)$.

[7] In this thesis, following the EKF literature, deductions based on linearisations will be assumed to be acceptable.

The Kalman filter and the EKF equations,

$$\begin{aligned}\hat{\mathbf{x}}(k+1|k+1) &= \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \\ \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}_{\nu\nu}(k+1|k)\mathbf{W}^T(k+1),\end{aligned}$$

are guaranteed to produce consistent updates if and only if the condition

$$\mathbf{W}(k+1) = \mathbf{S}_{\mathbf{x}\nu}(k+1|k)\mathbf{S}_{\nu\nu}^{-1}(k+1|k)$$

is satisfied.

2.3 The Mathematical Models

Mathematical models for the localisation problem will now be developed. The localisation problem was defined as the problem of determining the pose of a vehicle travelling within an environment, based upon observations of the environment made by the vehicle. The approach taken in this thesis is to consider the position and orientation, the pose, of the vehicle as the state $\mathbf{x}_v$ and thus form an estimate of the vehicle pose.

The environment is assumed to be modeled by a set of feature states $\mathbf{P}^N = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$, individually observable by the sensor, where $\mathbf{p}_i$ is the state of feature p_i. The features are assumed to be fixed in the global reference frame.

The sensor observes the relative location of the features with respect to the vehicle. Clearly, if the location of the features are known, then the pose of the vehicle may be estimated. More rigorously, in the case of known features, the relative observation of features will be shown in Section 2.4.1 to be equivalent to observations of the vehicle itself, and hence the application of the Kalman filter is straightforward.

The theory that will be developed in this thesis relies on one particular characteristic of the observation model. The observation model must be such that the observations of the feature states are made relative to the vehicle state. The model corresponds to a sensor such as a radar or a laser mounted on the vehicle.

It will be convenient to have specific models as well in order to aid the understanding of the theory. Specific models will be developed for the vehicle, the features, the process and the observation model. The specific models correspond to features, a vehicle and a sensor that were used to test the theory

in experiments described in Chapter 5. However, the theory is not limited to these specific models.

2.3.1 The State Vectors

A simple state vector $\mathbf{x}_v(t)$ of the vehicle at time t is defined, consisting of a Cartesian position $(x_v(t), y_v(t))$ and an orientation $\phi_v(t)$ with respect to the Cartesian reference frame. A simple state vector $\mathbf{p}_i(t)$ of a feature p_i at time t is also defined, as shown in Figure 2.2, consisting of a Cartesian position $(x_i(t), y_i(t))$ in the same reference frame as used for the vehicle,

$$\begin{aligned} \mathbf{x}_v(t) &= \begin{bmatrix} x_v(t) & y_v(t) & \phi_v(t) \end{bmatrix}^T \\ \mathbf{p}_i(t) &= \begin{bmatrix} x_i(t) & y_i(t) \end{bmatrix}^T . \end{aligned}$$

2.3.2 The Vehicle and the Sensor Model

The vehicle kinematics are described by the continuous vehicle kinematic model $\mathbf{f}^c_{\mathbf{x}_v}$. The general vehicle model may depend on time t, but the specific example of $\mathbf{f}^c_{\mathbf{x}_v}$ given does not,

$$\begin{aligned} \dot{\mathbf{x}}_v(t) &= \mathbf{f}^c_{\mathbf{x}_v}\left[\mathbf{x}_v(t), \mathbf{u}(t), \mathbf{v}(t), t\right] \\ \begin{bmatrix} \dot{x}_v(t) \\ \dot{y}_v(t) \\ \dot{\phi}_v(t) \end{bmatrix} &= \begin{bmatrix} [V(t) + v_v(t)]\cos(\phi_v(t) + [\gamma(t) + v_\gamma(t)]) \\ [V(t) + v_v(t)]\sin(\phi_v(t) + [\gamma(t) + v_\gamma(t)]) \\ \frac{[V(t)+v_v(t)]}{B}\sin(\gamma(t) + v_\gamma) \end{bmatrix}, \end{aligned}$$

where B is the base line of the vehicle and $\mathbf{u}(t) = \begin{bmatrix} V(t) & \gamma(t) \end{bmatrix}^T$ is the control input at time t consisting of a velocity input $V(t)$ and a steer input $\gamma(t)$, as shown in Figure 2.2. The process noise $\mathbf{v}(t) = \begin{bmatrix} v_v(t) & v_\gamma(t) \end{bmatrix}^T$ is assumed to be applied to the control inputs, $v_v(t)$ to the velocity input, and $v_\gamma(t)$ to the steer angle input. Assuming that the process noise is applied to the control inputs is convenient in terms of the analysis, gives some guidance on the choice of the process noise covariance, and has proven to be effective in real applications [46].

The vehicle is assumed to be equipped with a sensor that provides a measurement of range $r_i(t)$ and bearing $\theta_i(t)$ to an observed feature p_i relative to the vehicle, as shown in Figure 2.2. As for the vehicle model, the observation model may depend on time t, but the specific example does not. The observation $\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(t)$ of feature p_i at time t can be expressed as

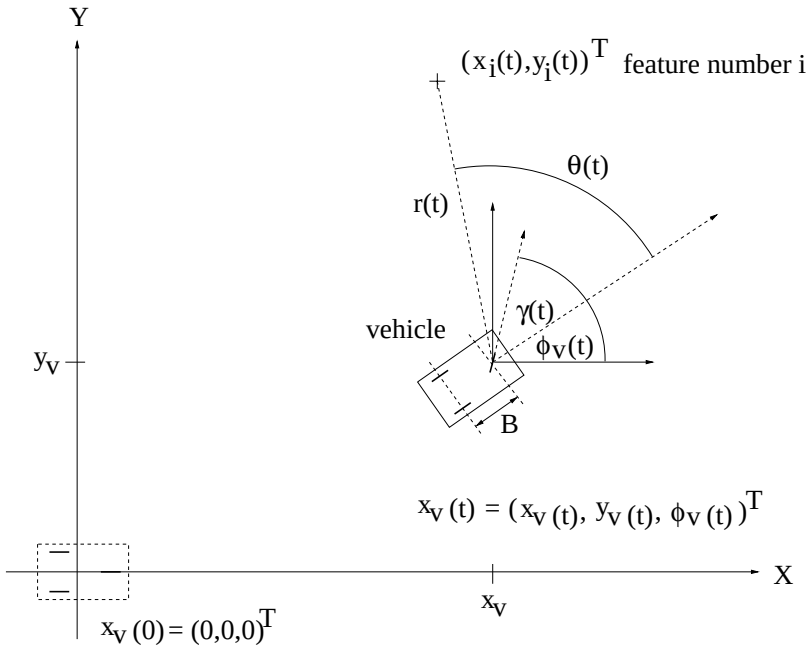

Figure 2.2: The vehicle and a feature.

$$\begin{aligned} \mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(t) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(t), \mathbf{p}_i(t), \mathbf{w}(t), t\right] \\ \begin{bmatrix} r_i(t) \\ \theta_i(t) \end{bmatrix} &= \begin{bmatrix} \sqrt{(x_v(t)-x_i(t))^2 + (y_v(t)-y_i(t))^2} + w_r(t) \\ \tan^{-1}\frac{y_v(t)-y_i(t)}{x_v(t)-x_i(t)} - \phi_v(t) + w_\theta(t) \end{bmatrix}, \end{aligned}$$

where $\mathbf{w}(t) = \begin{bmatrix} w_r(t) & w_\theta(t) \end{bmatrix}^T$ is the observation noise applied to range and bearing observations.

2.3.3 The Discrete-Time Models

The continuous time models formulated must be reformulated in discrete time to allow a computer based implementation. The discrete time representation will be used exclusively in subsequent developments. Although the theory developed in this thesis applies to periods of varying length, a constant period T between the time steps will be used for convenience.

The specific vehicle and feature state vectors at discrete time steps $t_k = kT$ are expressed as

$$\mathbf{x}_v(k) = \begin{bmatrix} x_v(k) & y_v(k) & \phi_v(k) \end{bmatrix}^T \tag{2.25}$$

$$\mathbf{p}_i(k) = \begin{bmatrix} x_i(k) & y_i(k) \end{bmatrix}^T. \tag{2.26}$$

The estimates of the vehicle and the feature states are denoted by $\hat{\mathbf{x}}_v(k|k)$ and $\hat{\mathbf{p}}_i(k|k)$ respectively, and the estimate errors are defined as

$$\begin{aligned}\tilde{\mathbf{x}}_v(k|k) &= \mathbf{x}_v(k) - \hat{\mathbf{x}}_v(k|k)\\ \tilde{\mathbf{p}}_i(k|k) &= \mathbf{p}_i(k) - \hat{\mathbf{p}}_i(k|k).\end{aligned}$$

The discrete time vehicle kinematic model $\mathbf{f}_{\mathbf{x}_v}$ is

$$\begin{aligned}\mathbf{x}_v(k+1) &= \mathbf{f}_{\mathbf{x}_v}\left[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k), k\right]\\ &= \begin{bmatrix} x_v(k) + T\left[V(k) + v_v(k)\right]\cos(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ x_v(k) + T\left[V(k) + v_v(k)\right]\sin(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ \phi_v(k) + \frac{T[V(k)+v_v(k)]}{B}\sin(\gamma(k) + v_\gamma(k)) \end{bmatrix} \\ &= \mathbf{x}_v(k) + T\begin{bmatrix} [V(k) + v_v(k)]\cos(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ [V(k) + v_v(k)]\sin(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ \frac{[V(k)+v_v(k)]}{B}\sin(\gamma(k) + v_\gamma(k)) \end{bmatrix},\end{aligned} \tag{2.27}$$

and the discrete time observation model is

$$\begin{aligned}\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}(k), k\right] &(2.28)\\ &= \begin{bmatrix} r_i(k) + w_r(k) \\ \theta_i(k) + w_\theta(k) \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{(x_i(k) - x_v(k))^2 + (y_i(k) - y_v(k))^2} + w_r(k) \\ \tan^{-1}\frac{y_i(k)-y_v(k)}{x_i(k)-x_v(k)} - \phi_v(k) + w_\theta(k) \end{bmatrix}. &(2.29)\end{aligned}$$

The process model for the features, assumed to be fixed in the global reference frame, is

$$\mathbf{p}_i(k+1) = \mathbf{p}_i(k).$$

Linearisations of these models will be needed in the next section and the following chapters. The discrete-time vehicle model may be linearised using a Taylors series about the vehicle state estimate, $\hat{\mathbf{x}}_v(k|k)$, as follows,

$$\mathbf{x}_v(k+1) \approx \mathbf{f}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k), \mathbf{0}, k\right] + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{v}(k), \tag{2.30}$$

where $\nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}$ and $\nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}}$ are the Jacobians of the vehicle model with respect to the vehicle state $\mathbf{x}_v$ and the process noise $\mathbf{v}$ respectively. In the vehicle model linearisation, it is assumed that the higher order terms in the series are negligible, that

$$\hat{\mathbf{x}}_v(k|k) \approx \mathrm{E}\left[\mathbf{x}_v(k)|\mathbf{Z}^k\right],$$

and that $\tilde{\mathbf{x}}_v(k|k)$ and $\mathbf{v}(k)$ are small. Similarly, the discrete-time observation model may be linearised using a Taylor series about the vehicle state estimate and the observed feature state estimate, $(\hat{\mathbf{x}}_v(k+1|k), \hat{\mathbf{p}}_i(k+1|k))$,

$$\begin{aligned}\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k+1) \approx\ & \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\hat{\mathbf{x}}_v(k+1|k), \hat{\mathbf{p}}_i(k+1|k), \mathbf{w}(k+1), k+1\right] + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k+1|k) \\ & + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \tilde{\mathbf{p}}_i(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1),\end{aligned} \tag{2.31}$$

where $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}$, $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}$ and $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}$ are the Jacobians of the observation with respect to the vehicle state $\mathbf{x}_v$, the feature state $\mathbf{p}_i$ and the observation noise $\mathbf{w}$ respectively. In the observation model linearisation, it is assumed that the higher order terms in the series are again negligible, that

$$\begin{aligned}\hat{\mathbf{x}}_v(k+1|k) &\approx \mathrm{E}\left[\mathbf{x}_v(k+1)|\mathbf{Z}^j\right] \\ \hat{\mathbf{p}}_i(k+1|k) &\approx \mathrm{E}\left[\mathbf{p}_i(k+1)|\mathbf{Z}^j\right],\end{aligned}$$

and that $\tilde{\mathbf{x}}_v(k+1|k)$, $\tilde{\mathbf{p}}_i(k+1|k)$ and $\mathbf{w}(k)$ are small.

2.4 Relative Observations and the Kalman Filter

The localisation problem, as defined in Section 2.1, is the problem of determining the pose of a vehicle traveling within an environment, based upon observations of the environment made relative to the vehicle. In this section, the effects of using relative observations is examined in terms of the Kalman filter. Special cases of relative observations will be considered in Sections 2.4.1 – 2.4.3 to differentiate, in the estimation sense, between localisation, map building, and simultaneous localisation and map

building.

From the linearisation of the observation model, Equation 2.31, the innovation is derived as

$$\begin{aligned}\nu(k+1) &= \mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k+1) - \hat{\mathbf{z}}_{\mathbf{x}_v,\mathbf{p}_i}(k+1|k) \\ &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\tilde{\mathbf{p}}_i(k+1|k) \\ &\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{w}(k+1),\end{aligned} \tag{2.32}$$

which is not constrained to an observation noise and a single estimate error.

The observation of a feature relative to the vehicle is not a COIN [8]observation.

. The variance of the innovation is

$$\begin{aligned}\mathbf{S}_{\nu\nu}(k+1|k) &= \mathrm{E}\left[\nu(k+1)\nu^T(k+1)\right] \\ &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\mathbf{P}_{ii}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} \\ &\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\mathbf{P}_{vi}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\mathbf{P}_{iv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} \\ &\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{w}},\end{aligned} \tag{2.33}$$

where

$$\begin{aligned}\mathbf{P}_{vv}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_v(k+1|k)\tilde{\mathbf{x}}_v^T(k+1|k)\right] \\ \mathbf{P}_{ii}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\tilde{\mathbf{p}}_i^T(k+1|k)\right] \\ \mathbf{P}_{iv}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\tilde{\mathbf{x}}_v^T(k+1|k)\right] \\ &= \mathbf{P}_{vi}^T(k+1|k),\end{aligned}$$

and the assumption was made that the observation noise is not correlated with the error in the vehicle estimate and the error in the feature estimate,

$$\begin{aligned}\mathrm{E}\left[\tilde{\mathbf{x}}_v(k+1|k)\mathbf{w}^T(k+1)\right] &= \mathbf{0} \\ \mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\mathbf{w}^T(k+1)\right] &= \mathbf{0}.\end{aligned}$$

[8]A COIN observation is defined in Section 2.2.4.

The covariance of the innovation and the vehicle state estimate, $\mathbf{S}_{\mathbf{x}_v\nu}(k+1|k)$, and covariance of the innovation and the feature state estimate, $\mathbf{S}_{\mathbf{p}_i\nu}(k+1|k)$, are therefore

$$\begin{aligned}\mathbf{S}_{\mathbf{x}_v\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_v(k+1|k)\nu^T(k+1)\right] \\ &\approx \mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} + \mathbf{P}_{vi}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} \qquad (2.34)\end{aligned}$$

$$\begin{aligned}\mathbf{S}_{\mathbf{p}_i\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\nu^T(k+1)\right] \\ &\approx \mathbf{P}_{ii}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} + \mathbf{P}_{iv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v}. \qquad (2.35)\end{aligned}$$

This leads to different optimal Kalman weighting matrices, according to Equation 2.23, for the vehicle and the feature estimate update, expressed respectively as

$$\mathbf{W}_{\mathbf{x}_v}(k+1) = \mathbf{S}_{\mathbf{x}_v\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k) \qquad (2.36)$$

$$\mathbf{W}_{\mathbf{p}_i}(k+1) = \mathbf{S}_{\mathbf{p}_i\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k). \qquad (2.37)$$

To obtain the optimal Kalman filter update on the vehicle and the observed feature estimate, the covariance between the vehicle and the feature estimate must be known.

For any Kalman weighting matrix, update of the estimate covariance depends on the covariance between the innovation and the state estimate, as shown by Equation 2.24. It follows from Equations 2.34 and 2.35 that the covariance update of the vehicle estimate and the feature estimate both depend on the covariance between the vehicle and the feature estimate.

To formulate a consistent covariance update for the vehicle and the observed feature estimate, the covariance between the vehicle and the feature estimate must be known.

The analysis in this section will be used in Sections 2.4.1 – 2.4.3 to differentiate between three different types of estimation scenarios. The first scenario examines the problem of localising a vehicle in a perfectly known environment, the second scenario examines the problem of building a map of the environment from a vehicle at perfectly known positions, and the third and last scenario examines the problem of simultaneously localising the vehicle and building a map.

2.4.1 Localisation in a Known Environment

Localising in a known environment can be interpreted as follows. In a known environment, the state of the observed feature p_i is known for all i and only the state of the vehicle is unknown. Therefore

state	information
vehicle state $\mathbf{x}_v$	unknown
feature state $\mathbf{p}_i$	known for all i

Table 2.1: Localising in a known environment.

$$
\begin{aligned}
\hat{\mathbf{p}}_i(k+1|\cdot) &= \mathbf{p}_i(k+1) \ \forall \ i \\
\tilde{\mathbf{p}}_i(k+1|\cdot) &= \mathbf{0} \ \forall \ i
\end{aligned}
$$

$$
\begin{aligned}
\hat{\mathbf{x}}_v(k+1|k) &= \mathbf{f}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k), \mathbf{0}, k\right] \\
\mathbf{P}_{vv}(k+1|k) &= \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \mathbf{P}_{vv}(k|k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}^T + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{Q}(k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}}^T
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k+1) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\hat{\mathbf{x}}_v(k+1), \hat{\mathbf{p}}_i(k+1), \mathbf{w}(k+1), k+1\right] + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k+1|k) \\
&\quad + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1) + \text{ higher order terms} \\
\nu(k+1) &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1). \qquad (2.38)
\end{aligned}
$$

The structure of the innovation in Equation 2.38 is the same as what the innovation structure would be if direct observations of the vehicle were made,

$$
\begin{aligned}
\mathbf{z}_{\mathbf{x}_v}(k+1) &= \mathbf{h}_{\mathbf{x}_v}\left[\mathbf{x}_v(k+1), \mathbf{w}(k+1), k+1\right] \qquad (2.39) \\
&\approx \mathbf{h}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k+1|k), \mathbf{0}, k+1\right] + \nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{w}} \mathbf{w}(k+1) \\
\nu(k+1) &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{w}} \mathbf{w}(k+1),
\end{aligned}
$$

where $\mathbf{z}_{\mathbf{x}_v}(k+1)$ is a direct observation of the vehicle according to the observation model $\mathbf{h}_{\mathbf{x}_v}$, and $\nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}$ and $\nabla\left(\mathbf{h}_{\mathbf{x}_v}\right)_{\mathbf{w}}$ are the Jacobians of the observation model with respect to $\mathbf{x}_v$ and $\mathbf{w}$.

Observations of a perfectly known environment made relative to a vehicle are COIN observations of the vehicle, since the innovation $\nu(k+1)$ depends only on the observation noise $\mathbf{w}(k+1)$ and a single estimate error, $\tilde{\mathbf{x}}_v(k+1|k)$. This leads to the following update equations.

Localising in a Known Environment

$$
\begin{aligned}
\mathbf{S}_{\nu\nu}(k+1|k) &= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{w}} \\
\mathbf{S}_{\mathbf{x}_v\nu}(k+1|k) &= \mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} \\
\mathbf{W}_{\mathbf{x}_v}(k+1) &= \mathbf{S}_{\mathbf{x}_v\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k)
\end{aligned}
$$

All the covariance information necessary for an update is contained in $\mathbf{P}_{vv}(k+1|k)$ and $\mathbf{R}(k+1)$. Provided that the observations are associated with the correct features [9], the application of the Kalman filter to the problem of localising a vehicle in a perfectly known environment is conceptually easy, following the above equations. It is also computationally not very demanding since the only state that has to be estimated is the vehicle state $\hat{\mathbf{x}}_v$ and the only covariance matrix that has to be maintained is the vehicle covariance matrix $\mathbf{P}_{vv}$. Although the estimated state and its covariance may be large for sophisticated vehicle models, they are not dependent on the number of features in the environment. Therefore, localising in a known environment is a comparatively easy problem.

2.4.2 Map Building

Map Building in this thesis refers to the problem of constructing a map of the environment when the position of the vehicle is always known perfectly.

vehicle state $\mathbf{x}_v$	known
feature state $\mathbf{p}_i$	unknown for all i

Table 2.2: Map building.

This problem is not usually encountered in practice, but is considered since it is an essential precursor to developments in the next section and subsequent chapters of the thesis. In map building,

[9] The association of an observation with a particular feature is referred to as "gating" [4, 11].

$$\begin{aligned}
\hat{\mathbf{x}}_v(k+1|\cdot) &= \mathbf{x}_v(k+1) \\
\tilde{\mathbf{x}}_v(k+1|\cdot) &= \mathbf{0}
\end{aligned}$$

$$\begin{aligned}
\hat{\mathbf{p}}_i(k+1|k) &= \hat{\mathbf{p}}_i(k|k) \ \forall \ i \\
\mathbf{P}_{ii}(k+1|k) &= \mathbf{P}_{ii}(k|k) \ \forall \ i
\end{aligned}$$

$$\begin{aligned}
\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k+1) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\hat{\mathbf{x}}_v(k+1), \hat{\mathbf{p}}_i(k+1), \mathbf{w}(k+1), k+1\right] + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \tilde{\mathbf{p}}_i(k+1|k) \\
&\quad + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1) + \text{ higher order terms} \\
\nu(k+1) &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \tilde{\mathbf{p}}_i(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1).
\end{aligned}$$

Hence the innovation depends only on the observation error $\mathbf{w}(k+1)$ and the error in the observed feature estimate $\tilde{\mathbf{p}}_i(k+1|k)$. The innovation structure is seen to be the innovation structure of a direct observation of feature p_i,

$$\begin{aligned}
\mathbf{z}_{\mathbf{p}_i}(k+1) &= \mathbf{h}_{\mathbf{p}}\left[\mathbf{p}_i(k+1), \mathbf{w}(k+1), k+1\right] \\
&\approx \mathbf{h}_{\mathbf{p}}\left[\hat{\mathbf{p}}_i(k+1|k), \mathbf{0}, k+1\right] + \nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{p}_i} \tilde{\mathbf{p}}_i(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1) \\
\nu(k+1) &\approx \nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{p}_i} \tilde{\mathbf{p}}_i(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{w}} \mathbf{w}(k+1),
\end{aligned} \tag{2.40}$$

where $\mathbf{z}_{\mathbf{p}_i}(k+1)$ is a direct observation of feature p_i according to the observation model $\mathbf{h}_{\mathbf{p}}$, and $\nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{p}_i}$ and $\nabla\left(\mathbf{h}_{\mathbf{p}}\right)_{\mathbf{w}}$ are the Jacobians of the observation model with respect to $\mathbf{p}_i(k+1)$ and $\mathbf{w}(k+1)$.

An observation of feature p_i made relative to a vehicle of perfectly known state is a COIN observation of feature p_i, leading to the following update equations.

Map Building

$$\begin{aligned}
\mathbf{S}_{\nu\nu}(k+1|k) &= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \mathbf{P}_{ii}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{R}(k+1) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}^T \\
\mathbf{S}_{\mathbf{p}_i\nu}(k+1|k) &= \mathbf{P}_{ii}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \\
\mathbf{W}_{\mathbf{p}_i}(k+1) &= \mathbf{S}_{\mathbf{p}_i\nu}(k+1|k) \mathbf{S}_{\nu\nu}^{-1}(k+1|k)
\end{aligned}$$

All the covariance information necessary for an update is contained in $\mathbf{P}_{ii}(k+1|k)$ and $\mathbf{R}(k+1)$.

Provided that the observation is associated with the correct feature, the application of the Kalman filter to the problem building a map from a vehicle of perfectly known pose is conceptually easy. Each observation is associated with a single feature, and is a COIN observation of the state of just that feature. Using the observations of a particular feature, a Kalman filter can be used to estimate the state of that feature. Since each feature estimate is computed independently of the estimate of other feature states, N independent Kalman Filters can be used to estimate an environment of N features: each filter assigned to a feature, and each processing only the observations of the particular feature it has been assigned to.

With every observation only one estimate, the estimate of the observed feature, is updated and consequently only one Kalman filter operates at any one time. Therefore, map building is comparable in terms of difficulty to localising in a known environment. There is an increase in storage requirement with the number of features in the map, but the increase is linear, and there is no increase in the computation time.

2.4.3 The Simultaneous Localisation And Map Building Problem

Simultaneously localising the position of the vehicle and building a map is significantly more complex than either localising the pose of the vehicle in a known environment or building a map when the position of the vehicle is always known perfectly. Neither the vehicle state nor the feature states are known with certainty.

vehicle state $\mathbf{x}_v$	unknown
feature state $\mathbf{p}_i$	unknown for all i

Table 2.3: Simultaneous localisation and map building.

The complexity of the problem arises from the fact that two states, the vehicle state and the observed feature state need to be estimated, from a single relative observation. Comparing the problem to an algebraic problem of one equation, given by the relative observation, and two unknowns, representing the vehicle and the feature states, suggests that the solution cannot be determined. To some extent, as will be shown in Section 3.7, this conclusion is true. However, it is not unreasonable to assume that the initial pose of the vehicle will be known to some degree. This assumption can also be justified by arguing that the initial pose may be *defined* to be a known position. The solution of the Simultaneous Localisation and Map Building (SLAM) problem can then be conceptionalised as follows. The vehicle's initial pose is assumed to be known with some certainty. This is defined as the initial vehicle state estimate. Upon the relative observation of a feature, the position of the observed feature may be estimated. This leads to the initialisation of the first feature state estimate. Reobserving the initialised feature allows the vehicle state estimate to be updated, and hence the

vehicle to be localised.

Operating within the view of only a single feature is clearly not a general solution to the SLAM problem. It is necessary for the vehicle to initialise feature estimates in all areas of the operational environment, so that the vehicle can be localised anywhere within the environment. Therefore, the solution of the SLAM problem requires that several feature states be estimated. In terms of estimation, the SLAM problem can be stated as the problem of simultaneously estimating the vehicle state and several feature states. The true complexity of this problem arises from the fact the observation depends on the observed feature state and the vehicle state,

$$\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k) = \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}(k), k\right], \tag{2.41}$$

and as a result, the innovation depends on the error in the vehicle state estimate and the error in the feature state estimate, as well as the observation noise,

$$\begin{aligned}\nu(k+1) &= \mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k+1) - \hat{\mathbf{z}}_{\mathbf{x}_v,\mathbf{p}_i}(k+1|k) \\ &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\tilde{\mathbf{p}}_i(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{w}(k+1).\end{aligned}$$

Consequently, the covariance of the innovation is

$$\begin{aligned}\mathbf{S}_{\nu\nu}(k+1|k) &= \mathrm{E}\left[\nu(k+1)\nu^T(k+1)\right] \\ &\approx \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\mathbf{P}_{ii}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} \\ &\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\mathbf{P}_{vi}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i} + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\mathbf{P}_{iv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} \\ &\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{R}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{w}},\end{aligned}$$

showing that the covariance $\mathbf{S}_{\nu\nu}(k+1|k)$ depends on the covariance $\mathbf{P}_{vv}(k+1|k)$ of the vehicle estimate, the covariance $\mathbf{P}_{ii}(k+1|k)$ of the observed feature estimate, and the covariance $\mathbf{P}_{vi}(k+1|k)$ between the vehicle estimate and the observed feature estimate. Knowledge of $\mathbf{P}_{vi}(k+1|k)$ is also shown by Equations 2.34 and 2.35 to be necessary to formulate a consistent covariance update of the vehicle and the observed feature estimate.

However, if $\hat{\mathbf{x}}_v$ is updated, $\mathbf{P}_{vi}$ must also be updated in preparation for a future observation of p_i, since $\mathbf{P}_{vi} = \mathrm{E}\left[\tilde{\mathbf{x}}_v\tilde{\mathbf{x}}_v^T\right]$ and $\tilde{\mathbf{x}}_v = \mathbf{x}_v - \hat{\mathbf{x}}_v$ will change after an update of $\hat{\mathbf{x}}_v$. Further, $\mathbf{P}_{vj}$ must also be updated in preparation for the observation of a different feature p_j. Consequently, the application

of the Kalman filter to the SLAM problem is not trivial.

The SLAM problem is at the core of this thesis. A detailed examination of how the Kalman filter can be applied to the problem is presented in Chapter 3.

2.5 Summary

This chapter has formulated the localisation problem as the problem of determining the vehicle vehicle pose based on observations of the environment made by a sensor mounted on the vehicle.

Estimation has been identified as the particular approach taken to solve the localisation problem. The vehicle pose is defined as a state that needs to be estimated, and the Kalman filter has been introduced as the algorithm used to form the estimate. The environment is assumed to consist of features that can be individually observed by the sensor, and the position of a feature is defined as the state of that feature.

Three particular cases of interest were identified. The first case examined the problem of localising a vehicle in a known environment. In terms of estimation, this was interpreted as the problem of estimating the state of the vehicle when the state of all features is known perfectly. It was shown that the application of the Kalman filter to this problem is straightforward.

The second case examined the problem of building a map of the environment when the pose of the vehicle is always known perfectly. In terms of estimation, this was interpreted as the problem of estimating the state of every feature in the environment when the state of the vehicle is always known perfectly. It was shown that the application of the Kalman filter to this problem is also straightforward.

Finally, the third case examined the problem of building a map of the environment and simultaneously using the map to localise the vehicle. This is referred to as the Simultaneous Localisation and Map Building problem, or the SLAM problem. In terms of estimation, this was interpreted as the problem of simultaneously estimating the state of the vehicle and the state of every feature in the environment. The SLAM problem was identified as a complex estimation problem, because neither the state of the vehicle nor the state of the features are known perfectly, yet both need to be estimated based on observations of the feature made relative to the vehicle. The investigation of and the solution to the SLAM problem is the main focus of this thesis.

Chapter 3

Simultaneous Localisation And Map Building

3.1 Introduction

This chapter examines the application of the Kalman filter to the problem of simultaneously localising the position of a vehicle and building a map of the environment. A solution to this problem would allow a vehicle to start at an unknown location in an unknown environment and then to build a map in which to localise itself; in principle to provide a truly autonomous navigational behaviour.

A solution, referred to as the Map Augmented Kalman (MAK) filter, to the Simultaneous Localisation and Map Building (SLAM) problem is introduced in Section 3.2. The MAK filter maintains in the covariance matrix a measure of all the correlations that arise between the errors in the feature estimates. In Section 3.3 the significance of these correlations is examined, and it is shown that all covariance terms maintained by the MAK filter are necessary to produce a consistent map. If the correlations are ignored, the map produced may be inconsistent. This is a key result in the analysis of the SLAM problem. Section 3.3 further shows that the MAK filter computes the optimal update for each feature estimate. Section 3.4 examines the evolution of correlations between the estimate errors and demonstrates that none of the correlations may be assumed to be zero. Section 3.5 shows that a map constructed by the MAK filter will be non-divergent. Finally, the limit of the map accuracy is investigated in Section 3.7. Key results are summarised in Section 3.8.

3.2 The Augmented State

To completely model the SLAM problem the vehicle state vector must be augmented with the state vectors of all the mapped features. This is necessary to keep track of all correlations between the vehicle and feature estimate errors. These correlations are essential to the solution of the SLAM Problem, as will be shown in Section 3.3.

At time k, the state of the vehicle is denoted by $\mathbf{x}_v(k)$ and the state of feature p_i is denoted by $\mathbf{p}_i(k)$. The map augmented vehicle state, $\mathbf{x}_{mav}(k)$, describing the state of the vehicle and all features in the environment at time k is defined as

$$\mathbf{x}_{mav}(k) \triangleq \begin{bmatrix} \mathbf{x}_v^T(k) & \mathbf{p}_1^T(k) & \dots & \mathbf{p}_N^T(k) \end{bmatrix}^T . \tag{3.1}$$

The estimate $\hat{\mathbf{x}}_{mav}(k|k)$ of the augmented state and its corresponding covariance $\mathbf{P}_{mav}(k|k)$ are

$$\begin{aligned}
\hat{\mathbf{x}}_{mav}(k|k) &= \begin{bmatrix} \hat{\mathbf{x}}_v^T(k|k) & \hat{\mathbf{p}}_1^T(k|k) & \dots & \hat{\mathbf{p}}_N^T(k|k) \end{bmatrix}^T \\
\mathbf{P}_{mav}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_{mav}(k|k)\tilde{\mathbf{x}}_{mav}^T(k|k)\right] \\
&= \mathrm{E}\left[\begin{bmatrix} \tilde{\mathbf{x}}_v(k|k) \\ \tilde{\mathbf{p}}_1(k|k) \\ \vdots \\ \tilde{\mathbf{p}}_N(k|k) \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_v(k|k) \\ \tilde{\mathbf{p}}_1(k|k) \\ \vdots \\ \tilde{\mathbf{p}}_N^T \end{bmatrix}^T\right] \\
&= \begin{bmatrix} \mathbf{P}_{vv}(k|k) & \mathbf{P}_{v1}(k|k) & \dots & \mathbf{P}_{vN}(k|k) \\ \mathbf{P}_{1v}(k|k) & \mathbf{P}_{11}(k|k) & \dots & \mathbf{P}_{1N}(k|k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv}(k|k) & \mathbf{P}_{N1}(k|k) & \dots & \mathbf{P}_{NN}(k|k) \end{bmatrix},
\end{aligned} \tag{3.2}$$

where

$$\begin{aligned}
\mathbf{P}_{vv}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_v(k|k)\tilde{\mathbf{x}}_v^T(k|k)\right] \\
\mathbf{P}_{ii}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_i(k|k)\tilde{\mathbf{p}}_i^T(k|k)\right] \\
\mathbf{P}_{vi}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_v(k|k)\tilde{\mathbf{p}}_i^T(k|k)\right] \\
\mathbf{P}_{ij}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_i(k|k)\tilde{\mathbf{p}}_j^T(k|k)\right] .
\end{aligned}$$

The terms $\mathbf{P}_{vv}(k|k)$ and $\mathbf{P}_{ii}(k|k)$ for $i = 1, \dots, N$ are the covariance matrices for the vehicle and the

N feature estimates respectively. The terms $\mathbf{P}_{vi}(k|k) = \mathbf{P}_{iv}^T(k|k)$ measure the correlation between the error in the vehicle and the feature estimate, for each feature p_i, and the terms $\mathbf{P}_{ij}(k|k) = \mathbf{P}_{ji}^T(k|k)$ measure the correlation between the errors in the estimates of two features p_i and p_j. The significance of these terms is investigated in Section 3.3.

The map augmented vehicle state structure allows the Kalman filter to maintain a measure of all correlations that arise between the errors in its estimates.

The map augmented vehicle model $\mathbf{f}_{\mathbf{x}_{mav}}$ is derived from the vehicle model $\mathbf{f}_{\mathbf{x}_v}$, Equation 2.30, and from the assumption that the features are fixed in the global reference frame,

$$\begin{aligned}
\mathbf{x}_v(k+1) &= \mathbf{f}_{\mathbf{x}_v}\left[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k), k\right] \\
&\approx \mathbf{f}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k), \mathbf{0}, k\right] + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{v}(k) \\
\mathbf{p}_i(k+1) &= \mathbf{p}_i(k) \quad i = 1, 2, \ldots, N
\end{aligned}$$

$$\begin{aligned}
\mathbf{x}_{mav}(k+1) &= \mathbf{f}_{\mathbf{x}_{mav}}\left[\mathbf{x}_{mav}(k), \mathbf{u}(k), \mathbf{v}(k), k\right] \\
&= \begin{bmatrix} \mathbf{x}_v^T(k+1) & \mathbf{p}_1^T(k+1) & \ldots & \mathbf{p}_N^T(k+1) \end{bmatrix}^T \\
&= \begin{bmatrix} \mathbf{f}_{\mathbf{x}_v}\left[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k), k\right] \\ \mathbf{p}_1(k) \\ \vdots \\ \mathbf{p}_N(k) \end{bmatrix} \\
&\approx \begin{bmatrix} \mathbf{f}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k), \mathbf{0}, k\right] + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{v}(k) \\ \mathbf{p}_1(k) \\ \vdots \\ \mathbf{p}_N(k) \end{bmatrix}.
\end{aligned} \tag{3.3}$$

The prediction of the map augmented vehicle state estimate $\hat{\mathbf{x}}_{mav}(k+1|k)$ is derived as follows,

$$\begin{aligned}
\hat{\mathbf{x}}_{mav}(k+1|k) &= \mathrm{E}\left[\mathbf{x}_{mav}(k+1)|\mathbf{Z}^k\right] \\
&\approx \begin{bmatrix} \mathbf{f}_{\mathbf{x}_v}\left[\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k), \mathbf{0}, k\right] \\ \hat{\mathbf{p}}_1(k|k) \\ \vdots \\ \hat{\mathbf{p}}_N(k|k) \end{bmatrix} \\
&= \mathbf{f}_{\mathbf{x}_{mav}}\left[\hat{\mathbf{x}}_{mav}(k|k), \mathbf{u}(k), \mathbf{0}, k\right]
\end{aligned} \tag{3.4}$$

$$\begin{aligned}
\tilde{\mathbf{x}}_{mav}(k+1|k) &= \mathbf{x}_{mav}(k+1) - \hat{\mathbf{x}}_{mav}(k+1|k) \\
&= \begin{bmatrix} \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \tilde{\mathbf{x}}_v(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{v}(k) \\ \tilde{\mathbf{p}}_1(k) \\ \vdots \\ \tilde{\mathbf{p}}_N(k) \end{bmatrix},
\end{aligned}$$

from which it follows that

$$\begin{aligned}
\mathbf{P}_{mav}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}_{mav}(k+1|k)\tilde{\mathbf{x}}_{mav}^T(k+1|k)\right] \\
&= \begin{bmatrix} \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \mathbf{P}_{vv}(k|k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}^T & \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \mathbf{P}_{v1}(k|k) & \dots & \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} \mathbf{P}_{vN}(k|k) \\ \mathbf{P}_{1v}(k|k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}^T & \mathbf{P}_{11}(k|k) & \dots & \mathbf{P}_{1N}(k|k) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv}(k|k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v}^T & \mathbf{P}_{N1}(k|k) & \dots & \mathbf{P}_{NN}(k|k) \end{bmatrix} \\
&\quad + \begin{bmatrix} \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}} \mathbf{Q}(k) \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}}^T & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \\
&= \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{x}_{mav}} \mathbf{P}_{mav}(k|k) \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{x}_{mav}}^T + \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}} \mathbf{Q}(k) \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}},
\end{aligned} \tag{3.5}$$

where

$$\begin{aligned}
\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{x}_{mav}} &= \begin{bmatrix} \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{x}_v} & \mathbf{0} & \mathbf{0} & \dots \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \\
\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}} &= \begin{bmatrix} \nabla\left(\mathbf{f}_{\mathbf{x}_v}\right)_{\mathbf{v}}^T & \mathbf{0} & \dots \end{bmatrix}^T .
\end{aligned}$$

Therefore, a consistent prediction stage can be derived for the map augmented vehicle state.

The observation model must also account for the state vector augmentation, and hence a new model is defined. However, the new observation model only represents a change of notation. The new observation model $\mathbf{h}_{\mathbf{x}_{mav},i}$ must describe the observation of a single feature as described by $\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}$, but must be expressed in terms of the map augmented vehicle state $\mathbf{x}_{mav}$. It follows that

$$\begin{aligned}
\mathbf{z}_{\mathbf{x}_{mav},i}(k) &= \mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k) \\
\mathbf{h}_{\mathbf{x}_{mav},i}\left[\mathbf{x}_{mav}(k), i, \mathbf{w}(k), k\right] &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}(k), k\right], \qquad (3.6)
\end{aligned}$$

and also

$$\begin{aligned}
\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_v} &= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \\
\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_i} &= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \qquad (3.7) \\
\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_j} &= \mathbf{0} \text{ ; for } j \neq i \\
\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} &= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} .
\end{aligned}$$

Using the identities in Equations 3.7, the Jacobian of the augmented observation model with respect to the feature augmented vehicle state is

$$\begin{aligned}
\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_{mav}} &= \begin{bmatrix} \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_v} & \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_1} & \dots & \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_N} \end{bmatrix} \\
&= \begin{bmatrix} \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} & \mathbf{0} & \dots & \mathbf{0} & \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} & \mathbf{0} & \dots \end{bmatrix}. \qquad (3.8)
\end{aligned}$$

A simplification of the notation makes the subsequent analysis easier to follow,

$$\begin{aligned}
\mathbf{H}_i &\triangleq \nabla \left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_{mav}} \\
\mathbf{H}_v &\triangleq \nabla \left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_v} \\
\mathbf{H}_p &\triangleq \nabla \left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{p}_i} \\
\mathbf{H}_w &\triangleq \nabla \left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} .
\end{aligned}$$

For example, the Jacobians $\nabla \left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{x}_{mav}}$ of the observations of features p_1, p_2 and p_3 are, respectively, [1]

$$\begin{aligned}
\mathbf{H}_1(k) &= \begin{bmatrix} \mathbf{H}_v(k) & \mathbf{H}_p(k) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots \end{bmatrix} \\
\mathbf{H}_2(k) &= \begin{bmatrix} \mathbf{H}_v(k) & \mathbf{0} & \mathbf{H}_p(k) & \mathbf{0} & \mathbf{0} & \dots \end{bmatrix} \\
\mathbf{H}_3(k) &= \begin{bmatrix} \mathbf{H}_v(k) & \mathbf{0} & \mathbf{0} & \mathbf{H}_p(k) & \mathbf{0} & \dots \end{bmatrix} .
\end{aligned} \tag{3.9}$$

The particular Kalman filter solution to the SLAM problem developed in this section is referred to as the Map Augmented Kalman (MAK) filter.

3.3 Significance of the Correlations

This section shows that maintenance of all the terms in the covariance matrix in Equation 3.2 is essential to the solution of the SLAM problem.

The augmented state allows the MAK filter to explicitly maintain correlations between errors in vehicle and feature, and feature and feature estimates. The significance of these correlations can be examined by considering an update obtained from an observation of feature p_i. The feature augmented vehicle state, the covariance matrix and the observation model are given below, with the time indices re-occurring inside an expression written as outer time indices to improve clarity,

[1] If the observation model is linear, the structure of $\mathbf{H}_i(k)$ is still preserved. However, the terms $\mathbf{H}_v(k)$ and $\mathbf{H}_p(k)$ are given by the observation model directly.

$$
\begin{aligned}
\hat{\mathbf{x}}_{mav}(k+1|k) &= \begin{bmatrix} \hat{\mathbf{x}}_v^T & \dots & \hat{\mathbf{p}}_i^T & \dots & \hat{\mathbf{p}}_j^T & \dots & \hat{\mathbf{p}}_N^T \end{bmatrix}^T (k|k) \\
\mathbf{P}_{mav}(k|k) &= \begin{bmatrix} \mathbf{P}_{vv} & \dots & \mathbf{P}_{vi} & \dots & \mathbf{P}_{vj} & \dots & \mathbf{P}_{vN} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{P}_{iv} & \dots & \mathbf{P}_{ii} & \dots & \mathbf{P}_{ij} & \dots & \mathbf{P}_{iN} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{P}_{jv} & \dots & \mathbf{P}_{ji} & \dots & \mathbf{P}_{jj} & \dots & \mathbf{P}_{jN} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv} & \dots & \mathbf{P}_{Ni} & \dots & \mathbf{P}_{Nj} & \dots & \mathbf{P}_{NN} \end{bmatrix} (k|k) \\
\mathbf{H}_i(k+1) &= \begin{bmatrix} \mathbf{H}_v & \mathbf{0} & \dots & \mathbf{0} & \mathbf{H}_p & \mathbf{0} & \dots \end{bmatrix} (k+1).
\end{aligned}
$$

From Equation 2.19, the variance of the innovation is

$$
\begin{aligned}
\mathbf{S}_i(k+1|k) &= \mathbf{H}_i(k+1)\mathbf{P}_{mav}(k+1|k)\mathbf{H}_i^T(k+1) + \mathbf{H}_w(k+1)\mathbf{R}(k+1)\mathbf{H}_w^T(k+1) \\
&= \mathbf{H}_v(k+1)\mathbf{P}_{vv}(k+1|k)\mathbf{H}_v^T(k+1) + \mathbf{H}_p(k+1)\mathbf{P}_{ii}(k+1|k)\mathbf{H}_p^T(k+1) \\
&\quad + \mathbf{H}_v(k+1)\mathbf{P}_{vi}(k+1|k)\mathbf{H}_p^T(k+1) + \mathbf{H}_p(k+1)\mathbf{P}_{iv}(k+1|k)\mathbf{H}_v^T(k+1) \\
&\quad + \mathbf{H}_w(k+1)\mathbf{R}(k+1)\mathbf{H}_w^T(k+1) \\
&= \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \mathbf{P}_{vv}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \mathbf{P}_{ii}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \\
&\quad + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} \mathbf{P}_{vi}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} \mathbf{P}_{iv}(k+1|k) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T \\
&\quad + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}} \mathbf{R}(k+1) \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}^T,
\end{aligned}
$$

which is recognised to be in the form of $\mathbf{S}_{\nu\nu}(k+1|k)$ deduced in Equation 2.33. Therefore, the MAK filter solution to the SLAM problem correctly computes the covariance of the innovation. Since any of the features p_i, $i = 1, 2, \dots, N$, may be observed, all the covariance terms $\mathbf{P}_{vi}(k+1|k)$, $i = 1, 2, \dots, N$, must be known so that the covariance of the innovation can always be computed. A covariance matrix that maintains the terms $\mathbf{P}_{vi}(k+1|k)$, $i = 1, 2, \dots, N$, must also maintain the terms $\mathbf{P}_{ij}(k+1|k)$, $i = 1, 2, \dots, N$ and $j = 1, 2, \dots, N$, to complete the covariance matrix, otherwise the condition $\mathbf{P}(k+1|k) = \mathrm{E}\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^T(k+1|k)\right]$, given in Equation 2.4 and assumed by the Kalman filter, is not satisfied.

The MAK filter ensures that the covariance of the innovation is computed correctly by maintaining the covariance between the vehicle and all feature estimates and also maintaining the covariance between all pairs of feature estimates.

The Kalman weighting matrix is examined next. In the MAK filter formulation, the Kalman weighting matrix is expressed as

$$\begin{aligned}
\mathbf{W}_i(k+1) &= \mathbf{P}_{mav}(k+1|k)\mathbf{H}_i^T(k+1)\mathbf{S}_i^{-1}(k+1|k) \\
&= \begin{bmatrix} \mathbf{P}_{vv}\mathbf{H}_v^T + \mathbf{P}_{vi}\mathbf{H}_p^T \\ \vdots \\ \mathbf{P}_{iv}\mathbf{H}_v^T + \mathbf{P}_{ii}\mathbf{H}_p^T \\ \vdots \\ \mathbf{P}_{jv}\mathbf{H}_v^T + \mathbf{P}_{ji}\mathbf{H}_p^T \\ \vdots \\ \mathbf{P}_{Nv}\mathbf{H}_v^T + \mathbf{P}_{Ni} \end{bmatrix} (k+1|k)\mathbf{S}_i^{-1}(k+1|k) \\
&= \begin{bmatrix} \mathbf{P}_{vv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T + \mathbf{P}_{vi}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \\ \vdots \\ \mathbf{P}_{iv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T + \mathbf{P}_{ii}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \\ \vdots \\ \mathbf{P}_{jv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T + \mathbf{P}_{ji}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \\ \vdots \\ \mathbf{P}_{Nv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}^T + \mathbf{P}_{Ni}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}^T \end{bmatrix} \mathbf{S}_{\nu\nu}^{-1}(k+1|k).
\end{aligned}$$

From the expression for the innovation given in Equation 2.32 and using the assumption

$$\mathrm{E}\left[\tilde{\mathbf{p}}_j(k+1|k)\mathbf{w}^T(k+1)\right] = \mathbf{0},$$

it follows that

$$\begin{aligned}
\mathbf{S}_{\mathbf{p}_j\nu}(k+1|k) &= \mathrm{E}\left[\tilde{\mathbf{p}}_j(k+1|k)\nu^T(k+1)\right] \\
&= \mathrm{E}[\tilde{\mathbf{p}}_j(k+1|k)[\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}\tilde{\mathbf{x}}_v(k+1|k) + \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}\tilde{\mathbf{p}}_i(k+1|k) \\
&\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{w}}\mathbf{w}(k+1)]] \\
&= \mathbf{P}_{jv}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{x}_v} + \mathbf{P}_{ji}(k+1|k)\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)^T_{\mathbf{p}_i}.
\end{aligned}$$

Hence, combining with Equations 2.34 and 2.35, $\mathbf{W}_i(k+1)$ is

$$\mathbf{W}_i(k+1) = \begin{bmatrix} \mathbf{S}_{\mathbf{x}_v\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k) \\ \vdots \\ \mathbf{S}_{\mathbf{p}_i\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k) \\ \vdots \\ \mathbf{S}_{\mathbf{p}_j\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k) \\ \vdots \\ \mathbf{S}_{\mathbf{p}_N\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k) \end{bmatrix}. \tag{3.10}$$

The submatrices $\mathbf{S}_{\mathbf{x}_v\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k)$ and $\mathbf{S}_{\mathbf{p}_i\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k)$ of $\mathbf{W}_i(k+1)$ are recognised as the Kalman weighting matrices for the update of the vehicle and the observed feature estimate, given by Equations 2.36 and 2.37 respectively, and showing the optimality of the MAK filter.

The submatrix $\mathbf{S}_{\mathbf{p}_j\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k)$ of $\mathbf{W}_i(k+1)$ illustrates that the MAK filter can also update the estimate of unobserved features, due to the correlation between the innovation and error in the estimate of the unobserved feature p_j , represented by the term $\mathbf{S}_{\mathbf{p}_j\nu}(k+1|k)$. The submatrix $\mathbf{S}_{\mathbf{p}_j\nu}(k+1|k)\mathbf{S}^{-1}_{\nu\nu}(k+1|k)$, consisting of the product of $\mathbf{S}_{\mathbf{p}_j\nu}(k+1|k)$ (the covariance of the estimate and the innovation) and the inverse of $\mathbf{S}_{\nu\nu}(k+1|k)$ (the covariance of the innovation) is in the form of the Kalman weighting matrix in Equation 2.24, further confirming the optimality of the MAK filter.

From Equation 3.10,

$$\begin{aligned}
\mathbf{W}_i(k+1) &= \begin{bmatrix} \mathrm{E}\left[\tilde{\mathbf{x}}_v(k+1|k)\nu^T(k+1)\right] \\ \vdots \\ \mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\nu^T(k+1)\right] \\ \vdots \\ \mathrm{E}\left[\tilde{\mathbf{p}}_j(k+1|k)\nu^T(k+1)\right] \\ \vdots \\ \mathrm{E}\left[\tilde{\mathbf{p}}_N(k+1|k)\nu^T(k+1)\right] \end{bmatrix} \mathbf{S}_{\nu\nu}^{-1}(k+1|k) \\
&= \mathrm{E}\left[\tilde{\mathbf{x}}_{mav}(k+1|k)\nu^T(k+1)\right]\mathbf{S}_{\nu\nu}^{-1}(k+1|k) \\
&= \mathbf{S}_{\mathbf{x}_{mav}\nu}(k+1|k)\mathbf{S}_{\nu\nu}^{-1}(k+1|k). \qquad (3.11)
\end{aligned}$$

Therefore, $\mathbf{W}_i(k+1)$ satisfies the condition, identified in Section 2.2.6, for a consistent update of $\hat{\mathbf{x}}_{mav}(k+1|k)$ and $\mathbf{P}_{mav}(k+1|k)$ according to Equations 3.12 and 3.13,

$$\hat{\mathbf{x}}_{mav}(k+1|k+1) = \hat{\mathbf{x}}_{mav}(k+1|k) + \mathbf{W}_i(k+1)\nu(k+1) \qquad (3.12)$$

$$\mathbf{P}_{mav}(k+1|k+1) = \mathbf{P}_{mav}(k+1|k) - \mathbf{W}_i(k+1)\mathbf{S}_i^T(k+1|k)\mathbf{W}_i^T(k+1). \qquad (3.13)$$

The results in this section may be interpreted as follows. In the derivation of Equation 2.23, the weighting matrix is chosen to minimise the expected mean squared error. This leads to a consistent and optimal, in the least squares sense, update of the state estimate $\hat{\mathbf{x}}(k+1|k)$, according to Equations 2.21 and 2.22. However, the state estimate $\hat{\mathbf{x}}_{mav}(k+1|k)$ in the MAK filter includes all the feature state estimates $\hat{\mathbf{p}}_i(k+1|k)$, $i = 1, 2, \ldots, N$, and the covariance matrix $\mathbf{P}_{mav}(k+1|k)$ is assumed to satisfy $\mathbf{P}_{mav}(k+1|k) = \mathrm{E}\left[\tilde{\mathbf{x}}_{mav}(k+1|k)\tilde{\mathbf{x}}_{mav}^T(k+1|k)\right]$. Therefore, a consistent and optimal, in the least squares sense, update of the vehicle and each feature estimate is computed by the Kalman filter according to Equations 3.12 and 3.13 if $\mathbf{W}_i(k+1)$ satisfies Equation 3.11.

- The MAK filter ensures that the vehicle estimate and the feature estimates are updated consistently.
- The MAK filter leads to the optimal update of the vehicle and all feature estimates with each observation.

3.4 Source of the Correlations

This section examines how correlations arise between the estimate errors in the MAK filter. It complements Section 3.3 by showing that the terms in $\mathbf{P}_{mav}$, identified in Section 3.3 as essential to the solution of the SLAM problem, must be assumed to be non-zero.

The analysis builds on the example developed in Section 3.3. The MAK filter state estimate $\hat{\mathbf{x}}_{mav}$ and covariance matrix $\mathbf{P}_{mav}$ are initially assumed to be

$$\hat{\mathbf{x}}_{mav}(0|0) = \begin{bmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{p}}_1 \\ \hat{\mathbf{p}}_2 \end{bmatrix} (0|0) \tag{3.14}$$

$$\mathbf{P}_{mav}(0|0) = \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_{22} \end{bmatrix} (0|0), \tag{3.15}$$

hence the covariances between all estimates are initially assumed zero. Two consecutive updates are considered. The first update follows an observation of p_1 and the second update follows an observation of p_2. Therefore,

$$\mathbf{H}_1(1) \triangleq \begin{bmatrix} \mathbf{H}_v & \mathbf{H}_p & \mathbf{0} \end{bmatrix} (1) \tag{3.16}$$

$$\mathbf{H}_1(2) \triangleq \begin{bmatrix} \mathbf{H}_v & 0 & \mathbf{H}_p \end{bmatrix} (2). \tag{3.17}$$

Examining Equation 3.5, it can be seen that the prediction $\mathbf{P}_{mav}(1|0)$ does not change the structure of $\mathbf{P}_{mav}(0|0)$, shown in Equation 3.15, hence $\mathbf{P}_{mav}(1|0)$ may be redefined as $\mathbf{P}_{mav}(1|0) = \mathbf{P}_{mav}(0|0)$ for the purpose of the analysis.

The first update is computed using

$$\begin{aligned} \mathbf{S}_1(1|0) &= \mathbf{H}_1(1)\mathbf{P}_{mav}(1|0)\mathbf{H}_1^T(1) + \mathbf{H}_w(1)\mathbf{R}(1)\mathbf{H}_w^T(1) \\ \mathbf{W}_1(1) &= \mathbf{P}_{mav}(1|0)\mathbf{H}_1^T(1)\mathbf{S}_1^{-1}(1|0) \\ \hat{\mathbf{x}}_{mav}(1|1) &= \hat{\mathbf{x}}_{mav}(1|0) + \mathbf{W}_1(1)\nu(1) \\ \mathbf{P}_{mav}(1|1) &= \mathbf{P}_{mav}(1|0) - \mathbf{W}_1(1)\mathbf{S}_1(1|0)\mathbf{W}_1^T(1), \end{aligned}$$

where $\nu(1)$ is the innovation at $k = 1$. The MAK filter weighting matrix $\mathbf{W}_1(1)$ is

$$
\begin{aligned}
\mathbf{W}_1(1) &= \begin{bmatrix} \mathbf{S}_{\mathbf{x}_v\nu}\mathbf{S}_{\nu\nu}^{-1} \\ \mathbf{S}_{\mathbf{p}_1\nu}\mathbf{S}_{\nu\nu}^{-1} \\ \mathbf{S}_{\mathbf{p}_2\nu}\mathbf{S}_{\nu\nu}^{-1} \end{bmatrix} (1|0) \\
&= \begin{bmatrix} \mathbf{P}_{vv}\mathbf{H}_v^T + \mathbf{P}_{v1}\mathbf{H}_p^T \\ \mathbf{P}_{1v}\mathbf{H}_v^T + \mathbf{P}_{11}\mathbf{H}_p^T \\ \mathbf{P}_{2v}\mathbf{H}_v^T + \mathbf{P}_{21}\mathbf{H}_p^T \end{bmatrix} (1|0)\mathbf{S}_1^{-1}(1|0) \\
&= \begin{bmatrix} \mathbf{P}_{vv}\mathbf{H}_v^T \\ \mathbf{P}_{11}\mathbf{H}_p^T \\ \mathbf{0} \end{bmatrix} (1|0)\mathbf{S}_1^{-1}(1|0), \qquad (3.18)
\end{aligned}
$$

and state estimate update is written explicitly as

$$\hat{\mathbf{x}}_v(1|1) = \hat{\mathbf{x}}_v(1|0) + \mathbf{P}_{vv}(1|0)\mathbf{H}_v^T(1)\nu(1) \qquad (3.19)$$

$$\hat{\mathbf{p}}_1(1|1) = \hat{\mathbf{p}}_1(1|0) + \mathbf{P}_{11}(1|0)\mathbf{H}_p^T(1)\nu(1) \qquad (3.20)$$

$$\hat{\mathbf{p}}_2(1|1) = \hat{\mathbf{p}}_2(1|0). \qquad (3.21)$$

Since $\mathbf{H}_1(1)$ represents a relative observation of feature p_1 with respect to the vehicle, $\mathbf{H}_v(1)$ and $\mathbf{H}_p(1)$ are both non-zero. This correlates the innovation $\nu(1)$ with the error $\tilde{\mathbf{x}}_v(1|0)$ in $\hat{\mathbf{x}}_v(1|0)$, and also with the error $\tilde{\mathbf{p}}_1(1|0)$ in $\hat{\mathbf{p}}_1(1|0)$. The non-zero term $\mathbf{P}_{vv}(1|0)\mathbf{H}_v^T(1) = \mathbf{S}_{\mathbf{x}_v\nu}(1|0)$ indicates the correlation between $\tilde{\mathbf{x}}_v(1|0)$ and $\nu(1)$, and the non-zero term $\mathbf{P}_{11}(1|0)\mathbf{H}_p^T(1) = \mathbf{S}_{\mathbf{p}_1\nu}(1|0)$ indicates the correlation between $\tilde{\mathbf{p}}_1(1|0)$ and $\nu(1)$. These correlations allow the MAK filter to update the estimate $\hat{\mathbf{x}}_v(1|0)$ of the vehicle state, Equation 3.19, and the estimate $\hat{\mathbf{p}}_1(1|0)$ of the p_1 feature state, Equation 3.20.

If $\mathbf{H}_1(1)$ consisted only of $\mathbf{H}_p(1)$, in effect making $\mathbf{H}_v(1)$ zero and corresponding to a direct observation of feature p_1, the innovation would be correlated only with the error $\tilde{\mathbf{p}}_1(1|0)$, as seen in $\mathbf{P}_{vv}(1|0)\mathbf{H}_v^T(1) = \mathbf{S}_{\mathbf{x}_v\nu}(1|0) = \mathbf{0}$. Consequently, only $\hat{\mathbf{p}}_1(1|0)$ would be updated and $\hat{\mathbf{x}}_v(1|1) = \hat{\mathbf{x}}_v(1|0)$. Similarly, if $\mathbf{H}_1(1)$ consisted only of $\mathbf{H}_v(1)$, in effect making $\mathbf{H}_p(1)$ zero and corresponding to a direct observation of the vehicle, the innovation would be correlated only with the error $\tilde{\mathbf{x}}_v(1|0)$, as seen in $\mathbf{P}_{11}(1|0)\mathbf{H}_p^T(1) = \mathbf{S}_{\mathbf{p}_1\nu}(1|0) = \mathbf{0}$. Consequently, only $\hat{\mathbf{x}}_v(1|0)$ would be updated and $\hat{\mathbf{p}}_1(1|1) = \hat{\mathbf{p}}_1(1|0)$.

Writing the covariance update explicitly as well,

$$\begin{aligned}\mathbf{P}_{vv}(1|1) &= \mathbf{P}_{vv}(1|0) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{x}_v}\right](1|0) \\ &= \mathbf{P}_{vv}(1|0) - \left[\mathbf{P}_{vv}\mathbf{H}_v^T\mathbf{S}_1^{-1}\mathbf{H}_v\mathbf{P}_{vv}\right](1|0) \qquad (3.22)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{v1}(1|1) &= \mathbf{P}_{v1}(1|0) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{p}_1}\right](1|0) \\ &= -\left[\mathbf{P}_{vv}\mathbf{H}_v^T\mathbf{S}_1^{-1}\mathbf{H}_p\mathbf{P}_{11}\right](1|0) \qquad (3.23)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{v2}(1|1) &= \mathbf{P}_{v2}(1|0) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{p}_2}\right](1|0) \\ &= \mathbf{0} \qquad (3.24)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{11}(1|1) &= \mathbf{P}_{11}(1|0) - \left[\mathbf{W}_{\mathbf{p}_1}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{p}_1}\right](1|0) \\ &= \mathbf{P}_{11}(1|0) - \left[\mathbf{P}_{11}\mathbf{H}_p^T\mathbf{S}_1^{-1}\mathbf{H}_p\mathbf{P}_{11}\right](1|0) \qquad (3.25)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{12}(1|1) &= \mathbf{P}_{12}(1|0) - \left[\mathbf{W}_{\mathbf{p}_1}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{p}_2}\right](1|0) \\ &= \mathbf{0} \qquad (3.26)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{22}(1|1) &= \mathbf{P}_{22}(1|0) - \left[\mathbf{W}_{\mathbf{p}_2}\mathbf{S}_1^{-1}\mathbf{W}_{\mathbf{p}_2}\right](1|0) \\ &= \mathbf{P}_{22}(1|0), \qquad (3.27)\end{aligned}$$

leads to the following structure in $\mathbf{P}_{mav}(1|1)$,

$$\mathbf{P}_{mav}(1|1) = \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{P}_{v1} & \mathbf{0} \\ \mathbf{P}_{1v} & \mathbf{P}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_{22} \end{bmatrix}(1|1). \qquad (3.28)$$

The error in the estimate of the vehicle state $\hat{\mathbf{x}}_v(1|1)$ and the error in the feature estimate $\hat{\mathbf{p}}_1(1|1)$ are now correlated, as shown by the non-zero term $\mathbf{P}_{v1}(1|1)$. This correlation arises because a relative observation of p_1 is used in the update. The terms $\mathbf{H}_v(1)$ and $\mathbf{H}_p(1)$ both appear in the update of $\mathbf{P}_{v1}(1|0)$, Equation 3.23, and only if both are non-zero, which is a direct consequence of a relative observation, will $\mathbf{P}_{v1}(1|1)$ be non-zero. This result can be interpreted as follows. If $\mathbf{H}_v(1)$ and $\mathbf{H}_p(1)$ are both non-zero, then, according to Equations 3.19 and 3.20, the vehicle state estimate $\hat{\mathbf{x}}_v(1|0)$ and

the feature p_1 state estimate $\hat{\mathbf{p}}_1(1|0)$ are both update using the same observation. This correlates the errors in $\hat{\mathbf{x}}_v(1|1)$ and $\hat{\mathbf{p}}_1(1|1)$, leading to the non-zero term $\mathbf{P}_{v1}(1|1)$.

If $\mathbf{H}_1(1)$ consisted only of $\mathbf{H}_p(1)$, only $\mathbf{P}_{11}(1|0)$ would be updated, according to Equation 3.25. All other terms in $\mathbf{P}_{mav}(1|0)$ would remain unchanged. Similarly, if $\mathbf{H}_1(1)$ consisted only of $\mathbf{H}_v(1)$, only $\mathbf{P}_{vv}(1|0)$ would be updated, according to Equation 3.22, and all other terms in $\mathbf{P}_{mav}(1|0)$ would remain unchanged. This is consistent with the estimate updates deduced for these two cases.

> A relative observation update correlates the error in the vehicle estimate with the error in the observed feature estimate, because the same observation is used to update the observed feature and the vehicle estimate.

Equation 3.5 is used again to conclude that the prediction $\mathbf{P}_{mav}(2|1)$ does not change the structure of $\mathbf{P}_{mav}(1|1)$, shown in Equations 3.28. Therefore $\mathbf{P}_{mav}(2|1)$ may be again redefined as $\mathbf{P}_{mav}(2|1) = \mathbf{P}_{mav}(1|1)$.

The second update follows the observation of feature p_2, and is therefore computed using

$$\begin{aligned}
\mathbf{S}_2(2|1) &= \mathbf{H}_2(2)\mathbf{P}_{mav}(2|1)\mathbf{H}_2^T(2) + \mathbf{H}_w\mathbf{R}(2)\mathbf{H}_w^T \\
\mathbf{W}_2(2) &= \mathbf{P}_{mav}(2|1)\mathbf{H}_2^T(2)\mathbf{S}_2^{-1}(2|1) \\
\hat{\mathbf{x}}_{mav}(2|2) &= \hat{\mathbf{x}}_{mav}(2|1) + \mathbf{W}_2(2)\nu(2) \\
\mathbf{P}_{mav}(2|2) &= \mathbf{P}_{mav}(2|1) - \mathbf{W}_2(2)\mathbf{S}_2(2|1)\mathbf{W}_2^T(2),
\end{aligned}$$

where $\nu(2)$ is the innovation at $k = 2$. The MAK filter weighting matrix for the second update, $\mathbf{W}_2(2)$, is

$$\begin{aligned}
\mathbf{W}_2(2) &= \begin{bmatrix} \mathbf{S}_{\mathbf{x}_v\nu}\mathbf{S}_{\nu\nu}^{-1} \\ \mathbf{S}_{\mathbf{p}_1\nu}\mathbf{S}_{\nu\nu}^{-1} \\ \mathbf{S}_{\mathbf{p}_2\nu}\mathbf{S}_{\nu\nu}^{-1} \end{bmatrix}(2|1) \\
&= \begin{bmatrix} \mathbf{P}_{vv}\mathbf{H}_v^T + \mathbf{P}_{v2}\mathbf{H}_p^T \\ \mathbf{P}_{1v}\mathbf{H}_v^T + \mathbf{P}_{12}\mathbf{H}_p^T \\ \mathbf{P}_{2v}\mathbf{H}_v^T + \mathbf{P}_{22}\mathbf{H}_p^T \end{bmatrix}(2|1)\mathbf{S}_2^{-1}(2|1) \\
&= \begin{bmatrix} \mathbf{P}_{vv}\mathbf{H}_v^T \\ \mathbf{P}_{1v}\mathbf{H}_v^T \\ \mathbf{P}_{22}\mathbf{H}_p^T \end{bmatrix}(2|1)\mathbf{S}_2^{-1}(2|1). \qquad (3.29)
\end{aligned}$$

The second state estimate update is written explicitly as

$$\hat{\mathbf{x}}_v(2|2) = \hat{\mathbf{x}}_v(2|1) + \mathbf{P}_{vv}(2|1)\mathbf{H}_v^T(2)\nu(2) \tag{3.30}$$

$$\hat{\mathbf{p}}_1(2|2) = \hat{\mathbf{p}}_1(2|1) + \mathbf{P}_{1v}(2|1)\mathbf{H}_v^T(2)\nu(2) \tag{3.31}$$

$$\hat{\mathbf{p}}_2(2|2) = \hat{\mathbf{p}}_2(2|1) + \mathbf{P}_{22}(2|1)\mathbf{H}_p^T(2)\nu(2). \tag{3.32}$$

As in the first update, the relative observation correlates the innovation $\nu(2)$ with the error in the vehicle estimate, $\tilde{\mathbf{x}}_v(2|1)$, and also with the error in the observed feature estimate, $\tilde{\mathbf{p}}_2(2|1)$. However, this time the innovation is also correlated with the error $\tilde{\mathbf{p}}_1(2|1)$ in the unobserved feature estimate, since $\tilde{\mathbf{p}}_1(2|1)$ is correlated with $\tilde{\mathbf{x}}_v(2|1)$. Therefore, $\mathbf{S}_{\mathbf{x}_v\nu}(2|1)$, $\mathbf{S}_{\mathbf{p}_1\nu}(2|1)$ and $\mathbf{S}_{\mathbf{p}_2\nu}(2|1)$ are all non-zero and each of $\hat{\mathbf{x}}_v(2|1)$, $\hat{\mathbf{p}}_1(2|1)$, and $\hat{\mathbf{p}}_2(2|1)$ is updated. If $\mathbf{H}_2(2)$ consisted only of $\mathbf{H}_p(2)$, corresponding to a direct observation of p_2, the innovation would only be correlated with $\tilde{\mathbf{p}}_2(2|1)$, hence only $\hat{\mathbf{p}}_2(2|1)$ would be updated. However, if $\mathbf{H}_2(2)$ consisted only of $\mathbf{H}_v(2)$, corresponding to a direct observation of the vehicle, the innovation would be correlated with $\tilde{\mathbf{x}}_v(2|1)$ and also with $\tilde{\mathbf{p}}_1(2|1)$, since $\tilde{\mathbf{p}}_1(2|1)$ is correlated with $\tilde{\mathbf{x}}_v(2|1)$ due to the first update. Therefore, $\hat{\mathbf{x}}_v(2|1)$ and $\hat{\mathbf{p}}_1(2|1)$ would be updated.

Writing the second covariance update explicitly,

$$\begin{aligned}\mathbf{P}_{vv}(2|2) &= \mathbf{P}_{vv}(2|1) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{x}_v}\right](2|1) \\ &= \mathbf{P}_{vv}(2|1) - \left[\mathbf{P}_{vv}\mathbf{H}_v^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_v\mathbf{P}_{vv}\right](2|1)\end{aligned} \tag{3.33}$$

$$\begin{aligned}\mathbf{P}_{v1}(2|2) &= \mathbf{P}_{v1}(2|1) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{p}_1}\right](2|1) \\ &= \mathbf{P}_{v1}(2|1) - \left[\mathbf{P}_{vv}\mathbf{H}_v^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_v\mathbf{P}_{v1}\right](2|1)\end{aligned} \tag{3.34}$$

$$\begin{aligned}\mathbf{P}_{v2}(2|2) &= \mathbf{P}_{v2}(2|1) - \left[\mathbf{W}_{\mathbf{x}_v}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{p}_2}\right](2|1) \\ &= -\left[\mathbf{P}_{vv}\mathbf{H}_v^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_p\mathbf{P}_{22}\right](2|1)\end{aligned} \tag{3.35}$$

$$\begin{aligned}\mathbf{P}_{11}(2|2) &= \mathbf{P}_{11}(2|1) - \left[\mathbf{W}_{\mathbf{p}_1}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{p}_1}\right](2|1) \\ &= \mathbf{P}_{11}(2|1) - \left[\mathbf{P}_{1v}\mathbf{H}_v^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_v\mathbf{P}_{v1}\right](2|1) \qquad (3.36)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{12}(2|2) &= \mathbf{P}_{12}(2|1) - \left[\mathbf{W}_{\mathbf{p}_1}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{p}_2}\right](2|1) \\ &= -\left[\mathbf{P}_{1v}\mathbf{H}_v^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_p\mathbf{P}_{22}\right](2|1) \qquad (3.37)\end{aligned}$$

$$\begin{aligned}\mathbf{P}_{22}(2|2) &= \mathbf{P}_{22}(2|1) - \left[\mathbf{W}_{\mathbf{p}_2}\mathbf{S}_2^{-1}\mathbf{W}_{\mathbf{p}_2}\right](2|1) \\ &= \mathbf{P}_{22}(2|1) - \left[\mathbf{P}_{22}\mathbf{H}_p^T\right]\mathbf{S}_2^{-1}\left[\mathbf{H}_p\mathbf{P}_{22}\right](2|1), \qquad (3.38)\end{aligned}$$

leads to the following structure in $\mathbf{P}_{mav}(2|2)$,

$$\mathbf{P}_{mav}(2|2) = \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{P}_{v1} & \mathbf{P}_{v2} \\ \mathbf{P}_{1v} & \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{2v} & \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix}(2|2). \qquad (3.39)$$

Each term in $\mathbf{P}_{mav}(2|2)$ is non-zero, indicating that all estimate errors are now correlated. As in the first update, the relative observation correlates the error in the vehicle estimate and the observed feature estimate, shown by the non-zero term $\mathbf{P}_{v2}(2|2)$. Again, both $\mathbf{H}_v(2)$ and $\mathbf{H}_p(2)$ have to be non-zero in the update of $\mathbf{P}_{v2}(2|1)$, Equation 3.35, for $\mathbf{P}_{v2}(2|2)$ to be non-zero.

If $\mathbf{H}_2(2)$ consisted only of $\mathbf{H}_p(2)$, only $\mathbf{P}_{22}(2|1)$ would be updated, according to Equation 3.38. All other terms in $\mathbf{P}_{mav}(2|1)$ would remain unchanged. However, if $\mathbf{H}_2(2)$ consisted only of $\mathbf{H}_v(2)$, the covariance update would not be limited to $\mathbf{P}_{vv}(2|1)$. The update would extend to the terms $\mathbf{P}_{11}(2|1)$, Equation 3.36, and $\mathbf{P}_{v1}(2|1)$, Equation 3.34, as well. This is due to the correlation between the error in $\hat{\mathbf{p}}_1(2|1)$ and $\hat{\mathbf{x}}_v(2|1)$, produced by the first update and represented by the non-zero term $\mathbf{P}_{v1}(2|1)$, and reflects the fact that both $\hat{\mathbf{x}}_v(2|1)$ and $\hat{\mathbf{p}}_1(2|1)$ are updated if $\mathbf{H}_v(2)$ appears in $\mathbf{H}_2(2)$.

Finally, the relative observation also leads to a non-zero term $\mathbf{P}_{12}(2|2)$. The update of $\mathbf{P}_{12}(2|1)$, Equation 3.37, involves the terms $\mathbf{H}_p(2)$ and $\mathbf{H}_v(2)$. Only if both are non-zero, corresponding to a relative observation, will $\mathbf{P}_{12}(2|2)$ be non-zero. This can be interpreted as follows. When $\mathbf{H}_v(2)$ and $\mathbf{H}_p(2)$ are non-zero, the observation is used to update $\hat{\mathbf{p}}_1(2|1)$, due to the non-zero term $\mathbf{P}_{v1}(2|1)$, and is also used to update $\hat{\mathbf{p}}_2(2|1)$. Since the same observation is used to update $\hat{\mathbf{p}}_1(2|1)$ and $\hat{\mathbf{p}}_2(2|1)$, the errors in $\hat{\mathbf{p}}_1(2|1)$ and $\hat{\mathbf{p}}_2(2|1)$ will become correlated, leading to the non-zero term $\mathbf{P}_{12}(2|2)$.

The innovation of a relative observation of feature p_i is correlated with the vehicle estimate error and the feature p_i estimate error. If the vehicle estimate error is initially correlated with a feature p_j estimate error, then the innovation will also be correlated with the feature p_j estimate error. Consequently, an update of feature p_i with the innovation will correlate the feature p_i estimate error with the feature p_j estimate error. Therefore, estimating the vehicle and the feature states using relative observations necessarily correlates all estimate errors with each other.

In Section 3.3 it was shown that each correlation between the vehicle and a feature and between two features must be maintained by the MAK covariance matrix in order to achieve a consistent update. Since these correlations have been identified in this section as unavoidable, an important conclusion follows.

It is not possible to solve the SLAM Problem for N features as N independent single feature filtering problems. To solve the SLAM Problem the vehicle state estimate $\hat{\mathbf{x}}_v$ has to be augmented with each feature state estimate $\hat{\mathbf{p}}_i$, where $i = 1, 2, \ldots, N$. Associated with the augmented state, denoted $\mathbf{x}_{mav}$, is an augmented covariance matrix, denoted $\mathbf{P}_{mav}$. During an update each term in $\mathbf{x}_{mav}$ and each term in $\mathbf{P}_{mav}$ must be updated if all the estimates are to remain consistent with their covariances.

The dimension of the map augmented vehicle state $\mathbf{x}_{mav}$ grows linearly with the number of features. Consequently, the dimension of the corresponding covariance matrix $\mathbf{P}_{mav}$ is of the order N^2 for an environment of N features. During an update each term of $\mathbf{P}_{mav}$ must be updated. This leads to the following conclusion. [2]

For an environment of N features, the solution to the SLAM Problem requires storage of the order of N^2 and computation of at least the same order.

3.5 Evolution of the Map

In the previous section is was established that the vehicle state estimate must be augmented with each feature state estimate in order to solve the SLAM Problem in a consistent manner. Once the map augmented vehicle state vector is constructed, the Kalman filter is used to estimate the state of the features and the vehicle simultaneously. The operation of the Kalman filter in this case proceeds as described in Chapter 2.

The value of the SLAM structure derived in Section 3.2 is further demonstrated by considering the evolution of the map. The analysis of the map evolution requires some observations to be made

[2] It is shown by Moutarlier [72] that the update of the covariance matrix of N feature estimates is of order N^2 if the features are observed individually.

about the properties of the matrices arising in the Kalman filter. In the following, the standard matrix theorems may be found in Appendix A.

Using Theorem 10 the following properties are established,

$$\begin{aligned}
\mathbf{P}(k|k) &= \mathrm{E}\left[\tilde{\mathbf{x}}(k|k)\tilde{\mathbf{x}}(k|k)^T\right] &\Rightarrow\ & \mathbf{P}(k|k) \text{ is psd} \\
\mathbf{Q}(k) &= \mathrm{E}\left[\mathbf{v}(k)\mathbf{v}^T(k)\right] &\Rightarrow\ & \mathbf{Q}(k) \text{ is psd} \\
\mathbf{R}(k) &= \mathrm{E}\left[\mathbf{w}(k)\mathbf{w}^T(k)\right] &\Rightarrow\ & \mathbf{R}(k) \text{ is psd,}
\end{aligned}$$

where psd stands for positive semidefinite. Using Theorems 11 and 7 it follows that

$$\begin{aligned}
\mathbf{P}(k+1|k) &= \nabla\mathbf{f}_{\mathbf{x}}\mathbf{P}_{mav}(k|k)\nabla\mathbf{f}_{\mathbf{x}}^T + \nabla\mathbf{f}_{\mathbf{v}}\mathbf{Q}(k)\nabla\mathbf{f}_{\mathbf{v}}^T \\
&\Rightarrow \mathbf{P}(k+1|k) \text{ is psd} \\
\\
\mathbf{S}(k+1|k) &= \nabla\mathbf{h}_{\mathbf{x}}\mathbf{P}(k+1|k)\nabla\mathbf{h}_{\mathbf{x}}^T + \nabla\mathbf{h}_{\mathbf{w}}\mathbf{R}(k+1)\nabla\mathbf{h}_{\mathbf{w}}^T \\
&\Rightarrow \mathbf{S}(k+1|k) \text{ is psd} \\
\\
&\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1) \text{ is psd.}
\end{aligned}$$

Therefore, the matrices $\mathbf{P}(k|k)$ and $\mathbf{P}(k+1|k)$, and the product $\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)$ are all positive semidefinite. This is a general result for the Kalman filter, and hence it follows that $\mathbf{P}_{mav}(k|k)$ and $\mathbf{P}_{mav}(k+1|k)$, and $\mathbf{W}_i(k+1)\mathbf{S}_i(k+1)\mathbf{W}_i^T(k+1)$ are also all positive semidefinite.

These results will be used in the next section to derive an important property of the determinant of the covariance matrix $\mathbf{P}$, and therefore $\mathbf{P}_{mav}$, which in turn will be used to explain the evolution of the map.

3.5.1 Evolution of the Feature Estimates

The evolution of sections of the map, groups of features, and in particular the evolution of individual feature estimates is considered. The general Kalman filter covariance matrix update equation is

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1). \tag{3.40}$$

Theorem 12 states that the determinant of a psd matrix cannot decrease when another psd matrix is added to it. Using Theorem 12, Equation 3.40 and the fact that $\mathbf{P}(k+1|k+1)$, $\mathbf{P}(k+1|k)$ and $\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)$ are all psd leads to

$$\begin{aligned}\det(\mathbf{P}(k+1|k+1)) &= \det(\mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1))\\ \det(\mathbf{P}(k+1|k+1)) &\leq \det(\mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)\\ &\quad +\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)),\end{aligned}$$

and hence

$$\det\left(\mathbf{P}(k+1|k+1)\right) \leq \det\left(\mathbf{P}(k+1|k)\right). \tag{3.41}$$

This result states that the determinant of the covariance matrix $\mathbf{P}$ cannot increase during an update.

A similar result can be derived for certain submatrices of $\mathbf{P}$. Since matrix addition and subtraction is performed element wise, if the same submatrix is formed in $\mathbf{P}(k+1|k+1)$, $\mathbf{P}(k+1|k)$ and $\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)$, then Equation 3.40 is still valid for all submatrices. The particular submatrices that will be considered are principle submatrices, constructed by deleting any number of columns and the corresponding rows. In general, if $[\mathbf{M}]_p$ denotes a principle submatrix of a matrix $\mathbf{M}$, then

$$[\mathbf{P}(k+1|k+1)]_p = [\mathbf{P}(k+1|k)]_p - [\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)]_p. \tag{3.42}$$

Theorem 13 states that any principal submatrix of a psd matrix will also be psd, therefore an equation similar to Equation 3.41 is derived for the principle submatrices of $\mathbf{P}(k+1|k+1)$ and $\mathbf{P}(k+1|k)$,

$$\begin{aligned}\det([\mathbf{P}(k+1|k+1)]_p) &= \det([\mathbf{P}(k+1|k)]_p - [\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1))]_p\\ \det([\mathbf{P}(k+1|k+1)]_p) &\leq \det([\mathbf{P}(k+1|k)]_p - [\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)]_p\\ &\quad +[\mathbf{W}(k+1)\mathbf{S}(k+1|k)\mathbf{W}^T(k+1)]_p),\end{aligned}$$

and hence

$$\det\left([\mathbf{P}(k+1|k+1)]_p\right) \leq \det\left([\mathbf{P}(k+1|k)]_p\right).$$

This result states that the determinant of any principle submatrix of the covariance matrix $\mathbf{P}$, and hence $\mathbf{P}_{mav}$, cannot increase during an update. In particular, the feature covariances $\mathbf{P}_{ii}$ are principle submatrices of $\mathbf{P}_{mav}$, as is the vehicle covariance $\mathbf{P}_{vv}$, so two important results follow,

$$\begin{aligned}\det(\mathbf{P}_{vv}(k+1|k+1)) &\leq \det(\mathbf{P}_{vv}(k+1|k)) \\ \det(\mathbf{P}_{ii}(k+1|k+1)) &\leq \det(\mathbf{P}_{ii}(k+1|k)).\end{aligned}$$

The determinant of the covariance matrix of any feature and the determinant of the covariance matrix of the vehicle do not increase during an update. Forming other principle matrices, the determinant of the covariance matrix of any group of estimates can be shown not to increase during an update. The groups may include the vehicle, or may consist only of feature estimates. For example, any group of two estimates taken from the map augmented vehicle state vector satisfies

$$\det\left(\begin{bmatrix}\mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm}\end{bmatrix}_{any}(k+1|k+1)\right) \leq \det\left(\begin{bmatrix}\mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm}\end{bmatrix}_{any}(k+1|k)\right),$$

where $\mathbf{P}_{nm}$ denotes the covariance between block n and m of $\mathbf{P}_{mav}$, and n or m may refer to the vehicle. The common time indices have been taken outside the submatrix to improve clarity.

As the prediction stage does not affect the feature estimates, Equation 3.5, it follows for the feature estimate covariance submatrices that

$$\begin{aligned}\det(\mathbf{P}_{ii}(k+1|k)) &= \det(\mathbf{P}_{ii}(k|k)) \\ \det\left(\begin{bmatrix}\mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm}\end{bmatrix}_{map}(k+1|k)\right) &= \det\left(\begin{bmatrix}\mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm}\end{bmatrix}_{map}(k|k)\right),\end{aligned}$$

and hence

$$\begin{aligned} \det(\mathbf{P}_{ii}(k+1|k+1)) &\leq \det(\mathbf{P}_{ii}(k|k)) \\ \det\left(\begin{bmatrix} \mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm} \end{bmatrix}_{map}(k+1|k+1)\right) &\leq \det\left(\begin{bmatrix} \mathbf{P}_{nn} & \mathbf{P}_{nm} \\ \mathbf{P}_{mn} & \mathbf{P}_{mm} \end{bmatrix}_{map}(k|k)\right), \end{aligned}$$

where $\mathbf{P}_{nm}$ again denotes the covariance between block n and m of $\mathbf{P}_{mav}$, but n and m only refer to features. This leads to another important conclusion.

> The determinant of the covariance matrix of all the feature estimates and the determinant of the covariance matrix of any group of features are both monotonically non-increasing functions of the time step. In particular, the determinant of the covariance matrix of each individual feature estimate is a non-increasing function of the time step.

The individual dimensions of the vehicle and the feature estimates can be examined by considering the diagonal entries of $\mathbf{P}_{mav}$. The q^{th} diagonal entry of $\mathbf{P}_{mav}$ is the variance of the estimate of the q^{th} dimension of the feature augmented state vector. It can be seen from Equation 3.40 that the update of the q^{th} diagonal entry of $\mathbf{P}_{mav}$ is formed by the subtraction of the q^{th} diagonal entry of $\mathbf{W}_i\mathbf{S}_i\mathbf{W}_i^T$, which, as shown by Theorem 13, must be non-negative. It follows that the variance in each dimension of the estimate of the feature augmented vehicle state cannot increase during an update,

$$\sigma^2_{any}(k+1|k+1) \leq \sigma^2_{any}(k+1|k),$$

where σ^2_{any} is the variance in some dimension of the estimate of the vehicle or a feature.

Further, since the prediction stage does not affect the feature estimates, the variance of the estimate in each dimension of each feature is a monotonically non-increasing function of the time step,

$$\sigma^2_{map}(k+1|k+1) \leq \sigma^2_{map}(k|k),$$

where σ^2_{map} is the variance in some dimension of the estimate of a feature.

> The map produced by the Kalman in the map augmented vehicle state vector form consists of feature estimates with non-increasing variances in each dimension.

3.5.2 Evolution of Relative Distances

Having established the behaviour of individual feature estimates and groups of feature estimates, the evolution of the estimate of relative distances between any two features in the map is now investigated. Selecting two arbitrary features p_i and p_j of the map, the relative distance can be expressed as a vector $\Delta\mathbf{p}_{ij}$,

$$\begin{aligned}\Delta\mathbf{p}_{ij} &= \begin{bmatrix} p_{j_1} - p_{i_1} \\ \vdots \\ p_{j_n} - p_{i_n} \end{bmatrix} \\ &= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_j \end{bmatrix},\end{aligned}$$

where p_{i_j} is the j^{th} dimension of feature p_i. A covariance matrix can be associated with this vector, defined in the usual way,

$$\begin{aligned}\Delta\mathbf{P}_{ij} &\triangleq E[(\Delta\tilde{\mathbf{p}}_{ij})(\Delta\tilde{\mathbf{p}}_{ij})^T] \\ &= \mathrm{E}\left[\begin{bmatrix} (\hat{p}_{j_1} - \hat{p}_{i_1}) - (p_{j_1} - p_{i_1}) \\ \vdots \\ (\hat{p}_{j_n} - \hat{p}_{i_n}) - (p_{j_n} - p_{i_n}) \end{bmatrix} \begin{bmatrix} (\hat{p}_{j_1} - \hat{p}_{i_1}) - (p_{j_1} - p_{i_1}) \\ \vdots \\ (\hat{p}_{j_n} - \hat{p}_{i_n}) - (p_{j_n} - p_{i_n}) \end{bmatrix}^T\right] \\ &= \mathrm{E}\left[\begin{bmatrix} (\hat{p}_{j_1} - p_{j_1}) - (\hat{p}_{i_1} - p_{i_1}) \\ \vdots \\ (\hat{p}_{j_n} - p_{j_n}) - (\hat{p}_{i_n} - p_{i_n}) \end{bmatrix} \begin{bmatrix} (\hat{p}_{j_1} - p_{j_1}) - (\hat{p}_{i_1} - p_{i_1}) \\ \vdots \\ (\hat{p}_{j_n} - p_{j_n}) - (\hat{p}_{i_n} - p_{i_n}) \end{bmatrix}^T\right] \\ &= \mathrm{E}\left[\begin{bmatrix} \tilde{p}_{j_1} - \tilde{p}_{i_1} \\ \vdots \\ \tilde{p}_{j_n} - \tilde{p}_{i_n} \end{bmatrix} \begin{bmatrix} \tilde{p}_{j_1} - \tilde{p}_{i_1} \\ \vdots \\ \tilde{p}_{j_n} - \tilde{p}_{i_n} \end{bmatrix}^T\right] \\ &= \mathrm{E}\left[\begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}_i \\ \tilde{\mathbf{p}}_j \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}_i \\ \tilde{\mathbf{p}}_j \end{bmatrix}^T \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix}\right] \\ &= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \mathrm{E}\left[\begin{bmatrix} \tilde{\mathbf{p}}_i \\ \tilde{\mathbf{p}}_j \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{p}}_i \\ \tilde{\mathbf{p}}_j \end{bmatrix}^T\right] \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} \\ &= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix} \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix},\end{aligned} \tag{3.43}$$

and hence

$$\det\left(\Delta\mathbf{P}_{ij}\right) = \det\left(\begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix}\begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix}\right).$$

Choosing the principle submatrix in Equation 3.42 to include only the covariances of features p_i and p_j and denoting the positive semidefinite matrix $[\mathbf{W}_d(k+1)\mathbf{S}_d(k+1|k)\mathbf{W}_d^T(k+1)]_p$, derived from the observation of an arbitrary feature p_d, by $\mathbf{M}_{psd}$ leads to

$$\begin{aligned} [\mathbf{P}_{mav}]_p &= \begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix} \\ [\mathbf{P}_{mav}(k+1|k)]_p &= [\mathbf{P}_{mav}(k+1|k+1)]_p + \mathbf{M}_{psd} \\ [\mathbf{P}_{mav}(k|k)]_p &= [\mathbf{P}_{mav}(k+1|k+1)]_p + \mathbf{M}_{psd}. \end{aligned} \tag{3.44}$$

The final expression follows since the prediction does not affect the feature estimates. Thus

$$\begin{aligned} \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix}[\mathbf{P}_{mav}(k|k)]_p\begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} &= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix}[\mathbf{P}_{mav}(k+1|k+1)]_p\begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} \\ &\quad + \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix}\mathbf{M}_{psd}\begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix}. \end{aligned}$$

Combining with Equation 3.43 and using Theorems 11 and 12 the following result is established, where $\mathbf{M}'_{psd}$ is psd,

$$\begin{aligned} \det\left(\Delta\mathbf{P}_{ij}(k|k)\right) &= \det\left(\Delta\mathbf{P}_{ij}(k+1|k+1) + \mathbf{M}'_{psd}\right) \\ &\geq \det\left(\Delta\mathbf{P}_{ij}(k+1|k+1)\right) \\ \Rightarrow \det\left(\Delta\mathbf{P}_{ij}(k+1|k+1)\right) &\leq \det\left(\Delta\mathbf{P}_{ij}(k|k)\right). \end{aligned} \tag{3.45}$$

Therefore, the following conclusion is reached.

The determinant of the covariance matrices of the relative distance vectors between all possible pairs of features are monotonically non-increasing functions of the time step.

Similarly to the investigation of individual dimensions described in Section 3.5.1, it is possible to analyse individual dimensions of a relative distance. The scalar relative distance in a particular dimension m between features p_i and p_j shall be denoted $\Delta_m p_{ij}$,

$$\begin{aligned}
\Delta_m p_{ij} &= p_{j_m} - p_{i_m} \\
&= \begin{bmatrix} \dots & 0 & -1 & 0 & \dots & 0 & 1 & 0 & \dots \end{bmatrix} \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_j \end{bmatrix} \\
&= \mathbf{u}_m^T \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_j \end{bmatrix}.
\end{aligned}$$

Denoting the variance of $\Delta_m p_{ij}$ by $\Delta_m P_{ij}$ and following the derivation of Equation 3.43 leads to

$$\Delta_m P_{ij} = \mathbf{u}_m^T \begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix} \mathbf{u}_m. \tag{3.46}$$

Combining Equations 3.44 and 3.46 shows that

$$\begin{aligned}
\mathbf{u}_m^T \left[\mathbf{P}_{mav}(k|k)\right]_p \mathbf{u}_m &= \mathbf{u}_m^T \left[\mathbf{P}_{mav}(k+1|k+1)\right]_p \mathbf{u}_m + \mathbf{u}_m^T \mathbf{M}_{psd} \mathbf{u}_m \\
\Delta_m P_{ij}(k|k) &= \Delta_m P_{ij}(k+1|k+1) + \mathbf{u}_m^T \mathbf{M}_{psd} \mathbf{u}_m.
\end{aligned}$$

Since $\mathbf{u}_m^T \mathbf{M}_{psd} \mathbf{u}_m \geq 0$, it follows that

$$\Delta_m P_{ij}(k+1|k+1) \leq \Delta_m P_{ij}(k|k).$$

The variance in each dimension of the estimate of the relative distances vector between any pair of features is a monotonically non-increasing function of the time step.

The results derived for relative distances also apply to all other linear combinations, such as the mean position, of an arbitrary number of feature position estimates.

3.6 Interpretation of the Results

This section interprets the results obtained in Sections 3.3 – 3.5.2. The interpretation uses the determinant of the covariance matrix as the measure of the total uncertainty in the estimate. The

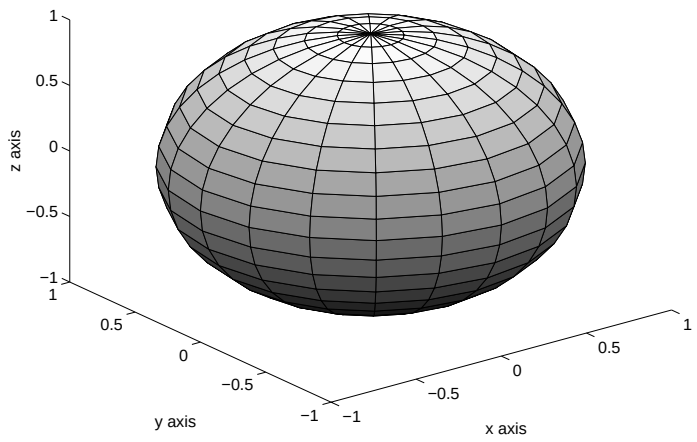

Figure 3.1: Confidence state space in three dimensions.

determinant is a convenient measure since it allows a scalar comparison between two multidimensional estimates. The use of the determinant can be justified by arguing that the covariance matrix of a multidimensional estimate represents a volume, which is a confidence state space similar to the scaler confidence interval represented by the variance of a scalar estimate. An example of a three dimensional confidence state space is shown in Figure 3.1.

It was shown in Section 3.5.1 that the determinant of the covariance matrix of any feature is monotonically non-increasing. Therefore, the confidence in any feature estimate cannot deteriorate. It was also shown in Section 3.5.1 that the variance in each dimension is monotonically non-increasing. Therefore, not only the confidence in the estimate, but also the confidence in each dimension of the estimate, cannot deteriorate for any feature. Since the map consists of the feature estimates, the results of Section 3.5.1 lead to the following interpretation.

The map constructed by the MAK filter will never become less certain.

The second important interpretation of the results regards the relative feature positions. The confidence in the relative feature positions is interpreted in this thesis as the rigidity of the map. The higher the confidence in the relative feature positions, the more rigid the map. This is justified by the fact that a high confidence in the relative position between two features p_i and p_j allows the position of feature p_i to be determined based upon the position of feature p_j with a confidence approaching the confidence in the position estimate of feature p_j. A high confidence in the relative position establishes a strong "link" and hence a high map rigidity.

Drawing on the results of Section 3.5.2 and again using the determinant as the measure of confidence, the confidence in the relative feature positions can be seen to be non-decreasing. The

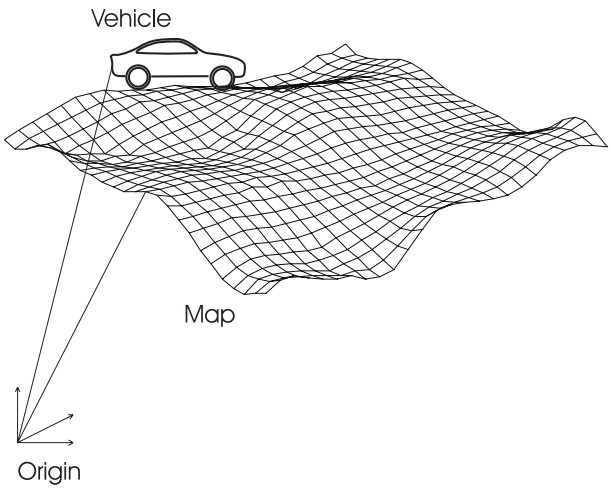

Figure 3.2: The relation between the vehicle and the map.

scalar variance, and hence the confidence, in each dimension of the relative distance was shown in Section 3.5.2 to be non-decreasing. These results lead to the following interpretation.

The rigidity of the map does not decrease.

The interpretations of certainty and rigidity are limited to the map. The covariance of the vehicle estimate and the covariance between the vehicle and the feature estimates are affected by the prediction stage as shown in Equation 3.5. Therefore, the determinant of a covariance matrix including the covariance of the vehicle is not monotonic. As a result, the confidence in the vehicle estimate and the confidence in the relative position between the vehicle and the map is not monotonic. This leads to the scenario shown in Figure 3.2, illustrating that the map evolves as a unit and the vehicle estimate as another unit.

3.7 Limit of the Map Accuracy

In the previous sections it was established that a map can be produced using the Kalman filter. It was shown that the certainty in the map, as measured by the determinant of the covariance of the feature estimates, does not deteriorate. This section examines theoretical limits to the certainty in the map produced. The investigation starts by introducing a notational convenience in Section 3.7.1. This is followed by an examination of the beginning of the map building process, the initialisation of the first feature. The structure of $\mathbf{P}_{mav}$ due to the initialisation is identified in Section 3.7.2. The structure of $\mathbf{P}_{mav}$ is then considered in an update using a relative observation, Section 3.7.3, and in

updates using terrain data and external vehicle position data, Section 3.7.4, leading to conclusions regarding the limits in the map certainty.

3.7.1 Equating the Vehicle and the Feature State Dimensions

The dimension of the vehicle state $\mathbf{x}_v$ is not necessarily the same as the dimension of the feature state $\mathbf{p}_i$. However, the subsequent analysis is simplified if the dimensions of $\mathbf{x}_v$ and $\mathbf{p}_i$ are the same. This is achieved by augmenting the lower dimensional state by "dummy" states. These states are defined as permanently zero for convenience, and the covariances associated with a "dummy" state estimate are also defined as zero.

For the vehicle and the feature states given by Equations 2.25 and 2.26,

$$\begin{aligned} \mathbf{x}_v(k) &= \begin{bmatrix} x_v(k) & y_v(k) & \phi_v(k) \end{bmatrix}^T \\ \mathbf{p}_i(k) &= \begin{bmatrix} x_i(k) & y_i(k) \end{bmatrix}^T, \end{aligned}$$

the augmentation leads to

$$\mathbf{x}_v(k) = \begin{bmatrix} x_v(k) & y_v(k) & \phi_v(k) \end{bmatrix}^T \tag{3.47}$$

$$\mathbf{p}_i(k) = \begin{bmatrix} x_i(k) & y_i(k) & \mathbf{0} \end{bmatrix}^T. \tag{3.48}$$

This augmentation is only a notational convenience and does not change the estimation process in any way. The map augmented vehicle state $\mathbf{x}_{mav}$ is formed as before, and the previous derivations and results still apply.

3.7.2 Initialisation of the First Feature

The covariance matrix $\mathbf{P}_{mav}(k|k)$ is assumed to decompose into two additive terms, $\mathbf{P}_0(k|k)$ and $\mathbf{P}_E$, both positive semidefinite,

$$\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k) + \mathbf{P}_E. \tag{3.49}$$

Further, $\mathbf{P}_E$ has a certain symmetry in the form

$$\mathbf{P}_E = \begin{bmatrix} \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \\ \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \end{bmatrix}.$$

The following paragraph describes why such a structure in $\mathbf{P}_{mav}(k|k)$ will arise.

When a feature is initialised, the current vehicle position estimate is added to the relative observation between the vehicle and the feature in order to form an absolute position estimate of the feature. If the vehicle position estimate has some uncertainty before the first feature estimate is initialised, the first feature estimate will inherit this position uncertainty. Similarly any features that are initialised before an update is made must have at least the initial vehicle position uncertainty as part of their overall position uncertainty. Therefore there will be a common position uncertainty that can be represented by an additive term common to the covariance matrix of the vehicle estimate and each of the feature estimates. The symmetry in $\mathbf{P}_E$ is therefore established, and the term $\mathcal{P}_\mathcal{E}$ arises due to the uncertainty that each feature inherits from the vehicle as it is initialised. $\mathbf{P}_E$ represents a minimum covariance in all the position estimates.

The structure of $\mathcal{P}_\mathcal{E}$ for the specific vehicle and feature states given by Equations 3.47 and 3.48 is

$$\mathcal{P}_\mathcal{E} = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & 0 \\ \sigma_{xy}^2 & \sigma_{yy}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{3.50}$$

where σ_{xx}^2, σ_{xy}^2, σ_{yy}^2 arise due to the initial uncertainty in the vehicle position estimate. Equation 3.50 contains zeros since the "dummy" variables have no common uncertainty with the vehicle orientation estimate.

3.7.3 Updating with Relative Observations

From Equations 2.29 and 3.6 it follows that a relative observation $\mathbf{z}_{\mathbf{x}_{mav},i}(k)$ of range $r_i(k)$ and bearing $\theta_i(k)$ made by the vehicle of a feature p_i can be written as

$$\begin{aligned}
\mathbf{z}_{\mathbf{x}_{mav},i}(k) &= \mathbf{h}_{\mathbf{x}_{mav},i}\left[\mathbf{x}_{mav}(k), i, \mathbf{w}(k), k\right] \\
&= \begin{bmatrix} r_i(k) + w_r(k) \\ \theta_i(k) + w_\theta(k) \end{bmatrix} \\
&= \begin{bmatrix} \sqrt{(x_i(k) - x_v(k))^2 + (y_i(k) - y_v(k))^2} + w_r(k) \\ \tan^{-1}\frac{y_i(k) - y_v(k)}{x_i(k) - x_v(k)} - \phi_v(k) + w_\theta(k) \end{bmatrix}.
\end{aligned}$$

The significant property of the relative observation is that a small positive change in x_v will have exactly the opposite effect on r_i and θ_i as a small positive change in x_i. Similarly, a small positive change in y_v will have exactly the opposite effect on r_i and θ_i as a small positive change in y_i. In general, a relative observation is such that a change of state in the observer has the opposite effect on the observation as a similar change of state in the feature. This can be summarised in terms of the partial derivatives,

$$\begin{aligned}
\frac{\partial r_i}{\partial x_v} &= -\frac{\partial r_i}{\partial x_i} & \frac{\partial \theta_i}{\partial x_v} &= -\frac{\partial \theta_i}{\partial x_i} \\
\frac{\partial r_i}{\partial y_v} &= -\frac{\partial r_i}{\partial y_i} & \frac{\partial \theta_i}{\partial y_v} &= -\frac{\partial \theta_i}{\partial y_i}.
\end{aligned}$$

The Jacobians of the relative observation with respect to the vehicle state, $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v}$, and the feature state, $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}$, are then derived as

$$\begin{aligned}
\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} &= \begin{bmatrix} \frac{\partial r_i}{\partial x_v} & \frac{\partial r_i}{\partial y_v} & \frac{\partial r_i}{\partial \phi_v} \\ \frac{\partial \theta_i}{\partial x_v} & \frac{\partial \theta_i}{\partial y_v} & \frac{\partial \theta_i}{\partial \phi_v} \end{bmatrix} \\
&= \begin{bmatrix} \frac{\partial r_i}{\partial x_v} & \frac{\partial r_i}{\partial y_v} & 0 \\ \frac{\partial \theta_i}{\partial x_v} & \frac{\partial \theta_i}{\partial y_v} & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \frac{\partial r_i}{\partial \phi_v} \\ 0 & 0 & \frac{\partial \theta_i}{\partial \phi_v} \end{bmatrix} \\
&= -\mathbf{H}_r + \mathbf{H}_{v0}
\end{aligned} \tag{3.51}$$

$$\begin{aligned}
\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} &= \begin{bmatrix} \frac{\partial r_i}{\partial x_i} & \frac{\partial r_i}{\partial y_i} & 0 \\ \frac{\partial \theta_i}{\partial x_i} & \frac{\partial \theta_i}{\partial y_i} & 0 \end{bmatrix} \\
&= \mathbf{H}_r,
\end{aligned} \tag{3.52}$$

where the zeros in Equation 3.52 reflect the augmented feature state. The term $\mathbf{H}_r$ appears in the two Jacobians with opposite sign due to the relative observation. The term $\mathbf{H}_{v0}$ arises due to

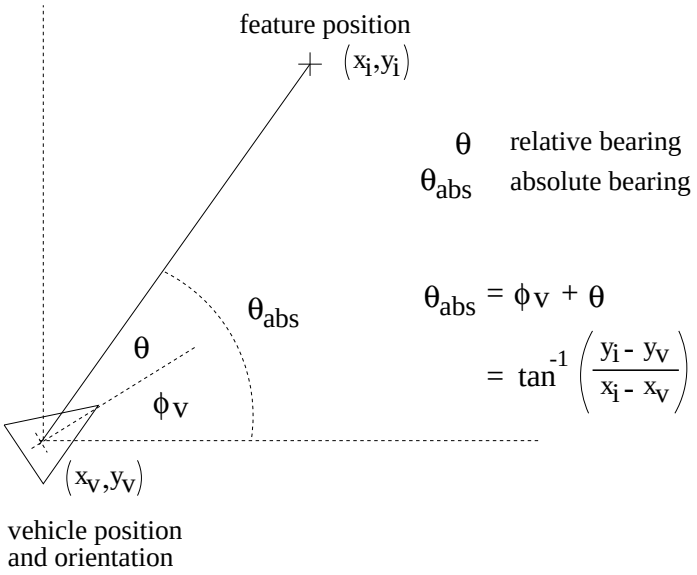

Figure 3.3: Relative observation of range and bearing.

the dependence of the relative observation on the orientation of the vehicle, and has the property $\mathcal{P}_{\mathcal{E}}\mathbf{H}_{v0} = \mathbf{0}$, due to the placement of zeros in $\mathcal{P}_{\mathcal{E}}$ and $\mathbf{H}_{v0}$. A term similar to $\mathbf{H}_{v0}$ could be introduced in the Jacobian $\nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i}$ if the relative observation depended on certain feature states that have no corresponding states in the vehicle.

From Equation 3.8 it follows that $\mathbf{H}_i(k)$ can be expressed in terms of the Jacobians just derived,

$$\begin{aligned}\mathbf{H}_i(k) &= \left[\begin{array}{ccccccc} \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{x}_v} & \mathbf{0} & \ldots & \mathbf{0} & \nabla\left(\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\right)_{\mathbf{p}_i} & 0 & \ldots \end{array}\right](k) \\ &= \left[\begin{array}{ccccccc} -\mathbf{H}_r(k) + \mathbf{H}_{v0}(k) & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{H}_r(k) & 0 & \ldots \end{array}\right], \end{aligned} \tag{3.53}$$

where the time index k outside the square brackets indicates that the Jacobians are taken at time step k. Although the Jacobians are different at each time step, the structure in Equation 3.53 is preserved.

Therefore, a certain symmetry exists in $\mathbf{H}_i(k)$, $\mathbf{H}_i(k+1), \ldots$ due to the nature of a relative observation. This symmetry combines with the symmetry in $\mathbf{P}_E$, established in Section 3.7.2, in an important way,

$$\begin{aligned}
\mathbf{P}_E\mathbf{H}_i^T(k) &= \begin{bmatrix} \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \\ \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{P}_\mathcal{E} & \mathcal{P}_\mathcal{E} & \dots & \mathcal{P}_\mathcal{E} \end{bmatrix} \begin{bmatrix} -\mathbf{H}_r^T(k) + \mathbf{H}_{v0}^T(k) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{H}_r^T(k) \\ \mathbf{0} \\ \vdots \end{bmatrix} \\
&= \mathbf{0},
\end{aligned}$$

and leads to a general conclusion regarding relative observations,

$$\mathbf{P}_E\mathbf{H}_i^T(k) = \mathbf{P}_E\mathbf{H}_i^T(k+1) = \mathbf{P}_E\mathbf{H}_i^T(k+2) = \cdots = \mathbf{0}.$$

The Kalman filter update in the case $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k) + \mathbf{P}_E$ will now be compared to the update in the case $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k)$. Thus, the effect of the additive term $\mathbf{P}_E$ on the filtering process is investigated.

Let $\mathbf{S}_0(k+1|k)$ and $\mathbf{W}_0(k+1)$ be the covariance of the innovation and the gain matrix respectively for the case $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k)$, and $\mathbf{S}_{0E}(k+1|k)$ and $\mathbf{W}_{0E}(k+1)$ be the covariance of the innovation and the gain matrix respectively for the case $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k) + \mathbf{P}_E$. If a relative observation of feature p_i is made the following expressions can be derived,

$$
\begin{aligned}
\mathbf{S}_{0E}(k+1|k) &= \mathbf{H}_i(k+1)\mathbf{P}_{0E}(k+1|k)\mathbf{H}_i^T(k+1) + \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} \mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}}^T \\
&= \mathbf{H}_i(k+1)\left(\mathbf{P}_0(k+1|k) + \mathbf{P}_E\right)\mathbf{H}_i^T(k+1) \\
&\quad +\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} \mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}}^T \\
&= \mathbf{H}_i(k+1)\mathbf{P}_0(k+1|k)\mathbf{H}_i^T(k+1) \\
&\quad +\mathbf{H}_i(k+1)\mathbf{P}_E\mathbf{H}_i^T(k+1) + \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} \mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}}^T \\
&= \mathbf{H}_i(k+1)\mathbf{P}_0(k+1|k)\mathbf{H}_i^T(k+1) + \nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}} \mathbf{R}(k+1)\nabla\left(\mathbf{h}_{\mathbf{x}_{mav},i}\right)_{\mathbf{w}}^T \\
\Rightarrow \mathbf{S}_{0E}(k+1|k) &= \mathbf{S}_0(k+1|k)
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{W}_{0E}(k+1) &= \mathbf{P}_{0E}(k+1|k)\mathbf{H}_i^T(k+1)\mathbf{S}_{0E}^{-1}(k+1|k) \\
&= \left(\mathbf{P}_0(k+1|k) + \mathbf{P}_E\right)\mathbf{H}_i^T(k+1)\mathbf{S}_0^{-1}(k+1|k) \\
&= \mathbf{P}_0(k+1|k)\mathbf{H}_i^T(k+1)\mathbf{S}_0^{-1}(k+1|k) + \mathbf{P}_E\mathbf{H}_i^T(k+1)\mathbf{S}_0^{-1}(k+1|k) \\
&= \mathbf{P}_0(k+1|k)\mathbf{H}_i(k+1)\mathbf{S}_0^{-1}(k+1|k) \\
\Rightarrow \mathbf{W}_{0E}(k+1) &= \mathbf{W}_0(k+1)
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{P}_{0E}(k+1|k) &= \mathbf{P}_{0E}(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}} \mathbf{Q}(k+1)\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}}^T \\
&= \left(\mathbf{P}_0(k|k) + \mathbf{P}_E\right) + \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}} \mathbf{Q}(k+1)\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}}^T \\
&= \left(\mathbf{P}_0(k|k) + \nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}} \mathbf{Q}(k+1)\nabla\left(\mathbf{f}_{\mathbf{x}_{mav}}\right)_{\mathbf{v}}^T\right) + \mathbf{P}_E \\
\Rightarrow \mathbf{P}_{0E}(k+1|k) &= \mathbf{P}_0(k+1|k) + \mathbf{P}_E
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{P}_{0E}(k+1|k+1) &= \mathbf{P}_{0E}(k+1|k) - \mathbf{W}_{0E}(k+1)\mathbf{S}_{0E}(k+1|k)\mathbf{W}_{0E}^T(k+1) \\
&= \mathbf{P}_{0E}(k+1|k) - \mathbf{W}_0(k+1)\mathbf{S}_0(k+1|k)\mathbf{W}_0^T(k+1) \\
&= \left(\mathbf{P}_0(k+1|k) + \mathbf{P}_E\right) - \mathbf{W}_0(k+1)\mathbf{S}_0(k+1|k)\mathbf{W}_0^T(k+1) \\
&= \left(\mathbf{P}_0(k+1|k) - \mathbf{W}_0(k+1)\mathbf{S}_0(k+1|k)\mathbf{W}_0^T(k+1)\right) + \mathbf{P}_E \\
\Rightarrow \mathbf{P}_{0E}(k+1|k+1) &= \mathbf{P}_0(k+1|k+1) + \mathbf{P}_E.
\end{aligned}
$$

These derivations show that if a term $\mathbf{P}_E$ can be identified as part of the covariance matrix at time step k, then the same term can be identified again at time step $k+1$. Further, the evolution of a covariance matrix that can be written as $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k) + \mathbf{P}_E$ can be seen in terms of the independent evolution of $\mathbf{P}_0(k|k)$ and $\mathbf{P}_E$. The latter has been identified as a constant, and the

former evolves as if the covariance matrix were $\mathbf{P}_{mav}(k|k) = \mathbf{P}_0(k|k)$. Therefore $\mathbf{P}_{mav}(k|k)$, which is the sum of $\mathbf{P}_0(k|k)$ and $\mathbf{P}_E$, has a lower bound given by $\mathbf{P}_E$.

$$\begin{aligned} \mathbf{P}_{mav}(k|k) &= \mathbf{P}_0(k|k) + \mathbf{P}_E \\ \Rightarrow \mathbf{P}_{mav}(k+1|k+1) &= \mathbf{P}_0(k+1|k+1) + \mathbf{P}_E \end{aligned}$$

The term $\mathbf{P}_E$ represents a lower limit in the covariance of the features, and hence a limit in the quality of the map, when relative observations are used.

Arguably, the feature positions will never be better known than the vehicle position was known initially. Features estimates initialised after an update has been performed will also inherit the uncertainty represented by $\mathbf{P}_E$ since it is not affected by an update and will therefore exist as part of the vehicle estimate covariance.

As the covariance matrix $\mathbf{P}_{mav}$ approaches $\mathbf{P}_E$, the correlations represented by $\mathbf{P}_E$ will dominate. This can be seen in terms of the correlation coefficients [18] r_i^{jl},

$$r_i^{jl} = \frac{\sigma_{j_i l_i}}{\sqrt{\sigma_{j_i j_i} \sigma_{l_i l_i}}}, \tag{3.54}$$

where $\sigma^2_{j_i j_i}$ is the variance of the i^{th} dimension of the estimate of feature number j, $\sigma^2_{l_i l_i}$ is the variance of the i^{th} dimension of the estimate of feature number l, $\sigma^2_{j_i l_i}$ is the covariance of the i^{th} dimension of the estimate of feature number j and the i^{th} dimension of the estimate of feature number l. It follows that

$$r_i^{jl} \to 1 \text{ as } \mathbf{P}_{mav} \to \mathbf{P}_E \text{ ; for } 1 \leq j \leq N, 1 \leq l \leq N,$$

for all dimensions i that have a common uncertainty represented in $\mathcal{P}_\mathcal{E}$.

As $\mathbf{P}_{mav}$ approaches the limit $\mathbf{P}_E$, the feature estimate errors become perfectly correlated.

The correlation coefficients of Equation 3.54 are also related to map rigidity, discussed in Section 3.6. This is demonstrated as follows. From Equation 3.43 the covariance $\Delta\mathbf{P}_{ij}^{0E}$ of a relative distance $\Delta\mathbf{p}_{ij}$ when $\mathbf{P}_{mav} = \mathbf{P}_0 + \mathbf{P}_E$ is

$$
\begin{aligned}
\Delta\mathbf{P}_{ij}^{0E} &= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ii}+\mathcal{P}_{\mathcal{E}} & \mathbf{P}_{ij}+\mathcal{P}_{\mathcal{E}} \\ \mathbf{P}_{ji}+\mathcal{P}_{\mathcal{E}} & \mathbf{P}_{jj}+\mathcal{P}_{\mathcal{E}} \end{bmatrix} \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix} \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} + \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathcal{P}_{\mathcal{E}} & \mathcal{P}_{\mathcal{E}} \\ \mathcal{P}_{\mathcal{E}} & \mathcal{P}_{\mathcal{E}} \end{bmatrix} \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ii} & \mathbf{P}_{ij} \\ \mathbf{P}_{ji} & \mathbf{P}_{jj} \end{bmatrix} \begin{bmatrix} -\mathbf{I} \\ \mathbf{I} \end{bmatrix} \\
&= \Delta\mathbf{P}_{ij}^{0},
\end{aligned}
$$

where $\Delta\mathbf{P}_{ij}^{0}$ is the covariance of the relative distance $\Delta\mathbf{p}_{ij}$ when $\mathbf{P}_{mav} = \mathbf{P}_0$. Therefore, the covariance of a relative distance is independent of $\mathbf{P}_E$. Consequently,

$$\Delta\mathbf{P}_{ij} \to \mathbf{0} \text{ as } \mathbf{P}_{mav} \to \mathbf{P}_E.$$

Since the map rigidity was interpreted in Section 3.6 as the confidence in the estimate of the relative position between the features in the map, and the determinant of a covariance matrix represents the confidence in the estimate, it follows that, as $\mathbf{P}_{mav}$ approaches the limit $\mathbf{P}_E$, the map becomes perfectly rigid.

> The limit in the map accuracy is represented by covariance term $\mathcal{P}_{\mathcal{E}}$, common to the vehicle and all feature estimates. It arises due to the initial uncertainty in the vehicle position estimate. As this limit is approached, all feature estimates become perfectly correlated, leading to a perfectly rigid map. Although the map is perfectly rigid, the registration of the map with the global coordinate reference frame is not known perfectly, and is limited by $\mathcal{P}_{\mathcal{E}}$.

In order to produce better quality maps than is possible with relative observations of the features, terrain data and vehicle position data can be incorporated. This is examined in the next section.

3.7.4 Incorporation of Terrain Data and Vehicle Position Data

When external terrain data or vehicle state data is incorporated, an independent observation of a feature state or an independent observation of the vehicle state is available. An example of external terrain data is a previously surveyed map of the environment, and an example of external vehicle state data is a GPS fix. In both cases there is a significant difference in the structure of the observation model as compared to a relative observation,

$$\begin{aligned}
\mathbf{z}_{vehicle}(k) &= \mathbf{h}_{vehicle}\left[\mathbf{x}_v(k), \mathbf{w}(k)\right] \\
\mathbf{z}_{terrain,i}(k) &= \mathbf{h}_{terrain}\left[\mathbf{p}_i(k), \mathbf{w}(k)\right].
\end{aligned}$$

Hence

$$\begin{aligned}
\nabla\left(\mathbf{h}_{vehicle}\right)_{\mathbf{x}_{mav}} &= \left[\begin{array}{cccc} \nabla\left(\mathbf{h}_{vehicle}\right)_{\mathbf{x}_v} & \mathbf{0} & \dots \end{array}\right] \\
&= \left[\begin{array}{ccc} \mathbf{H}_{veh} & \mathbf{0} & \dots \end{array}\right]
\end{aligned}$$

$$\begin{aligned}
\nabla\left(\mathbf{h}_{terrain}\right)_{\mathbf{x}_{mav}} &= \left[\begin{array}{ccccccc} \mathbf{0} & \dots & \mathbf{0} & \nabla\left(\mathbf{h}_{terrain}\right)_{\mathbf{p}_i} & \mathbf{0} & \dots \end{array}\right] \\
&= \left[\begin{array}{ccccccc} \mathbf{0} & \dots & \mathbf{0} & \mathbf{H}_{ter} & \mathbf{0} & \dots \end{array}\right],
\end{aligned}$$

where $\mathbf{H}_{veh}$ and $\mathbf{H}_{ter}$ have been introduced to simplify the notation.

The effect of incorporating external vehicle data is now investigated by considering an update.

$$\begin{aligned}
\mathbf{S}_{veh}(k+1|k) &= \nabla\left(\mathbf{h}_{vehicle}\right)_{\mathbf{x}_{mav}} \mathbf{P}_{mav}(k+1|k) \nabla\left(\mathbf{h}_{vehicle}\right)^T_{\mathbf{x}_{mav}} + \nabla\left(\mathbf{h}_{vehicle}\right)_{\mathbf{w}} \mathbf{R}(k+1) \nabla\left(\mathbf{h}_{vehicle}\right)^T_{\mathbf{w}} \\
&= \mathbf{H}_{veh}(k+1) \mathbf{P}_{vv}(k+1|k) \mathbf{H}^T_{veh}(k+1) + \nabla\left(\mathbf{h}_{vehicle}\right)_{\mathbf{w}} \mathbf{R}(k+1) \nabla\left(\mathbf{h}_{vehicle}\right)^T_{\mathbf{w}}
\end{aligned}$$

$$\begin{aligned}
\mathbf{W}_{veh}(k+1) &= \mathbf{P}_{mav}(k+1|k) \nabla\left(\mathbf{h}_{vehicle}\right)^T_{\mathbf{x}_{mav}} \mathbf{S}^{-1}_{veh}(k+1|k) \\
&= \left[\begin{array}{cccc} \mathbf{P}_{vv} & \mathbf{P}_{v1} & \dots & \mathbf{P}_{vN} \\ \mathbf{P}_{1v} & \mathbf{P}_{11} & \dots & \mathbf{P}_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv} & \mathbf{P}_{N1} & \dots & \mathbf{P}_{NN} \end{array}\right] (k+1|k) \left[\begin{array}{c} \mathbf{H}^T_{veh}(k+1) \\ \mathbf{0} \\ \vdots \end{array}\right] \mathbf{S}^{-1}_{veh}(k+1|k) \\
&= \left[\begin{array}{c} \mathbf{P}_{vv} \\ \mathbf{P}_{1v} \\ \vdots \\ \mathbf{P}_{Nv} \end{array}\right] \mathbf{H}^T_{veh}(k+1) \mathbf{S}^{-1}_{veh}(k+1|k)
\end{aligned}$$

$$\mathbf{W}_{veh}(k+1)\mathbf{S}_{veh}(k+1|k)\mathbf{W}_{veh}^{T}(k+1) = \begin{bmatrix} \mathbf{P}_{vv}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{vv}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{vv}\Lambda\mathbf{P}_{vN} \\ \mathbf{P}_{1v}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{1v}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{1v}\Lambda\mathbf{P}_{vN} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{Nv}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{Nv}\Lambda\mathbf{P}_{vN} \end{bmatrix}(k+1|k),$$

where $\Lambda(k+1|k)$ is $\mathbf{H}_{veh}^{T}(k+1)\mathbf{S}_{veh}^{-1}(k+1|k)\mathbf{H}_{veh}(k+1)$. Hence

$$\begin{aligned} \mathbf{P}_{mav}(k+1|k+1) &= \mathbf{P}_{mav}(k+1|k) - \mathbf{W}_{veh}(k+1)\mathbf{S}_{veh}(k+1|k)\mathbf{W}_{veh}^{T}(k+1) \\ &= \begin{bmatrix} \mathbf{P}_{vv} & \mathbf{P}_{v1} & \dots & \mathbf{P}_{vN} \\ \mathbf{P}_{1v} & \mathbf{P}_{11} & \dots & \mathbf{P}_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv} & \mathbf{P}_{N1} & \dots & \mathbf{P}_{NN} \end{bmatrix}(k+1|k) \\ &\quad - \begin{bmatrix} \mathbf{P}_{vv}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{vv}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{vv}\Lambda\mathbf{P}_{vN} \\ \mathbf{P}_{1v}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{1v}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{1v}\Lambda\mathbf{P}_{vN} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{Nv}\Lambda\mathbf{P}_{vv} & \mathbf{P}_{Nv}\Lambda\mathbf{P}_{v1} & \dots & \mathbf{P}_{Nv}\Lambda\mathbf{P}_{vN} \end{bmatrix}(k+1|k). \end{aligned}$$

The ideal vehicle observation is such that the each vehicle state is observed independently and the observation noise is zero. As the ideal vehicle observation is approached,

$$\begin{aligned} \mathbf{H}_{veh}(k+1) &= \mathbf{I} \\ \mathbf{R}(k+1) &\rightarrow \mathbf{0} \\ \Rightarrow \Lambda(k+1|k) &\rightarrow \mathbf{P}_{vv}^{-1}(k+1|k). \end{aligned}$$

Consequently,

$$\mathbf{P}_{mav}(k+1|k+1) = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{11} - \mathbf{P}_{1v}\mathbf{P}_{vv}^{-1}\mathbf{P}_{v1} & \dots & \mathbf{P}_{1N} - \mathbf{P}_{1v}\mathbf{P}_{vv}^{-1}\mathbf{P}_{vN} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{P}_{N1} - \mathbf{P}_{Nv}\mathbf{P}_{vv}^{-1}\mathbf{P}_{v1} & \dots & \mathbf{P}_{NN} - \mathbf{P}_{Nv}\mathbf{P}_{vv}^{-1}\mathbf{P}_{vN} \end{bmatrix}(k+1|k).$$

A similar argument for the incorporation of ideal terrain data of feature p_i leads to

$$\mathbf{P}_{mav}(k+1|k+1) = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \ddots \\ \dots & \mathbf{P}_{i-1i-1} - \mathbf{P}_{i-1i}\mathbf{P}_{ii}^{-1}\mathbf{P}_{ii-1} & \mathbf{0} & \mathbf{P}_{i-1i+1} - \mathbf{P}_{i-1i}\mathbf{P}_{ii}^{-1}\mathbf{P}_{ii+1} & \dots \\ \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots \\ \dots & \mathbf{P}_{i+1i-1} - \mathbf{P}_{i+1i}\mathbf{P}_{ii}^{-1}\mathbf{P}_{ii-1} & \mathbf{0} & \mathbf{P}_{i+1i+1} - \mathbf{P}_{i+1i}\mathbf{P}_{ii}^{-1}\mathbf{P}_{ii+1} & \dots \\ \ddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

where all terms on the right hand side are at time step $(k+1|k)$.

The updated covariance matrix after the inclusion of ideal vehicle or terrain data exhibits a structure very different from the structure exhibited after an update using relative observations, described in Section 3.7.3. The inclusion of vehicle or terrain data can potentially eliminate the common covariance term $\mathcal{P}_{\mathcal{E}}$ identified in Section 3.7.2. The elimination of $\mathcal{P}_{\mathcal{E}}$ would not follow from an ideal relative observation, since the results in Section 3.7.3 apply even if $\mathbf{H}_i = \begin{bmatrix} -\mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots \end{bmatrix}$ and $\mathbf{R} = \mathbf{0}$. The essential property of $\mathbf{h}_{vehicle}$ and $\mathbf{h}_{terrain}$ is that the observation is not relative. This is necessary in order to eliminate the common covariance term $\mathcal{P}_{\mathcal{E}}$. The inclusion of terrain or vehicle data can be seen as a decorrelation of the estimate errors, since the common covariance is reduced. However, since no assumptions about the observation model were made in the demonstration of non-decreasing map rigidity, given in Section 3.6, it follows that the inclusion of terrain of vehicle position data does not reduce the map rigidity.

The limit in the map accuracy associated with relative observations can be surpassed if terrain or vehicle data is incorporated. The inclusion of such data decorrelates the estimation errors, but does not decrease the map rigidity.

3.8 Summary

In this chapter the application of the Kalman filter to the SLAM problem was investigated. It was shown that consistency in the map and the vehicle position estimate is achieved only if a feature augmented vehicle state is used. Treating the SLAM problem of N features as a set of N independent filtering problems leads to filter inconsistencies due to neglected correlations, and hence results in an inconsistent map. It was further shown that an augmented state approach leads to feature position estimates with non-increasing covariances. This was interpreted as a non-deteriorating certainty in the map. Similarly it was shown that relative distance estimates have non-increasing covariances, and interpreted as a non-decreasing rigidity in the map.

The initial uncertainty in the vehicle position was identified as a limit to the accuracy of the map which cannot be surpassed by the using relative observations of the environment. Consequently it is also a limit to the accuracy of the localisation of the vehicle. External terrain or vehicle data was shown as the solution to obtain vehicle localisation and map accuracies beyond the limit set by the initial uncertainty of the vehicle position.

It was also concluded that the storage requirement of the MAK filter scales as N^2 and the computational burden scales by at least as much. The large storage requirement and heavy computational burden make the MAK filter impractical for real time implementation, making alternative solutions necessary for the SLAM problem. Alternative solutions are investigated in Chapter 4.

Chapter 4

Sub-Optimal Algorithms

4.1 Introduction

This chapter presents and examines two possible alternative sub-optimal algorithms for the solution of the Simultaneous Localisation and Map Building (SLAM) problem. These algorithms are aimed at providing solutions to the SLAM problem that do not suffer from the high memory and computation requirements associated with the Map Augmented Kalman (MAK) filter solution.

The two algorithms represent two different approaches to the problem, both with the view of eliminating the need to maintain a measure of correlations arising between estimate errors, and therefore eliminating the high memory and computation requirements.

The first approach, referred to as the Bounded Region (BOR) filter, modifies the assumptions of the Kalman filter by assuming that the process and the observation noise are bounded. This permits the BOR filter to compute bounded estimates, and has the advantage that the estimate updates do not depend on estimate error correlations. Consequently, it is not necessary to maintain a measure of such correlations, and a low memory and computational burden is achieved.

The second algorithm, referred to as the Relative (REL) filter, does not modify the process and the observation noise assumptions of the Kalman filter. However, it modifies the definition of the states to be estimated. Unlike the MAK filter, the REL filter does not estimate the absolute state of single features, but estimates relative states, such as relative distances and angles, between features. These states can be observed with no knowledge of the vehicle state. Hence the updates do not depend on the error correlations between different state estimates, again leading to a low memory and computational burden.

Section 4.2 examines the BOR filter, starting with an overview in Section 4.2.1. The application of the BOR filter to the SLAM problem is considered in Section 4.2.2 and the choice of a particular

type of BOR filter is made and justified in Section 4.2.3. A description of the data structure is given in Section 4.2.4. The BOR filter prediction stage is described in Section 4.2.5 and the update of the estimates are described in Sections 4.2.6 – 4.2.9. Section 4.2.10 outlines the gating and the feature initialisations and Section 4.2.11 summarises the operation of the BOR filter. Section 4.2.12 finishes with a short discussion of the filter.

Section 4.3 examines the REL filter, again starting with an overview in Section 4.3.1, followed by a description of the data structure in Section 4.3.2. In Section 4.3.3 the REL filter is presented for the case when the motion of the vehicle during a single scan of the sensor is insignificant. Sections 4.3.4 – 4.3.8 examine the case when the motion of the vehicle is significant and modify the filter to take the motion during scans into consideration. Section 4.3.9 summarises the operation of the REL filter and is followed by a short discussion of the filter in Section 4.3.10.

Section 4.4 summarises the chapter and makes some final remarks about the two filters examined.

4.2 The Bounded Region Filter

4.2.1 Overview

Unlike the Kalman filter, the BOR filter assumes that the estimates are contained within bounds of a known size. The filter operates by manipulating the perimeters of the bounding regions, and has the advantage that the manipulation does not require knowledge of correlations. The background to BOR filters, in particular their application to the SLAM problem, will now be briefly discussed.

State estimates and their uncertainties can be considered as valid regions of the state space [34, 63]. A valid region of the state space is a volume within which the true state of the system must lie. It is not known where in the valid region the true state lies, but it is known that the true state lies within the valid region. State estimates formulated as bounded valid regions will be referred to as bounded estimates, denoted $\hat{\mathbf{x}}|_b$.

Two bounded estimates of a state can be combined by intersecting the valid regions of the two estimates since the true state must simultaneously lie within both valid regions. This is the principle of bounded region filters, and is shown in Figure 4.1 for the two dimensional case. Correlations need not be considered when combining two bounded estimates, making BOR filters a potential solution to the SLAM problem [25]. This is a unique insight of the thesis.

Milanese [67] identified several drawbacks of statistical estimation theory that are not associated with BOR filters. In particular, its is argued that:

- Maximum-Likelihood Estimators can only be constructed for particular assumptions on the probability density function of the process and observation noise.

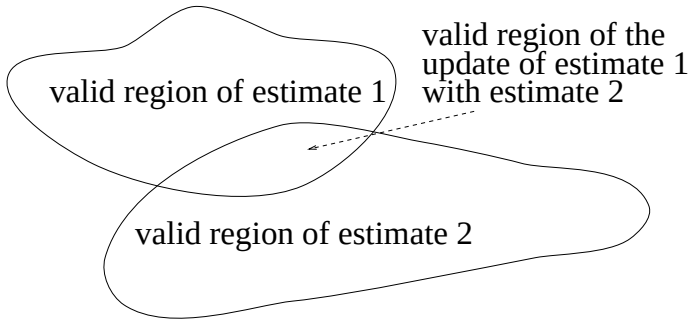

Figure 4.1: The bounded region estimate update.

- It is difficult to evaluate whether the number of independent noise sources is sufficiently large to produce an efficient estimate with respect to the Cramer-Rao Lower Bound [74].
- For the case where the number of independent noise sources is small, an upper bound on the covariance of the estimate would be useful. However, it is difficult to evaluate a tight bound.
- It is difficult to estimate the effects of errors in the assumed statistical description of the uncertainty.

Milanese saw the advantages of bounded region filters in cases where "a limited number of measurements and/or very inadequate information on the error statistic are available" [67]. Sabater in [81] argues along similar lines when stating that, "a probabilistic description of errors is not always available and only a bound on them is known ... set membership [bounded region] approach avoids general assumptions of unbiased and independent measurements taken by probabilistic approaches." More relevant to the SLAM problem is Sabater's following argument, "when the flow of incoming information and the number of constraints is large, there is a heavy computational overhead in the maintenance and updating of large covariance matrices."

Sabater identifies two main sorts of bounded region filters, one sort using polytopes and the other using ellipsoids to describe the valid regions.

Ellipsoids are used by Schweppe [82], in one of the first publications on the bounded region approach, due to the advantage of describing a region in terms of a finite number of parameters. The estimator was developed to track an evasive target whose pattern of evasive manoeuvres was not known, but the magnitude of the actions, such as accelerations, were limited. The lack of knowledge concerning the manoeuvres makes them a form of process noise, but the pattern in the manoeuvres does not allow an uncorrelated random process to be used as the noise model. Consequently, the application of statistical based estimators is difficult. However, the limits on the possible manoeuvres allows a bounded region estimator to be formulated.

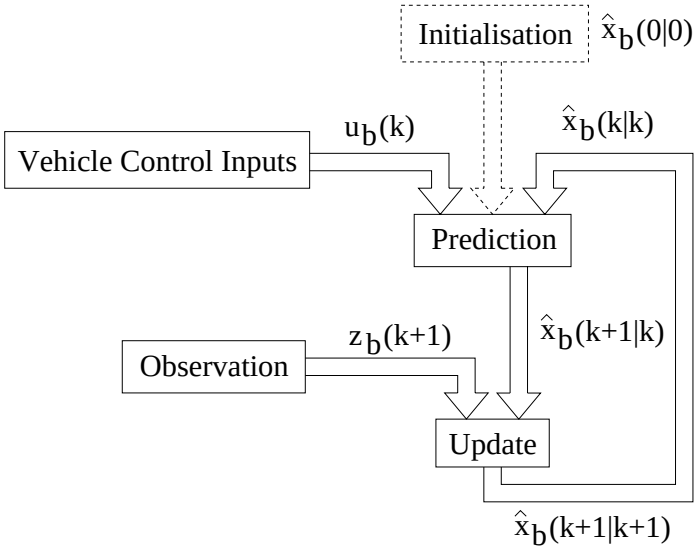

Figure 4.2: The estimation process in a BOR filter.

Fogel [31] also uses ellipsoids and concentrates on the problem of optimality. In particular, an algorithm is developed that computes the minimum volume ellipsoid. Bertsekas [7] uses matrices representing ellipsoids and develops the update as the convex combination of the matrices.

Mo [68] uses the polytope approach and develops algorithms to compute exact polytope bounds. Consequently, the solution is complex and computationally intensive. A simplification is made by Broman [16] by limiting the polytopes to a fixed dimension. A valid fixed dimensional polytope bound is derived by computing an outer approximation to the exact polytope bound. The bounded region estimate is therefore reduced in complexity, but also suffers a loss in information.

Bounded region estimates are published to this day in terms of the ellipsoid approach [63] and also the polytope approach [19].

The estimation process in a BOR filter can be broken down into a prediction stage and an update stage, similar to the Kalman filter, as shown in Figure 4.2 [1]. However, the prediction stage computes a bounded prediction from a bounded estimate, and the update stage computes a bounded updated estimate from the bounded prediction. An important difference between the Kalman and the BOR filter is that the BOR filter explicitly transforms the observation into state space before performing the update. This is necessary, since the update is be performed in state space. The estimation process in state space is illustrated in Figure 4.3.

[1] The subscript b on the control input $\mathbf{u}_b(k)$ and the observation $\mathbf{z}_b(k+1)$ indicates that certain assumptions, identified below, have been made about them.

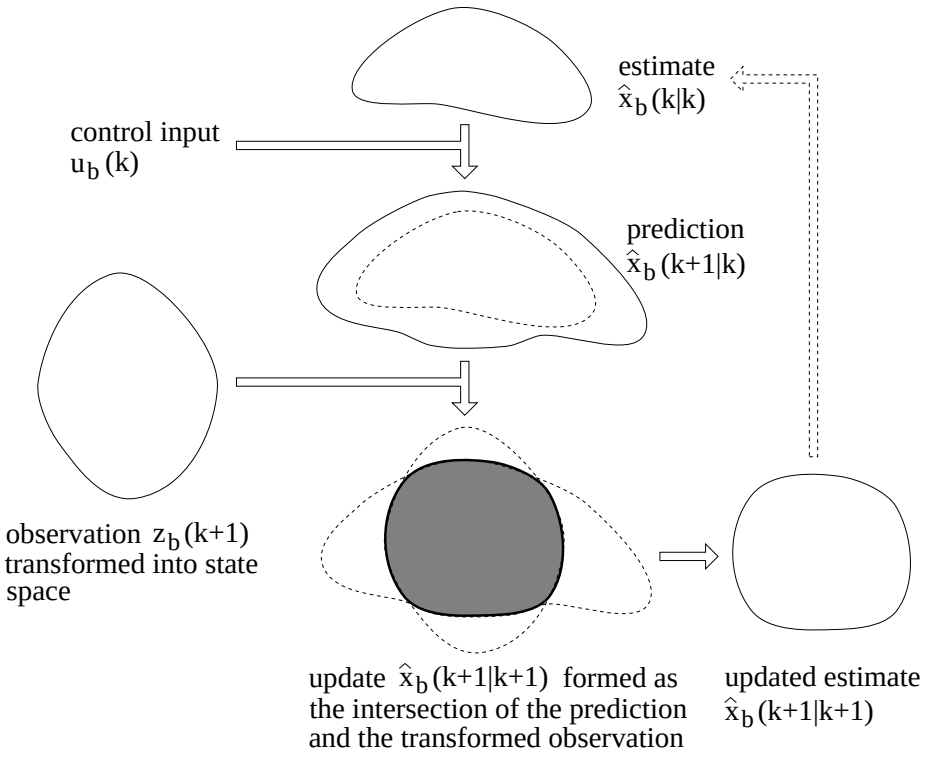

Figure 4.3: The estimation process in state space.

For a general non-linear system described by the process and observation models,

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}[\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k)] \\ \mathbf{z}(k) &= \mathbf{h}[\mathbf{x}(k), \mathbf{w}(k)], \end{aligned}$$

where, as in Chapter 2, $\mathbf{x}(k)$ is the state, $\mathbf{u}(k)$ the control input, $\mathbf{v}(k)$ and $\mathbf{w}(k)$ are the process and observation noise respectively, and $\mathbf{z}(k)$ is the observation made according to the observation model $\mathbf{h}$, the application of a BOR filter makes the following assumptions:

i. The process noise is bounded. This is necessary to compute a bounded prediction. The assumption is justified by considering the physical sources of the noise, such as rough terrain in the case of a vehicle model, and arguing that the noise sources are not expected to produce noise of infinite magnitude.

ii. The observation noise is bounded. This is necessary to compute a bounded updated estimate. The assumption is also justified by considering the physical sources of the noise, such as quantisation error in a sensor, and again arguing that the noise sources are not expected to produce noise of infinite magnitude.

iii. The observation can be transformed into state space. This is necessary to compute a bounded updated estimate. The assumption may be invalid in certain systems, making the application of a BOR filter considerably more difficult or even impossible. For example, it is not possible to transform a single position observation into a velocity state estimate, but several position observations may be combined into a velocity measurement. Therefore, though with some difficulty, the BOR filter could be applied to the estimation of the velocity.

iv. The control inputs are bounded. This is necessary to compute a bounded prediction. The assumption is justified by arguing that an infinite control input, such as an infinite velocity, requires infinite energy, which is clearly not possible.

v. The observations are bounded. This is necessary to compute a bounded updated estimate. The assumption is justified by considering the physics of a sensor; for example, measurement of an infinite range would require an infinite time, again an impossibility.

vi. An initial bounded estimate of the state is known. This is necessary to initialise the bounded state estimate, and can be clearly seen as a reasonable assumption in most applications.

According to these assumptions, the system is

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}[\mathbf{x}(k), \mathbf{u}(k), \mathbf{v}(k)] \\ \mathbf{z}(k) &= \mathbf{h}[\mathbf{x}(k), \mathbf{w}(k)], \end{aligned}$$

with the conditions,

$$\begin{aligned} |\mathbf{v}(k)|_{\mathbf{v}} &\preceq_v \mathbf{V}(k) \\ |\mathbf{w}(k)|_{\mathbf{w}} &\preceq_w \mathbf{W}(k) \\ |\mathbf{u}(k)|_{\mathbf{u}} &\preceq_u \mathbf{U}(k) \\ |\mathbf{x}(0)|_{\mathbf{x}} &\preceq_x \mathbf{X}, \end{aligned}$$

where $|\cdot|$ are some operators to be defined, $\preceq$ are some comparators to be defined, and $\mathbf{V}(k)$, $\mathbf{W}(k)$ and $\mathbf{U}(k)$ are the bounds on the process noise, the observation noise and the control input respectively. $\mathbf{X}$ is the initial bound on the state.

The literature also reports drawbacks of BOR filters [35]. The main drawbacks reported are:

- It is usually difficult to identify bounds on the process and observation noise accurately.

- Each operation performed by a BOR filter is very conservative, leading to very conservative, and therefore not very informative, estimates.

Although the combination of BOR and Bayesian approaches have been investigated [35, 36], this alternative will not be pursued in the solution to the SLAM problem, since the inclusion of Bayesian approaches results in the correlation problem examined in Chapter 3.

4.2.2 The BOR Filter and the SLAM Problem

In the SLAM problem,

$$\begin{aligned}
\mathbf{x}_v(k+1) &= \mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k)] \\
\mathbf{p}_i(k+1) &= \mathbf{p}_i(k);\ i = 1, 2, \ldots, N \\
\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}(k)];\ i = 1, 2, \ldots, N,
\end{aligned}$$

where, as in Chapter 2, $\mathbf{x}_v(k)$ is the state of the vehicle and $\mathbf{p}_i(k)$ is the state of feature p_i, and $\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k)$ is the observation of feature p_i made relative to the vehicle according to the observation model $\mathbf{h}_{\mathbf{x}_v,\mathbf{p}}$, application of a BOR filter makes the following additional assumption:

vii. The observation can be transformed into the state space of the vehicle *a*nd the observed feature. This is necessary to compute an updated bounded estimate of the vehicle and the observed feature. As in the general observation-to-state transformation assumption stated in iii, this assumption may be invalid, possibly making the application of the BOR filter impossible. However, the assumption can usually be justified by the "reversibility" of a relative observation. For example, in a range-bearing observation, the range of the feature from the vehicle is the same as the range of the vehicle from the feature, and the bearing in the global reference frame [2] of the feature from the vehicle is the same in modulus π as the bearing of the vehicle from the feature. Therefore, the observation of a feature relative to the vehicle can be interpreted as an observation of the vehicle relative to the feature [3], leading to the update of both the feature and the vehicle.

For the application of the BOR filter, the SLAM problem is expressed as

[2] The bearing in the global reference will be referred to as the "absolute" bearing to differentiate it from the "relative" bearing which is measured with respect to the vehicle orientation. Figure 4.9 illustrates the difference.

[3] The implied observation of the vehicle relative to the feature will be referred to as the "reverse observation."

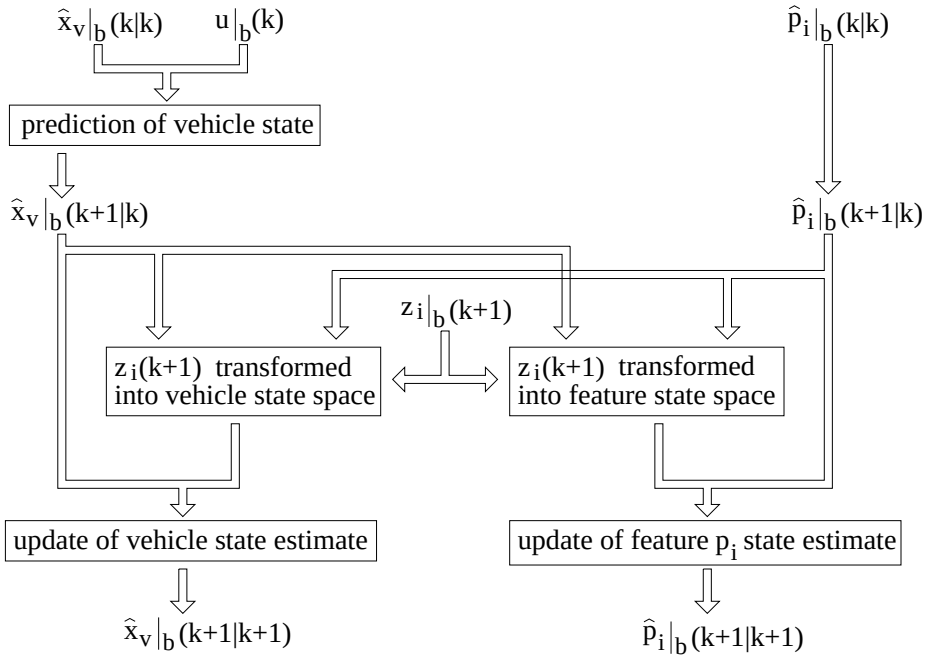

Figure 4.4: The BOR filter applied to the SLAM problem.

$$\begin{aligned}
\mathbf{x}_v(k+1) &= \mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k)] && (4.1)\\
\mathbf{p}_i(k+1) &= \mathbf{p}_i(k);\ i = 1, 2, \ldots, N \\
\mathbf{z}_{\mathbf{x}_v,\mathbf{p}_i}(k) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}(k)];\ i = 1, 2, \ldots, N, && (4.2)
\end{aligned}$$

with

$$\begin{aligned}
|\mathbf{v}(k)|_{\mathbf{v}} &\preceq_v \mathbf{V}(k) \\
|\mathbf{w}(k)|_{\mathbf{w}} &\preceq_w \mathbf{W}(k) \\
|\mathbf{u}(k)|_{\mathbf{u}} &\preceq_u \mathbf{U}(k) \\
|\mathbf{x}_v(0)|_{\mathbf{x}} &\preceq_x \mathbf{X}_v,
\end{aligned}$$

where $\mathbf{X}_v$ is the initial bound on the vehicle state.

The application of the BOR filter to the SLAM problem is as follows. The bounded vehicle state is predicted forwards using the bounds on the control input and the process noise. The bounded feature estimates are not affected in the prediction stage, since the features are assumed fixed in the

global coordinate frame, as stated in Section 2.3. The observation is transformed into feature and also into vehicle state space, so that both the vehicle and the feature state estimate can be updated, as shown in Figure 4.4. The filter then repeats recursively.

The transformations compute the "useful" information in the observation for the vehicle and the feature updates. A similar computation is performed in the Kalman filter. However, the Kalman filter identifies the "useful" information in terms of the covariance of the innovation and the estimate errors, as shown by Equation 2.34 for the vehicle estimate and Equation 2.35 for the feature estimate. In the Kalman filter, the covariances are highly interdependent, as identified in Section 2.4.3, and lead to a serious computational problem, as identified in Section 3.4. In the BOR filter, these interdependencies do not arise, since the transformation of the observation only involves the vehicle and the observed feature estimates. Therefore, for each feature in the map, a single bounded estimate is sufficient. Further, only the vehicle and the observed feature estimates are updated with each observation.

The storage requirement for the solution of the SLAM problem using the BOR filter increases only linearly with the number of features in the environment and the computational burden does not increase at all.

4.2.3 Implementation of the BOR Filter

To implement a BOR solution to the SLAM problem, a bounded estimate type must be chosen. As identified in Section 4.2.1, two main types exists: polytopes and ellipsoids. In this thesis, polytopes were chosen for the following reasons:

- The most simple bounded region update belongs to the polytope approach, in particular to a polytope that defines a bounded region in terms of an upper and a lower bound in each dimension. Such a polytope shall be referred to as a Hyperbound Polytope (HYPO). Updating a HYPO with another HYPO requires only two comparisons of scalar values for each dimension. This simplicity is not achievable using the ellipsoid approach.

- The region of intersection formed by HYPOs can be exactly represented by another HYPO, and eliminates the need to form an outer approximation to the intersection. This is not the case for ellipsoids, since "ellipsoids are not closed under intersection" [36]. However, the need to form outer approximations to HYPOs may arise at other stages in the filter.

- The detection of overlap between two HYPOs, which is needed to gate observations [4], is very simple. By comparison, the detection of ellipsoid overlap is tedious [36].

[4] Gating in the BOR filter is described in Section 4.2.10.

- The finite number of parameters associated with ellipsoids, motivating the choice of Schweppe [82], also applies to HYPOs.

A HYPO bounded estimate for a state $\mathbf{x}$ can be expressed as

$$\hat{\mathbf{x}}|_b \quad = \quad \{\mathbf{x} \mid \mathbf{x}_i^{min} \leq \mathbf{x}_i \leq \mathbf{x}_i^{max}\}$$

where $\mathbf{x}_i$ is the i^{th} dimension of $\mathbf{x}$ and $\mathbf{x}_i^{min}$ and $\mathbf{x}_i^{max}$ are, respectively, the minimum and maximum bounds in the i^{th} dimension.

The process noise bounds may also be defined independently in each dimension, leading to a simple manipulation of the HYPO estimate bounds during the prediction. This definition is justified by the fact that each dimension of the process noise usually models a different source, such as the noise on the velocity and the noise on the steer angle in the vehicle model presented in Section 2.3.2, and is assumed to be independent of the other noise sources. Similarly, the observation noise bounds may also be defined independently, leading to a simple manipulation of the observation bound during the transformation of the observation into state space. This definition is again justified by the fact that each dimension of the observation noise usually models a different source, such as the noise on the range and the noise on the bearing in the observation model presented in Section 2.3.2.

Therefore, $|\cdot|$ is defined as

$$|\mathbf{a}| \quad = \quad \{\mathbf{a}' \mid \mathbf{a}'_i = \|\mathbf{a}_i\| \; \forall \; i\},$$

where $\mathbf{a}_i$ is the i^{th} dimension of $\mathbf{a}$ and $\|a\|$ is the absolute value of a. The definition of $\preceq$ is

$$\begin{aligned} \mathbf{a} \quad &\preceq \quad \mathbf{b} \\ \Rightarrow \mathbf{a}_i \quad &\leq \quad \mathbf{b}_i \; \forall \; i, \end{aligned}$$

where $\mathbf{a}_i$ and $\mathbf{b}_i$ are the i^{th} dimensions of $\mathbf{a}$ and $\mathbf{b}$ respectively. Following these definitions, $\mathbf{V}(k)$, $\mathbf{W}(k)$, $\mathbf{U}(k)$, $\mathbf{X}$ define bounds in individual dimensions.

A general development of a HYPO-BOR filter solution to the SLAM problem is not very instructive, since the transformation of the observation into vehicle and feature state space is highly dependent on the particular vehicle and feature state representations and the particular observation model. Therefore, to give a more complete exposition, the development will be aimed at the specific

models presented in Section 2.3.2.

The BOR estimates for the vehicle and the feature states are defined in Section 4.2.4, followed by a description of the vehicle state prediction in Section 4.2.5. The first step towards the transformation of the observation into vehicle and feature states is taken in Section 4.2.6, where the bearing observation is transformed from relative to the vehicle orientation to absolute within the global reference frame. The complete transformation of the observation into feature state and the update of the feature estimate is described in Section 4.2.7. Similarly, the transformation into vehicle state and the update of the vehicle state is described in Section 4.2.8. The orientation of the vehicle is considered separately in Section 4.2.9.

4.2.4 The BOR State Estimates

The feature and the vehicle state were defined in Equations 2.26 and 2.25 as

$$\mathbf{p}_i(k) = \begin{bmatrix} x_i(k) & y_i(k) \end{bmatrix}^T \tag{4.3}$$

$$\mathbf{x}_v(k) = \begin{bmatrix} x_v(k) & y_v(k) & \phi_v(k) \end{bmatrix}^T. \tag{4.4}$$

A HYPO bounded state estimate $\hat{\mathbf{x}}_v|_b(k|k)$ for the vehicle and $\hat{\mathbf{p}}_i|_b(k|k)$ for the feature can be formulated respectively as follows,

$$\hat{\mathbf{p}}_i|_b(k|k) = \begin{bmatrix} \hat{x}_i|_{min}(k|k) \\ \hat{x}_i|_{max}(k|k) \\ \hat{y}_i|_{min}(k|k) \\ \hat{y}_i|_{max}(k|k) \end{bmatrix} \tag{4.5}$$

$$\hat{\mathbf{x}}_v|_b(k|k) = \begin{bmatrix} \hat{x}_v|_{min}(k|k) \\ \hat{x}_v|_{max}(k|k) \\ \hat{y}_v|_{min}(k|k) \\ \hat{y}_v|_{max}(k|k) \\ \hat{\phi}_v|_{right}(k|k) \\ \hat{\phi}_v|_{left}(k|k) \end{bmatrix}, \tag{4.6}$$

and

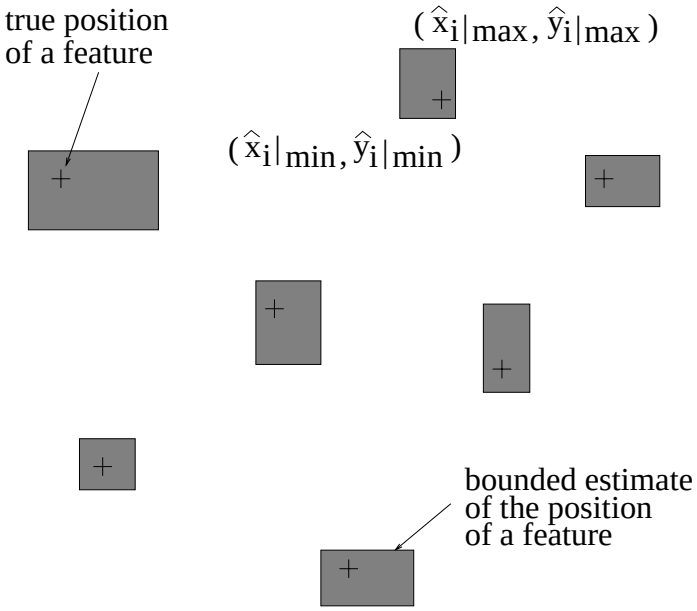

Figure 4.5: A bounded region map.

$$\begin{aligned}
\hat{x}_i|_{min}(k|k) &\leq \hat{x}_i|_{max}(k|k) \\
\hat{y}_i|_{min}(k|k) &\leq \hat{y}_i|_{max}(k|k) \\
\hat{x}_v|_{min}(k|k) &\leq \hat{x}_v|_{max}(k|k) \\
\hat{y}_v|_{min}(k|k) &\leq \hat{y}_v|_{max}(k|k).
\end{aligned}$$

These definitions lead to rectangular bounded regions for the position estimate defined by minimum and maximum bounds on each dimension. The orientation estimate specifies 'left' and 'right' bounds on the true orientation of the vehicle, since maximum and minimum bounds are not well defined for bearings. Using 'left' and 'right' bounds is in the spirit of HYPO bounded estimates. Starting with the direction of the 'right' bound and rotating in the anti-clockwise (positive) sense, it must be true that the true orientation of the vehicle will be reached before or as the direction of the 'left' bound is reached. The definition of the 'left' and the 'right' bound of the vehicle orientation estimate will also be used to define the 'left' and 'right' bounds of angle estimates in general.

The BOR map thus consists of a bounded state estimate of each feature in the map, as shown in Figure 4.5 and defined by Equation 4.5. The BOR vehicle estimate, defined by Equation 4.6, is shown in Figure 4.6.

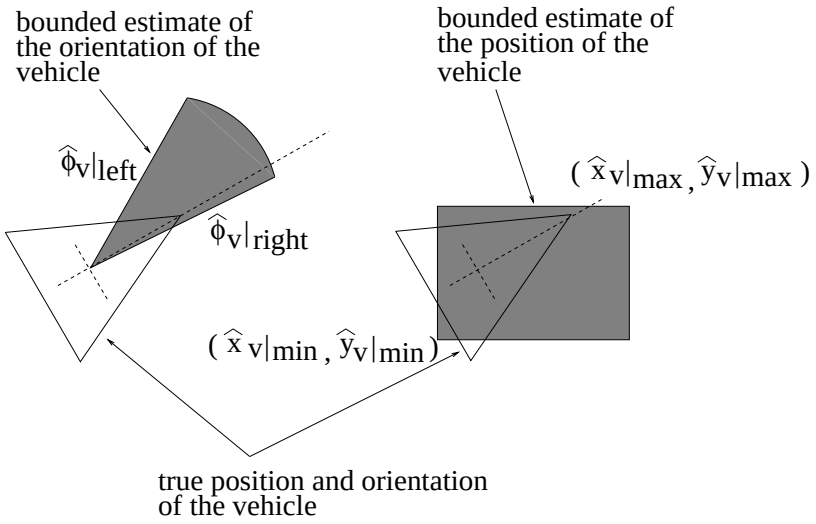

Figure 4.6: A bounded vehicle state estimate.

4.2.5 Prediction of the Vehicle State

The predicted vehicle state estimate is a valid region estimate which encloses the actual position of the vehicle. For the prediction, Equation 4.1 is combined with the rectangular valid region defined by Equation 4.6,

$$\begin{aligned}
\hat{\mathbf{x}}_v|_b(k+1|k) &= \begin{bmatrix} \hat{x}_v|_{min}(k+1|k) \\ \hat{x}_v|_{max}(k+1|k) \\ \hat{y}_v|_{min}(k+1|k) \\ \hat{y}_v|_{max}(k+1|k) \\ \hat{\phi}_v|_{right}(k+1|k) \\ \hat{\phi}_v|_{left}(k+1|k) \end{bmatrix} \\
&= \begin{bmatrix} \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_x\right) \\ \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_x\right) \\ \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_y\right) \\ \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_y\right) \\ \text{right}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_\phi\right) \\ \text{left}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left([\mathbf{f}_{\mathbf{x}_v}[\mathbf{x}_v,\mathbf{u}(k),\mathbf{v}]]_\phi\right) \end{bmatrix}.
\end{aligned} \tag{4.7}$$

where the max and min operators are taken over the bounds of $\mathbf{x}_v$ and $\mathbf{v}$ given by $\hat{\mathbf{x}}_v|_b(k|k)$ and $\mathbf{V}(k)$ respectively. The operators 'right' and 'left' are such that all possible orientations of the vehicle that can be achieved by taking $\mathbf{x}_v$ and $\mathbf{v}$ over the bounds $\hat{\mathbf{x}}_v|_b(k|k)$ and $\mathbf{V}(k)$ are captured

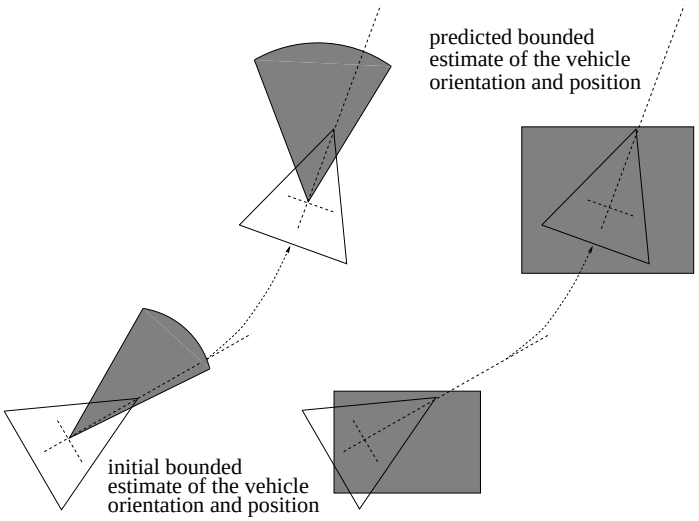

Figure 4.7: Forming the prediction of the vehicle state.

by the orientation bounds $\hat{\phi}_v|_{right}(k+1|k)$ and $\hat{\phi}_v|_{left}(k+1|k)$. The prediction process is shown in Figure 4.7.

The vehicle kinematics, given by Equation 2.27, are

$$\begin{aligned}
\mathbf{x}_v(k+1) &= \mathbf{f}_{\mathbf{x}_v}\left[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k)\right] \\
&= \begin{bmatrix} x_v(k) T\left[V(k) + v_v(k)\right] \cos(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ y_v(k) T\left[V(k) + v_v(k)\right] \sin(\phi_v(k) + [\gamma(k) + v_\gamma(k)]) \\ \phi_v(k) \frac{T[V(k)+v_v(k)]}{B} \sin(\gamma(k) + v_\gamma(k)) \end{bmatrix},
\end{aligned}$$

Combined with Equation 4.7, the predicted valid region is

$$\hat{\mathbf{x}}_v|_b(k+1|k) = \begin{bmatrix} \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(x_v + T\left[V(k)+v_v\right]\cos(\phi_v+\gamma(k)+v_\gamma)\right) \\ \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(x_v + T\left[V(k)+v_v\right]\cos(\phi_v+\gamma(k)+v_\gamma)\right) \\ \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(y_v + T\left[V(k)+v_v\right]\sin(\phi_v+\gamma(k)+v_\gamma)\right) \\ \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(y_v + T\left[V(k)+v_v\right]\sin(\phi_v+\gamma(k)+v_\gamma)\right) \\ \text{right}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(\phi_v + \frac{T[V(k)+v_v]}{B}\sin(\gamma(k)+v_\gamma)\right) \\ \text{left}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(\phi_v + \frac{T[V(k)+v_v]}{B}\sin(\gamma(k)+v_\gamma)\right) \end{bmatrix}$$

$$= \begin{bmatrix} \hat{x}_v|_{min}(k|k) + \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(T\left[V(k)+v_v\right]\cos(\phi_v+\gamma(k)+v_\gamma)\right) \\ \hat{x}_v|_{max}(k|k) + \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(T\left[V(k)+v_v\right]\cos(\phi_v+\gamma(k)+v_\gamma)\right) \\ \hat{y}_v|_{min}(k|k) + \min\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(T\left[V(k)+v_v\right]\sin(\phi_v+\gamma(k)+v_\gamma)\right) \\ \hat{y}_v|_{max}(k|k) + \max\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(T\left[V(k)+v_v\right]\sin(\phi_v+\gamma(k)+v_\gamma)\right) \\ \text{right}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(\phi_v + \frac{T[V(k)+v_v]}{B}\sin(\gamma(k)+v_\gamma)\right) \\ \text{left}\limits_{\hat{\mathbf{x}}_v|_b(k|k),\mathbf{V}(k)} \left(\phi_v + \frac{T[V(k)+v_v]}{B}\sin(\gamma(k)+v_\gamma)\right) \end{bmatrix}.$$

The computation of $\hat{\mathbf{x}}_v|_b(k+1|k)$ must take into account the non-linear trigonometric functions $\sin(\cdot)$ and $\cos(\cdot)$. For example, given an angle α, the maximum of $\sin(\alpha)$ for α with 'left' bound α_{left} and 'right' bound α_{right} will not necessarily occur on the bounds. To obtain the true maximum of $\sin(\alpha)$ it must be examined whether $\frac{\pi}{2}$ lies between α_{left} and α_{right}. If this is the case, the maximum is 1 regardless of the actual values of α_{left} and α_{right}. Otherwise, the maximum occurs on one of the bounds and is the larger of $\sin(\alpha_{left})$ and $\sin(\alpha_{right})$. The extremes of $\sin(\alpha)$ and $\cos(\alpha)$ for 'left' bound α_{left} and 'right' bound α_{right} are summarised below.

$\max(\sin(\alpha))$	=	1	$\frac{\pi}{2}$ within $\alpha_{left}, \alpha_{right}$
$\max(\sin(\alpha))$	=	$\max(\sin(\alpha_{left}), \sin(\alpha_{right}))$	otherwise
$\min(\sin(\alpha))$	=	-1	$\frac{-\pi}{2}$ within $\alpha_{left}, \alpha_{right}$
$\min(\sin(\alpha))$	=	$\min(\sin(\alpha_{left}), \sin(\alpha_{right}))$	otherwise
$\max(\cos(\alpha))$	=	1	0 within $\alpha_{left}, \alpha_{right}$
$\max(\cos(\alpha))$	=	$\max(\cos(\alpha_{left}), \cos(\alpha_{right}))$	otherwise
$\min(\cos(\alpha))$	=	-1	π within $\alpha_{left}, \alpha_{right}$
$\min(\cos(\alpha))$	=	$\min(\cos(\alpha_{left}), \cos(\alpha_{right}))$	otherwise

4.2.6 Bounded Observations

Like the Kalman filter, the BOR filter uses observations to update the prediction of the vehicle state and to estimate the position of the features. The update is performed as the intersection of two valid

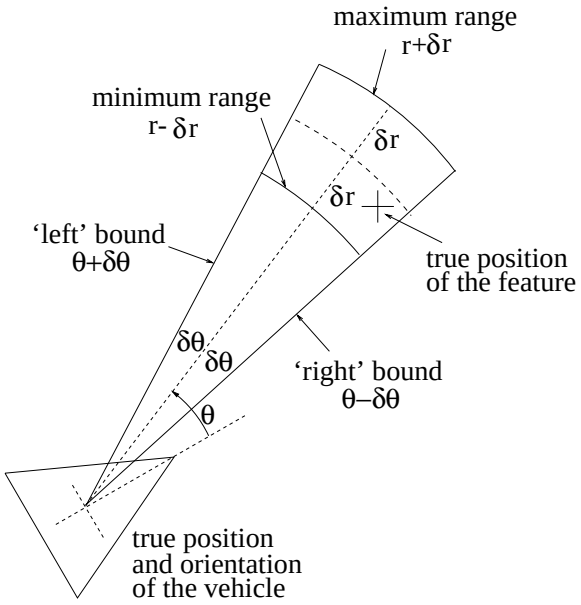

Figure 4.8: A bounded observation.

regions in the state space, as described in Section 4.2.1. One of the valid regions in a BOR filter update is the current estimate of the state, the second valid region is obtained from the observation.

For the range-bearing sensor described by Equation 2.29, the observation model described by Equation 4.2 leads to a bounded range and bearing observation as shown in Figure 4.8. The observation is made relative to the vehicle: the bearing observation is made relative to the vehicle orientation and the range observation is made relative to the vehicle position.

To formulate a bounded region for the observed feature, the relative observation has to be converted into an absolute observation of the feature position, hence the observation has to be transformed into feature state space. The first step in forming an absolute observation of the position of the feature is to find the absolute bearing of the observation, shown in Figure 4.9. The absolute bearing will be given by bounds since the relative bearing and the orientation of the vehicle are both given by bounds. A method to find the bounds on the absolute bearing of an observation is now described.

Assume that the bound on the vehicle orientation is given by $\phi + \delta\phi$ ('left' bound) and $\phi - \delta\phi$ ('right' bound), where $\delta\phi$ is the *s*pread of the bound [5]. Similarly, assume that the bound on the bearing observation with respect to the vehicle orientation is given by $\theta + \delta\theta$ ('left' bound) and $\theta - \delta\theta$ ('right' bound) [6]. The absolute bearing observation θ_{abs} and its bounds $\theta_{abs}|_{left} = \theta_{abs} + \delta\theta_{abs}$ and $\theta_{abs}|_{right} = \theta_{abs} - \delta\theta_{abs}$ are

[5] The bound on the vehicle orientation is given by the current vehicle estimate.
[6] The spread of the observation is given by the bound $\mathbf{W}$ on the observation noise.

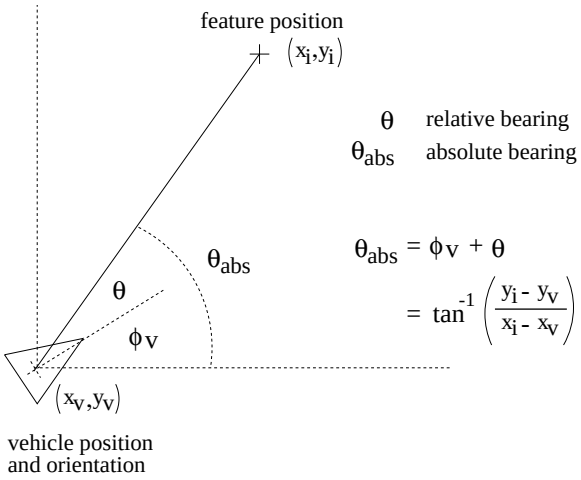

Figure 4.9: Relative bearing versus absolute bearing.

$$
\begin{aligned}
\theta_{abs} &= \theta + \phi \\
\\
\theta_{abs}|_{left} &= \theta_{abs} + \delta\theta_{abs} \\
&= \theta + \phi + (\delta\theta + \delta\phi) \\
\\
\theta_{abs}|_{right} &= \theta_{abs} - \delta\theta_{abs} \\
&= \theta + \phi - (\delta\theta + \delta\phi) \\
\\
\delta\phi, \delta\theta &\geq 0.
\end{aligned}
$$

An absolute bearing observation provides bounds between which the bearing of the true position of the feature with respect to the true position of the vehicle must lie.

The absolute bearing observation also provides bounds between which the bearing of the true position of the vehicle with respect to the true position of the observed feature must lie. To see this consider a feature observed by the vehicle. Assume there is a ray that connects the true position of the vehicle to the true position of the feature. Define the bearing α of this ray as $-\pi < \alpha \leq \pi$ and observe that it must be within the 'left' and 'right' bounds of the absolute bearing observation. The true absolute bearing of the reverse observation (the bearing of the true vehicle position with respect to the true feature position) can be represented by a second ray with a bearing of $\alpha + \pi$. It can now be seen that rotating the 'right' bound through π will produce a valid 'right' bound for the reverse absolute bearing observation. Similarly, rotating the 'left' bound through π will produce a valid 'left' bound for the reverse absolute bearing observation, as shown in Figure 4.10.

The maximum range and the minimum range bounds of an observation will also be the maximum

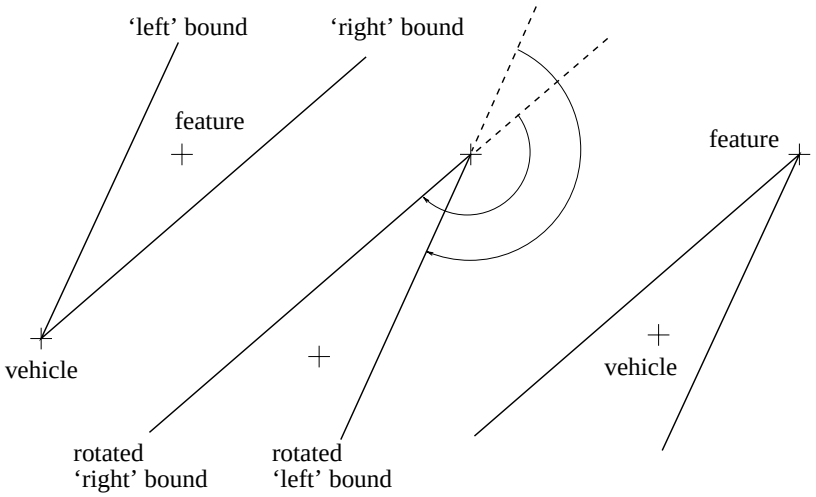

Figure 4.10: Forming the reversed observation.

range and the minimum range bounds of the reverse observation respectively.

Thus an observation can be used to construct bounds on the bearing of the true position of the feature with respect to the true position of the vehicle. Combining these bounds with the bounds on the range of the observation and on the position of the vehicle, it is possible to construct bounds on the position of the observed feature, completing the transformation of the observation into feature state space, and to use the constructed bounds to update the current position estimate of the feature. This is examined in Section 4.2.7.

4.2.7 Position Update of a Feature

Given the (unknown) true position of the vehicle and the bounds on the range and the absolute bearing of the observation, a valid region for the position of the observed feature can be formulated. The true position of the feature must lie within the valid region. In order to simplify the subsequent update, the valid region is formulated in terms of four straight edges, as shown in Figure 4.11. Each straight edge has a valid side and an invalid side. The area that falls on the valid side of all four edges is the valid region. This formulation in terms of straight edges results in a slight increase of the valid region. Thus some information is lost since the valid region will enclose an area that is certain not to contain the true position of the feature. However, this information loss is justified by being both conservative and providing a simple update rule. A further advantage of this approach is that it can incorporate degenerate observations, observations that are not bounded in a certain direction. Such observations have been identified in [81].

Construction of each of the four edges requires the unknown position of the vehicle, which will be referred to as the reference point. The 'left' and 'right' bounds on the absolute bearing pass

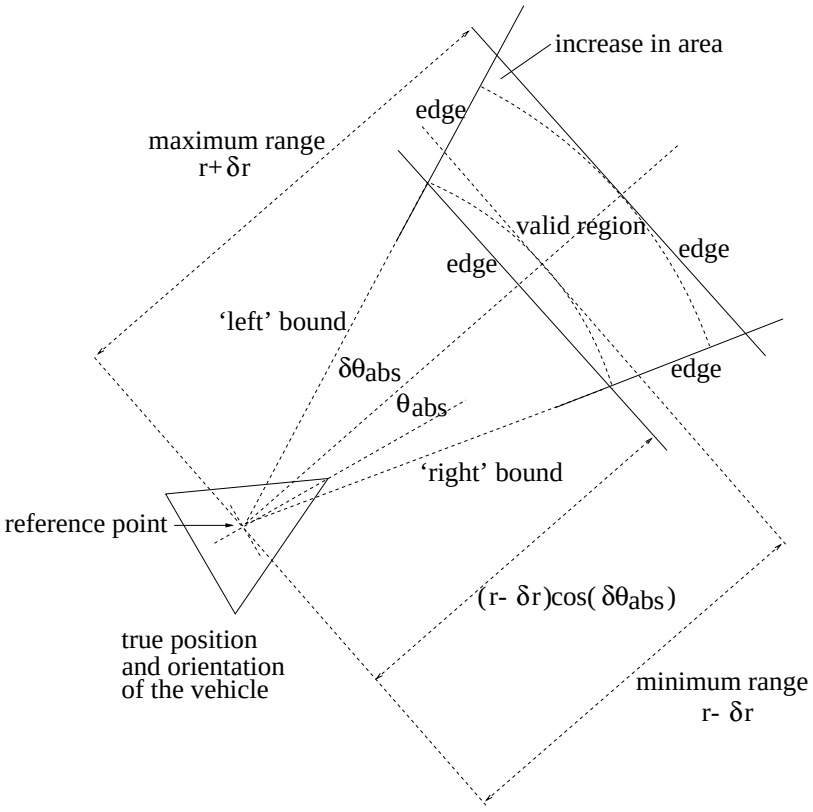

Figure 4.11: Forming a valid region from an observation.

through the reference point. The edges derived from the bounds on the range of the observation are perpendicular to the absolute bearing of the observation. The far bound corresponds to the maximum range and the perpendicular distance to the reference point is $r + \delta r$, which is the upper bound on the range. The close bound corresponds to the minimum range and the perpendicular distance to the reference point is $(r - \delta r)\cos(\delta\phi_{abs})$, due to the curvature, as shown in Figure 4.11.

If a bounded estimate of the position of the vehicle is available, the construction of the valid region for the position of the observed feature must account for the uncertainty in the position of the vehicle. This is achieved by taking the union of all the valid regions that correspond to all possible vehicle positions.

The union of valid regions can be obtained by using different reference points for the construction of each of the edges so that the valid region enclosed by the edges is maximised, as shown in Figure 4.12. To see this, consider the case where the valid region is not maximised. It is then possible to find a vehicle position that satisfies the bounds on the vehicle position and a feature position that satisfies the bounds on the observation given this vehicle position, but which does not fall into the valid region. The maximisation of the valid region can be performed independently with

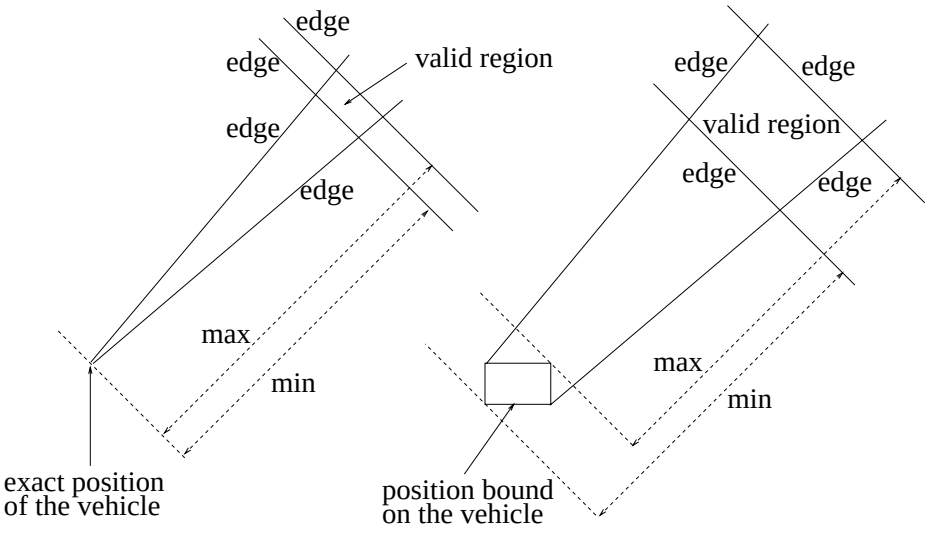

Figure 4.12: The effect of vehicle position uncertainty.

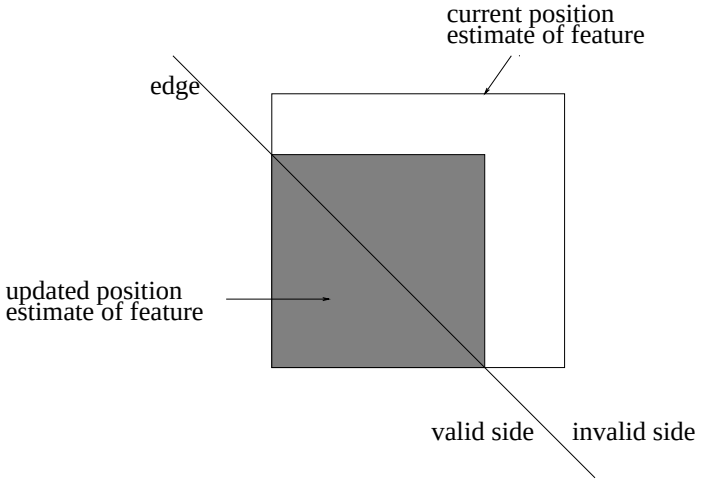

Figure 4.13: Update of the feature position estimate.

each of the four edges.

After an edge has been identified, it is used to partition the current feature position estimate into valid and a invalid regions. To conserve the structure of the feature estimate, the valid region is approximated by the smallest rectangular region, with sides parallel to the sides of the current feature position estimate, that completely encloses the valid region. The new rectangular region is the updated bounded position estimate of the feature, as shown Figure 4.13. This procedure is repeated for all the four edges used to formulate the valid region.

The position update of the vehicle estimate is similar to the position update of the feature estimate just examined.

4.2.8 Position Update of the Vehicle

As explained in Section 4.2.6, the observation of the feature relative to the vehicle and knowledge of the feature position are sufficient to constrain the vehicle position. The 'left' and 'right' bounds on the bearing of the position of the feature with respect to the position of the vehicle can be rotated by π radians to give 'left' and 'right' bounds on the bearing of the position of the vehicle with respect to the position of the observed feature, as shown in Figure 4.10. The rotated bounds are then combined with the current feature position estimate to give a valid region for the position of the vehicle. The valid region is again expressed in terms of edges which are then used to partition the current estimate of the vehicle position to form the update.

In addition to updating the position estimate of the vehicle, it is also necessary to update the orientation estimate of the vehicle. This is examined in Section 4.2.9.

4.2.9 Orientation Update of the Vehicle

The orientation update is achieved by formulating a bound on the vehicle orientation from the observation and intersecting that with the current orientation bound, as shown in Figure 4.14. To formulate a bound on the vehicle orientation from the observation, the current position estimate of the observed feature and the vehicle are considered. The position bounds define limits on the possible absolute bearing of the observation, as shown in Figure 4.15. The bounds on the absolute bearing are found by considering two rays and the bearing of the centroid of the feature position bound with respect to the centroid of the vehicle position bound. Both rays have their origin within the position bound of the vehicle and end within the position bound of the feature. The ends of the first ray are fixed such that the change in bearing needed to rotate in the positive sense from the bearing of the centroids onto the bearing β_l of the first ray is maximised. The ends of the second ray are fixed such that the change in bearing needed to rotate in the negative sense from the bearing of the centroids onto the bearing β_r of the second ray is maximised, as shown in Figure 4.15. The absolute bearing of the true feature position with respect to the true vehicle position must be within the bounds defined by β_l, the 'left' bound, and β_r, the 'right' bound. If the position bounds of the feature and the vehicle overlap, β_l and β_r cover all possible bearings and therefore no orientation update is possible.

The calculated bounds are rotated in the negative sense by the relative bearing of the actual observation given by the sensor. This gives a bound on the vehicle orientation for a perfect sensor. For an imperfect sensor, the rotated bounds have to be increased by the spread $\delta\theta$ of the bounds on the bearing of the observation, as shown in Figure 4.16. The rotated and increased orientation bounds, ϕ'_l, the 'left' bound, and ϕ'_r, the 'right' bound, can now be intersected with the current

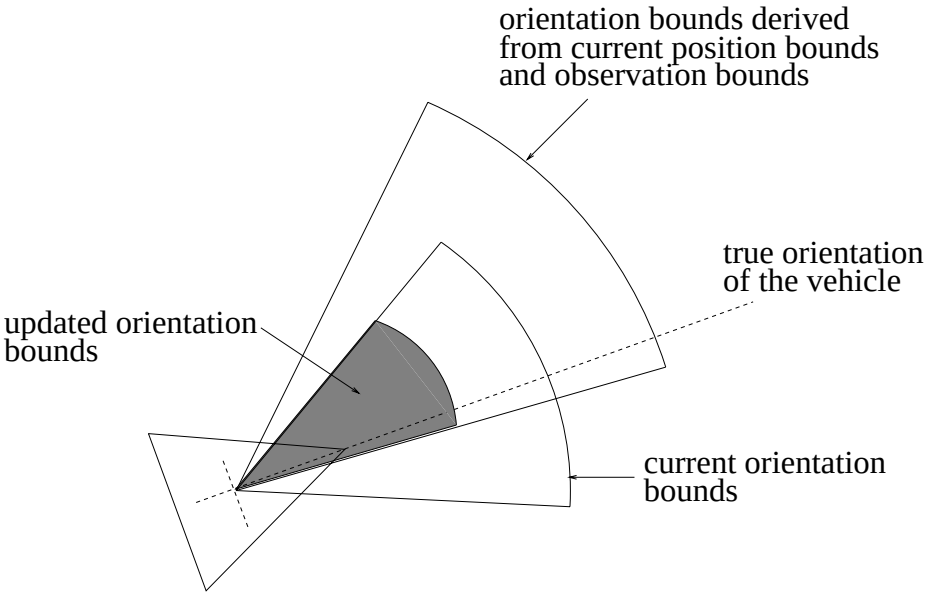

Figure 4.14: Updating the bound on the orientation of the vehicle.

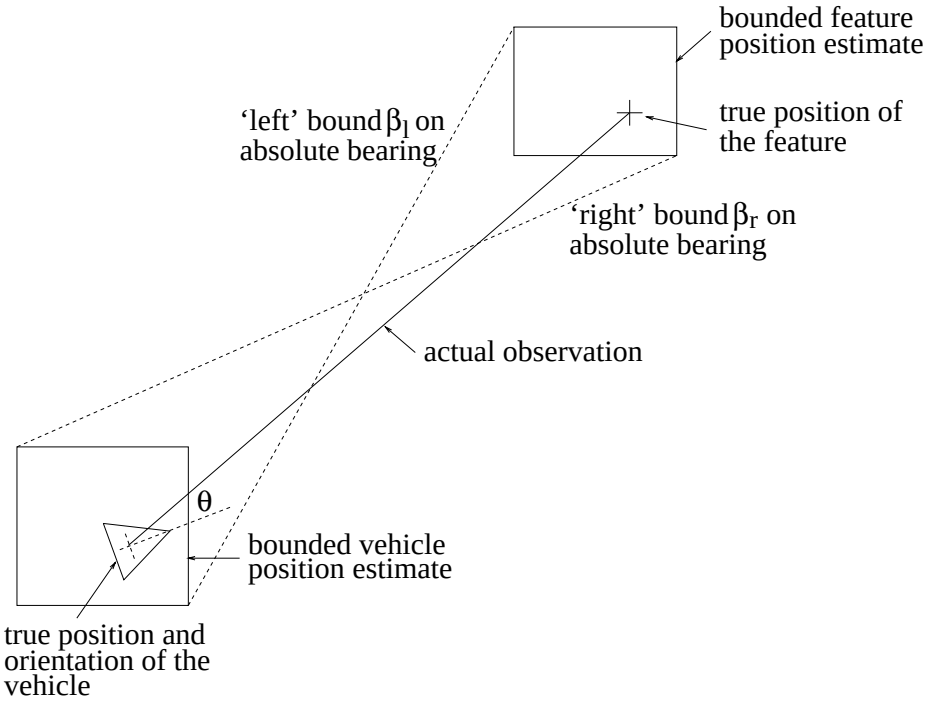

Figure 4.15: Forming the bounds on the possible absolute bearing.

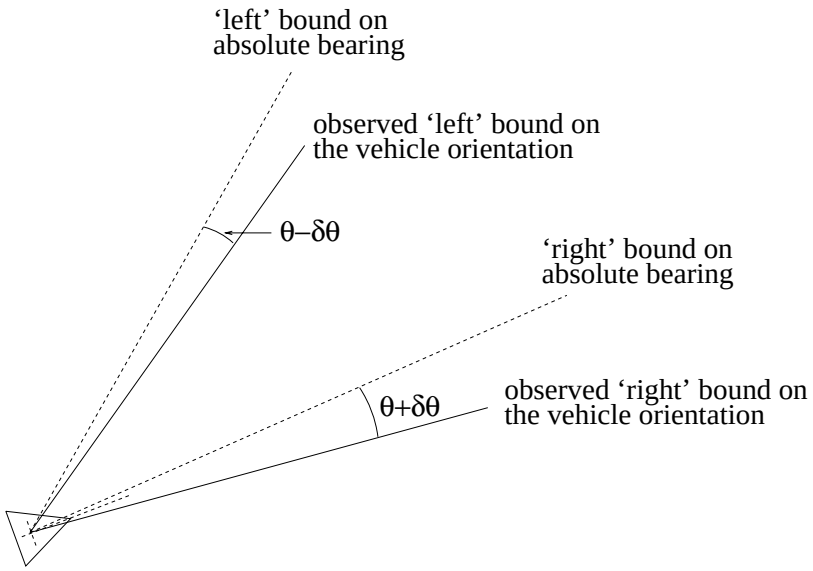

Figure 4.16: Forming the observed bounds on the vehicle orientation.

vehicle orientation bound to produce the orientation update,

$$\begin{aligned} \phi'_l &= \beta_l - \theta + \delta\theta \\ \phi'_r &= \beta_r - \theta - \delta\theta. \end{aligned}$$

The orientation update and the position update for the vehicle and the feature can be repeated consecutively to improve the updates.

4.2.10 Gating and Feature Estimate Initiation

Gating is performed by constructing the valid region for the observed feature as described in Section 4.2.7. The valid region is then checked for an overlap with the bounded estimates of the features in the current map. If one, and only one, bounded estimate overlaps with the valid region, then the feature corresponding to that estimate has gated successfully.

If a feature gates successfully, the observation is used to update the position estimate of the vehicle and the feature, and to update the orientation estimate of the vehicle. If no feature gates successfully, the observation is used to initiate a new feature estimate. The feature estimate is initiated by constructing the smallest rectangle that completely covers the valid region for the observed feature.

4.2.11 Operation of the Bounded Region Filter

This section summarises the operation of the BOR filter.

Start

An observation of a feature is made. The vehicle estimate is predicted forward to the time of the observation. The new observation is gated with features already in the map. If gating is successful, the observation is used to update the map, otherwise it is used to add a new feature to the map.

Prediction, Section 4.2.5

- Predict the new bound on the position and orientation of the vehicle using the current vehicle bounds, the bounds on the control inputs and the process noise, according to Equation 4.7.

Gating, Section 4.2.10

- Calculate the absolute bearing bounds on the observation.
- Combine the range bounds and the absolute bearing bounds with the vehicle position bounds.
- If all four edges that define the valid region of the observed feature produce valid partitions for only one feature estimate, the particular feature gates successfully.

Feature Position Update, Section 4.2.7

- Combine the range bounds and the absolute bearing bounds with the vehicle position bounds to obtain valid partitions of the feature position bound.
- After each partitioning, the smallest rectangle that covers the valid partition is the new feature position bound to be further partitioned. The final rectangle is the feature position update.

Vehicle Position Update, Section 4.2.8

- Calculate the absolute bearing bounds for the reversed observation.
- Proceed as for the feature update, but start the observation with the feature position bound and partition the vehicle position bound.

Vehicle Orientation Update, Section 4.2.9

- Calculate the bounds on the possible absolute bearing of the observation by considering the current position bounds of the vehicle and the observed feature.
- Rotate the calculated bounds by the bearing of the actual observation and increase them to account for uncertainty in the observation. This leads to the observed vehicle orientation bounds.

- Intersect the observed vehicle orientation bounds with the current vehicle orientation bounds to produce the update.

End

- Repeat the Orientation and Position Updates until no significant reduction in the spread of the bounds is achieved.

4.2.12 Summary and Discussion

The BOR filter operates with assumptions very different to those of the Kalman filter. It is assumed that the process and observation noise are bounded. This is justified by considering the physical sources of the noise, such as the slippage of the wheels or the bias in the laser rangefinder timers, and arguing that the noise sources are not expected to produce noise of infinite magnitude.

This assumption is very distinct to the assumption of Gaussian noise distribution made by the Kalman filter and leads to bounded state estimates. The update in the BOR filter can be expressed as the intersection of bounded estimates, and does not have to consider the correlations between any estimate errors. This allows the BOR filter to update estimates individually and still achieve consistency. In particular, after the observation of a feature, only the vehicle estimate and the estimate of the observed features is updated. This is not possible in the MAK filter due to the correlations, as described in Section 3.3, and leads to the following conclusion.

The storage requirement for the solution of the SLAM problem using the BOR filter increases only linearly with the number of features in the environment and the computational burden does not increase at all.

In contrast, the storage and the computational demand for the solution of the SLAM problem using the MAK filter increases as the square of the number of features in the environment, as identified in Section 3.4.

Although the BOR filter has a very significant practical advantage over the MAK filter, it also has disadvantages. The BOR filter only delivers bounds on the true position and orientation of the vehicle and the true position of the features. This is different to the MAK filter, which delivers a mean of the state estimate. Most higher level navigational systems require the knowledge of the AGV pose in terms of a specific position and orientation [13, 43], rather than a bound, consequently the solution provided by the BOR filter may be inadequate.

However, if the bounds within which the orientation and the positions are known to lie are sufficiently small, the effect of the bounds may be insignificant. The centroid of the bounded estimate could be used by the higher level navigational system with little difference to using a mean estimate

with a certain covariance. How small the bounds have to be and how small the BOR filter can make them will depend on the specific application. The applicability of the BOR filter to an actual SLAM problem is investigated in Chapter 5.

The actual implementation of the BOR filter described in this thesis has used the HYPO approach, defining the bounded estimate in terms of separate bounds on each dimension of the state. The bounds on the vehicle and feature positions are given by maximum and minimum bounds on their Cartesian coordinates in a fixed global reference frame, and the bounds on the vehicle orientation is given by two bearings between which the true vehicle orientation must lie. Consequently, the bounds are aligned with the fixed global coordinate frame. This is an arbitrary constraint on the estimates, and is therefore unfortunate, but was accepted in view of the simple update rule it permitted. Ellipsoidal bounded estimates can eliminate this constraint at the cost of increased update complexity [62].

In the particular development presented in this thesis, the bounds on the noise were understood as individual bounds on the magnitude for each dimension of the noise. However, other types of bounds can be envisaged, such as bounds that combine several dimensions and model the interdependency between the different noise sources. This was not pursued since no reasonable justification was seen for the additional complexity.

As will be shown and explained in Chapter 5, the performance of the BOR filter may be unsatisfactory. Therefore, an alternative solution to the SLAM problem is developed in Section 5.8.

4.3 The Relative Filter

4.3.1 Overview

This section develops a fundamentally new solution to the SLAM problem, unique to this thesis. The development builds closely on the insights of Chapters 2 and 3, and arrives at a solution to the SLAM problem that is both consistent and computationally feasible. The solution, referred to as the REL filter, is based on relative distances and angles between the features in the environment.

Given simultaneous [7] range-bearing observations of two distinct features, it is possible to calculate a measure of the relative distance between the features without knowing the state of the vehicle. This is shown in Figure 4.17, where the relative distance d is uniquely determined by r_1, r_2 and $\delta\theta_{12}$. The consequence of this in terms of the SLAM problem can be seen as follows [24].

The observation $\mathbf{z}_{\mathbf{p}_{ij}}$ of a Relative State (REST), such as a relative distance, between two distinct features p_i and p_j can be derived from the simultaneous observation of the features p_i and p_j,

$$\mathbf{z}_{\mathbf{p}_i}(k) = \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}_i(k), k\right] \tag{4.8}$$

$$\mathbf{z}_{\mathbf{p}_j}(k) = \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_j(k), \mathbf{w}_j(k), k\right] \tag{4.9}$$

$$\mathbf{z}_{\mathbf{p}_{ij}}(k) = \mathbf{h}_{2\mathbf{p}}\left[\mathbf{p}_i(k), \mathbf{p}_j(k), \mathbf{w}_i(k), \mathbf{w}_j(k), k\right], \tag{4.10}$$

where $\mathbf{w}_i(k)$ and $\mathbf{w}_j(k)$ are the observation noise in the observation of p_i and p_j respectively, and the observation $\mathbf{z}_{\mathbf{p}_{ij}}(k)$ of the REST, according to the observation model $\mathbf{h}_{2\mathbf{p}}$, does not depend on the vehicle state $\mathbf{x}_v(k)$. If the REST, rather than the individual feature states, is considered as the observed state, Equation 4.10 may be written as

$$\mathbf{z}_{\mathbf{p}_{ij}}(k) = \mathbf{h}_{\delta\mathbf{2p}}\left[\delta\mathbf{p}_{ij}(k), \mathbf{w}(k), k\right], \tag{4.11}$$

where $\delta\mathbf{p}_{ij}(k)$ denotes the REST and the observation noise $\mathbf{w}(k)$ depends only on $\mathbf{w}_i(k)$ and $\mathbf{w}_j(k)$.

A Taylor series expansion of $\mathbf{h}_{\delta\mathbf{2p}}$ about the estimate $\hat{\delta}\mathbf{p}_{ij}(k+1|k)$ of $\delta\mathbf{p}_{ij}(k+1)$ is

$$\begin{aligned}\mathbf{z}_{\mathbf{p}_{ij}}(k+1) = {} & \mathbf{h}_{\delta\mathbf{2p}}\left[\hat{\delta}\mathbf{p}_{ij}(k+1|k), \mathbf{0}, k+1\right] + \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\delta\mathbf{p}_{ij}} \tilde{\delta}\mathbf{p}_{ij}(k+1|k) \\ & + \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\mathbf{w}} \mathbf{w}(k+1) + \text{ higher order terms},\end{aligned}$$

[7] Simultaneous observations were not assumed in the formulation of the localisation problem, Section 2.3.2, but they lead to a simple REL filter and are therefore considered first. A simultaneous observation of several features is referred to as a 'scan' of the features.

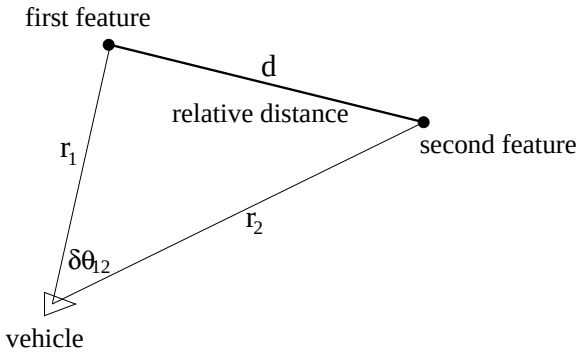

Figure 4.17: Observing a relative distance.

where $\tilde{\delta}\mathbf{p}_{ij}(k+1|k)$ is the error in $\hat{\delta}\mathbf{p}_{ij}(k+1|k)$, and $\nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\delta\mathbf{p}_{ij}}$ and $\nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\mathbf{w}}$ are the Jacobians of $\mathbf{h}_{\delta\mathbf{2p}}$ with respect to $\delta\mathbf{p}_{ij}$ and $\mathbf{w}$. Using the assumption that $\mathbf{w}(k+1)$ is small and zero mean, that the higher order terms are negligible, and that

$$\hat{\delta}\mathbf{p}_{ij}(k+1|k) \quad = \quad \mathrm{E}\left[\delta\mathbf{p}_{ij}(k+1)|\mathbf{Z}^k\right],$$

leads to

$$\begin{aligned}
\mathbf{z}_{\mathbf{p}_{ij}}(k+1) &\approx \mathbf{h}_{\delta\mathbf{2p}}\left[\hat{\delta}\mathbf{p}_{ij}(k+1|k), \mathbf{0}, k+1\right] + \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\delta\mathbf{p}_{ij}} \tilde{\delta}\mathbf{p}_{ij}(k+1|k) \\
&\quad + \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\mathbf{w}} \mathbf{w}(k+1) \\
\\
\hat{\mathbf{z}}_{\mathbf{p}_{ij}}(k+1|k) &= \mathrm{E}\left[\mathbf{z}_{\mathbf{p}_{ij}}(k+1)|\mathbf{Z}^k\right] \\
&\approx \mathbf{h}_{\delta\mathbf{2p}}\left[\hat{\delta}\mathbf{p}_{ij}(k+1|k), \mathbf{0}, k+1\right] \\
\\
\nu(k+1) &\approx \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\delta\mathbf{p}_{ij}} \tilde{\delta}\mathbf{p}_{ij}(k+1|k) + \nabla\left(\mathbf{h}_{\delta\mathbf{2p}}\right)_{\mathbf{w}} \mathbf{w}(k+1). \qquad (4.12)
\end{aligned}$$

The observation noise $\mathbf{w}_i$ in Equation 4.8 and $\mathbf{w}_j$ in Equation 4.8 are assumed not to be correlated with the feature states, as stated in Section 2.4, hence

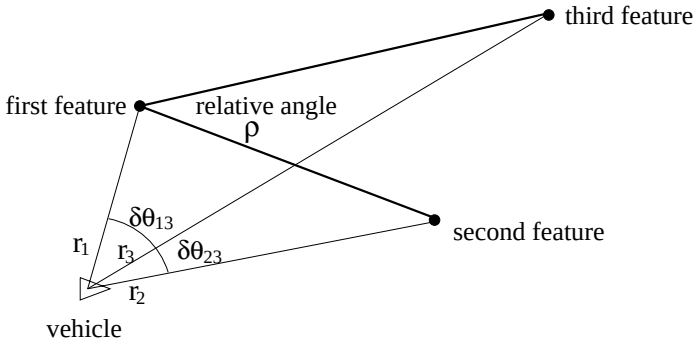

Figure 4.18: Observing a relative angle.

$$\begin{aligned}
\mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\mathbf{w}_i^T(k+1)\right] &= \mathbf{0} \\
\mathrm{E}\left[\tilde{\mathbf{p}}_i(k+1|k)\mathbf{w}_j^T(k+1)\right] &= \mathbf{0} \\
\mathrm{E}\left[\tilde{\mathbf{p}}_j(k+1|k)\mathbf{w}_i^T(k+1)\right] &= \mathbf{0} \\
\mathrm{E}\left[\tilde{\mathbf{p}}_j(k+1|k)\mathbf{w}_j^T(k+1)\right] &= \mathbf{0}.
\end{aligned}$$

Since $\mathbf{w}(k+1)$ in Equation 4.12 depends only on $\mathbf{w}_i(k+1)$ and $\mathbf{w}_j(k+1)$, it follows that

$$\mathrm{E}\left[\tilde{\delta}\mathbf{p}_{ij}(k+1|k)\mathbf{w}^T(k+1)\right] = \mathbf{0}.$$

The innovation $\nu(k+1)$ in Equation 4.12 depends only on the observation noise $\mathbf{w}(k+1)$ and the estimate error $\tilde{\delta}\mathbf{p}_{ij}(k+1|k)$, and $\mathbf{w}(k+1)$ is not correlated with $\tilde{\delta}\mathbf{p}_{ij}(k+1|k)$. Therefore, $\mathbf{z}_{\mathbf{p}_{ij}}(k+1)$ is a COIN observation of $\delta\mathbf{p}_{ij}(k+1)$, as defined in Section 2.2.4. This has the important consequence that all the covariance information for the update of the estimate and the update of its covariance is contained in the covariance of the observation noise and the covariance of the estimate, as shown in Sections 2.2.4 and 2.2.5.

The estimation of relative distances between features can be performed without the need to consider the correlation between errors in different relative distance estimates, because the innovation in the observation of a relative distance depends only on the observation noise and the current relative distance estimate error.

A COIN observation is not limited to two features. For example, a relative angle formed by three features can also be observed with no knowledge of the vehicle state, if simultaneous range-bearing observations of the three features are available. This is shown in Figure 4.18, where the relative

angle ρ is uniquely determined by r_1, r_2, r_3, $\delta\theta_{13}$ and $\delta\theta_{23}$, even if the pose of the vehicle is not known.

The observation $\mathbf{z}_{\mathbf{p}_{ijl}}$ of a REST, such as a relative angle, between three distinct features p_i, p_j and p_l can be expressed as

$$\mathbf{z}_{\mathbf{p}_{ijl}}(k) = \mathbf{h}_{3\mathbf{p}}\left[\mathbf{p}_i(k), \mathbf{p}_j(k), \mathbf{p}_l(k), \mathbf{w}_i(k), \mathbf{w}_j(k), \mathbf{w}_l(k), k\right]. \tag{4.13}$$

Again considering the REST rather than the individual feature states as the observed state, Equation 4.13 may be written as

$$\mathbf{z}_{\mathbf{p}_{ijl}}(k) = \mathbf{h}_{\delta\mathbf{3p}}\left[\delta\mathbf{p}_{ijl}(k), \mathbf{w}(k), k\right], \tag{4.14}$$

where $\delta\mathbf{p}_{ijl}(k)$ denotes the REST and the observation noise $\mathbf{w}(k)$ depends only on $\mathbf{w}_i(k)$, $\mathbf{w}_j(k)$ and $\mathbf{w}_l(k)$. Once again, the observation model has been modified to a COIN observation by considering a new state, $\delta\mathbf{p}_{ijl}$, consisting of a combination of the features. All the necessary covariance information for the update of the estimate $\hat{\delta}\mathbf{p}_{ijl}(k+1|k)$ of $\delta\mathbf{p}_{ijl}(k+1)$ and the update of the covariance of $\delta\mathbf{p}_{ijl}(k+1|k)$ is contained in the covariance of the observation noise and the covariance of the estimate $\delta\mathbf{p}_{ijl}(k+1|k)$.

> The estimation of relative angles formed by three features can be performed without the need to consider the correlation between errors in different relative angle estimates, because the innovation in the the observation of a relative angle depends only on the observation noise and the current relative angle estimate error.

From the consideration of REST observations, the following general conclusion is drawn.

> Relative states between features can be observed according to COIN [8]observations, making their estimation computationally very inexpensive. Therefore, estimating such states in an unknown environment is a straightforward task for an AGV.

Hence a potential solution to the SLAM problem is to

- consider Relative States between features, such as relative distances and angles, and to
- combine the feature observations made relative to the vehicle into REST observations.

[8]A COIN observation is defined in Section 2.2.4.

This is the basis of the REL filter. Although the filter does not maintain an estimate of the vehicle, the vehicle can be localised relative to the features in the map since observations of these features are made relative to the vehicle. Individual feature observations are used to localise the vehicle relative to the features, combined feature observations are used to estimate the relative states between the features.

Past publications on the localisation and also the SLAM problem have some common elements with the approach presented here, but do not achieve the understanding of the advantage of COIN observations in terms of the SLAM problem. The insight presented here is unique to this thesis.

Li [59] suggests a navigational system that relies on scanning the environment and constructing a panoramic representation of it. Although the system uses observations that depend on the position of the vehicle, therefore differentiating it from the REL filter, the panoramic observation contains invariants. These invariants, such as a topological arrangement of features in the environment, do not depend on the state of the vehicle. However, the invariants are not exploited.

Yeh [92] uses a function to classify features in the observation. The function takes the observation as its argument and computes the identity of the feature. The identification, similar to the REL filter, attempts to extract information that is not dependent on the state of the vehicle, " ... the value returned from such functions are invariant to changes of viewpoint and can be evaluated directly from the image measurements without prior knowledge of the position and orientation of the camera [92]." Again, the invariants are not exploited in a solution to the SLAM problem.

The work of Buelthoff [17] also considers invariants for the purpose of navigation. A motion vector is calculated for all features in the environment. It is then argued that features within a surface will have the same motion vector, due to the motion of the vehicle, but obstacles that protrude from the surface will not. Buelthoff extracts the obstacles, but it is also possible to extract the surface and thereby identify features belonging to a group. This information is not dependent on the state of the vehicle, and could therefore be used in a REL filter.

Approaches to the SLAM problem that use neural networks aim to identify relationships between features in the environment. An example of neural networks used in map building is given by Morellas [71]. The connection between features formed by a neural network can be compared to the relative distance estimate formed by the REL filter, but the neural network does not provide quantitative information about the environment.

The detailed development of the REL filter will concentrate on a range-bearing sensor and will only deal with relative distance and angle estimates. However, as was pointed out in this section, the approach is not restricted to range-bearing sensors, nor to estimates of relative distances and relative angles.

4.3.2 Structure of the Relative Map

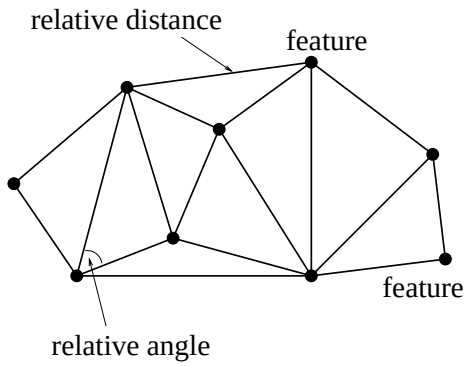

Figure 4.19: A relative map.

Figure 4.19 is an example of a relative map. The map consists of estimates of the relative distance between features, the nodes of the map, and their neighbours. On their own, relative distances do not guarantee an unambiguous map. To eliminate the ambiguity angles are introduced. Given a feature and two of its neighbours, an estimate of the angle subtended by the two neighbouring features is also maintained by the map.

The estimates consist of a mean and a variance. These two parameters are stored for each estimate of relative distance and relative angle. A measure of the correlation between the errors in different estimates is not stored, since this information was identified as unnecessary in Section 4.3.1. The data structure, consisting of isolated estimates and their covariances but not including covariances between different estimates, resembles the data structure used in building a map from a vehicle of perfectly known state, described in Section 2.4.2. This is a consequence of the fact that COIN observations of the estimated states are used in both cases, allowing the state estimates to be formed independently of each other. In map building, COIN observations are a result of the perfect knowledge of the vehicle state, leading to an innovation that depends only on the observation noise and the estimate error in the observed feature. In the REL filter, COIN observations are a result of estimating Relative States between features. In map building, the influence of the vehicle estimate on the innovation is eliminated since the estimate error is zero, and in the REL filter the influence of the vehicle estimate on the innovation is eliminated by forming observations that are independent of the vehicle state.

4.3.3 Building a Relative Map

An important consideration in a REL filter is how the observation made by the sensor on the vehicle, Equation 2.28, is transformed into a REST observation, such as Equations 4.11 and 4.14. If the sensor

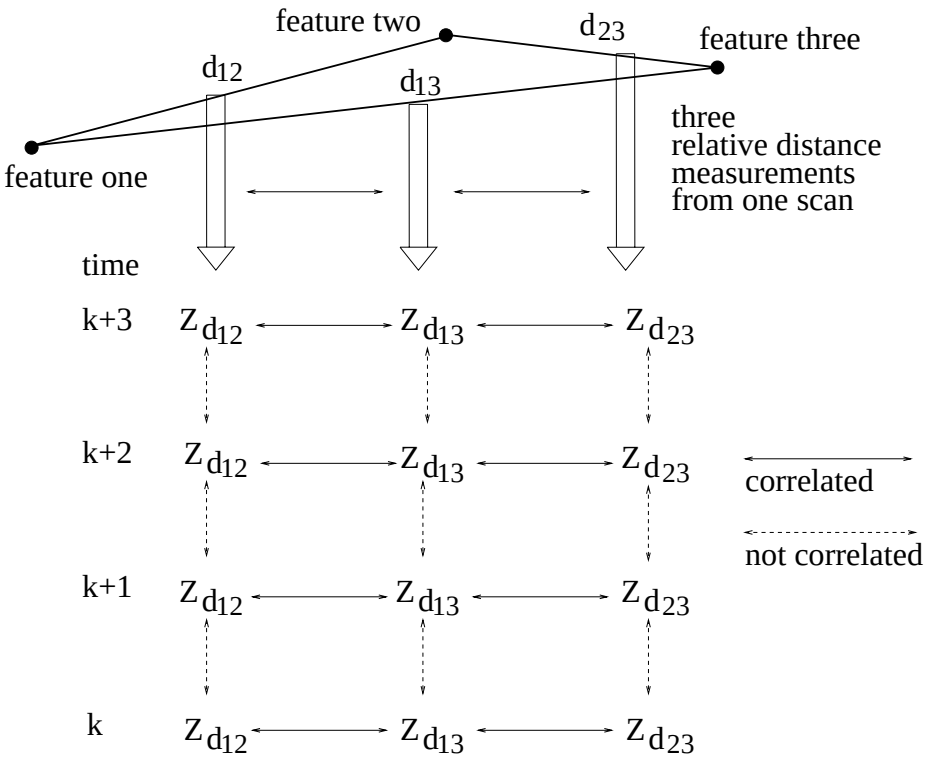

Figure 4.20: Simultaneous observations of three features, scanned at each time step.

provides simultaneous observations of several features, COIN-REST observations can be formed as described in Section 4.3.1, and shown in Figures 4.17 and 4.18 for relative distances and angles in the case of range-bearing observations. The COIN-REST observations lead to simple updates of the REST estimates, as identified in Sections 2.2.4 and 2.2.5, since they are COIN observations.

Simultaneous observations of several features can be used to form observations of several RESTs. For example, the three simultaneous observations shown in Figure 4.18 can be transformed into observations of three relative angles and three relative distances, hence a total of six REST observations. Each of these REST observations is also be a COIN observation, since their innovation depends only on the REST observation noise, derived from the observation noise in the individual feature observations, and the estimate error of the particular REST for which the REST observation was formed. In the example of Figure 4.18, the innovation in each of the relative distance observations, formed by using any two feature observations, has the form of Equation 4.12. Therefore, all RESTs of which observations can be constructed from the simultaneous observation of several features may be updated, each update formed independently of the others.

A REST observation may have correlated errors with another REST observation from the same scan. In Figure 4.18, the observation noise in the observation of the first feature propagates onto the constructed observation of the relative distance between the first and the second feature, and

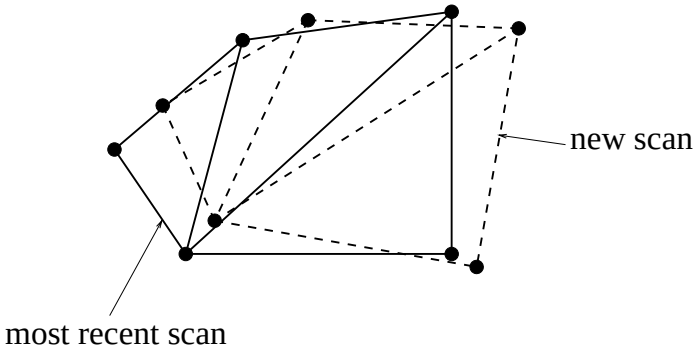

Figure 4.21: Aligning the new scan.

it also propagates onto the constructed observation of the relative distance between the first and the third feature. Hence these two constructed relative distance observations will have correlated errors. However, both constructed observations will only have errors due to the noise in the feature observations, which are assumed uncorrelated with the feature states, and hence uncorrelated with the relative distance between them. Therefore, the constructed observations are COIN observations of the relative distances, and each relative distance estimate is updated independently of the other relative distance estimates. The correlation of errors in three constructed relative distance observations are shown in Figure 4.20.

In order to identify which REST observation corresponds to which current REST estimate, the most recent scan is stored. The new scan can then be aligned with the most recent scan and the new observations associated with the current estimates. The alignment of scans, also known as the correspondence problem, has been investigated in several publications, for example in [69]. A very similar problem is found in submarine navigation. Observations of the sea bed are taken by submarines and the rotation and translation needed to align successive observations are calculated, for example by the ICP approach described in Section 1.2.3. The calculated rotation and translation indicates the rotation and translation of the submarine [23]. The alignment relies on a high observation rate compared to the motion of the vehicle to ensure that a correct alignment between successive observations is achieved.

If simultaneous observations of features are not available, then the approach needs to be extended to take account of the motion of the vehicle between the observations. However, it is still possible to build a relative map and to exploit the advantages of COIN observations. The following sections describe how to build a relative map when simultaneous observations of the features are not available.

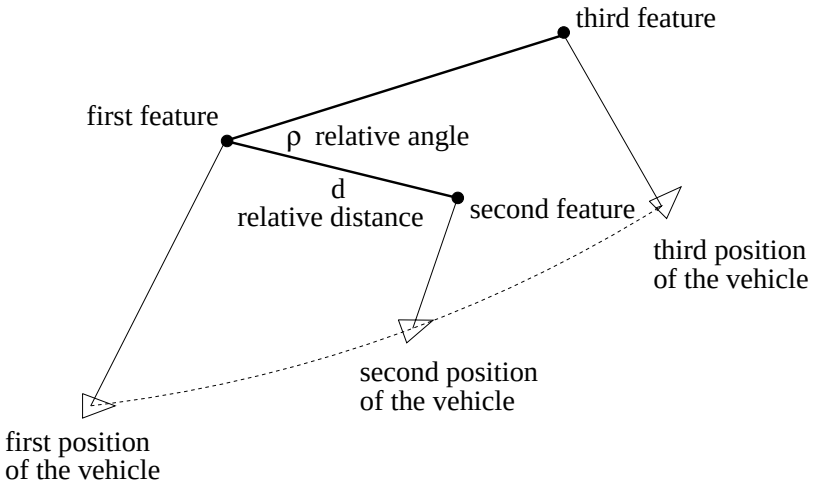

Figure 4.22: Incorporating the motion of the vehicle

4.3.4 Incorporating the Motion of the Vehicle

When the motion of the vehicle cannot be ignored between feature observations, Equation 4.10 is no longer valid. However, given the observation of two distinct features and a measure of the motion of the vehicle between the two observations, it is possible to construct a measure of the relative distance between the two features, as shown in Figure 4.22. Similarly, given the observation of three distinct features and the motion of the vehicle between the observations, it is possible to construct a measure of the relative angle subtended by the three features, as also shown in Figure 4.22. In both cases, the state of the vehicle, different at each observation, is not needed to formulate the REST observation, only the relative motion of the vehicle between observations is needed.

For the relative approach to be applied, the vehicle state estimate must not used in the construction of the REST observations. Using the vehicle state estimate would introduce errors in the constructed REST observations which may be correlated with the errors in the current REST estimates. Such observations would not be COIN observations, knowledge of the introduced correlations would have to be maintained, and the computational advantage of the relative approach would be lost.

In Section 4.3.5 the measure of the relative motion of the vehicle will be developed. Section 4.3.6 combines the relative motion of the vehicle with the feature observations to construct observations of RESTs.

4.3.5 Measuring the Relative Motion of the Vehicle

The state of the vehicle evolves according to the discrete process model in Equation 2.27,

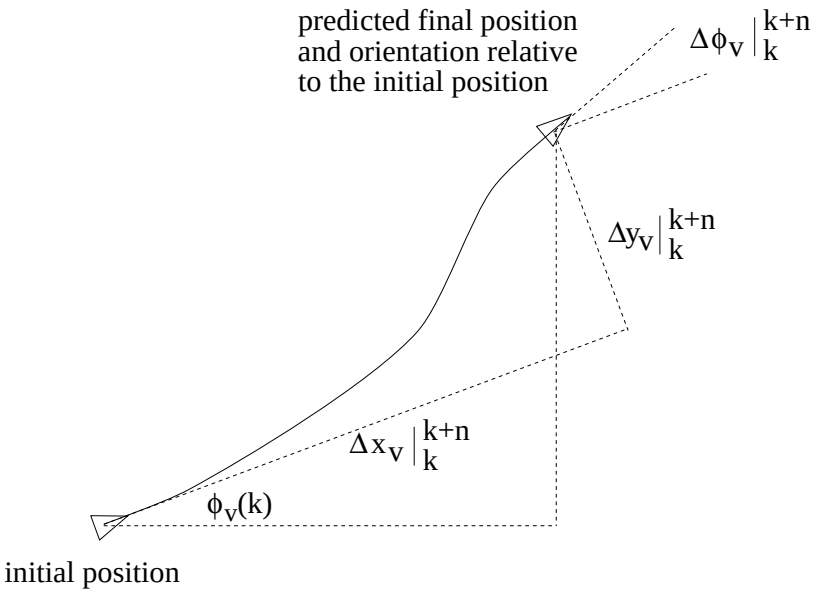

Figure 4.23: Predicting the relative motion for a given set of control inputs.

$$\mathbf{x}_v(k+1) = \mathbf{f}_{\mathbf{x}_v}\left[\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k), k\right].$$

The relative motion of the vehicle from a time k to a time $k+n$, in the reference frame of the vehicle at time step k, is described by a vector $\Delta\mathbf{x}_v|_k^{k+n}$,

$$\mathbf{x}_v(k+1) \quad = \quad \mathbf{x}_v(k) + \left[\Delta\mathbf{x}_v|_k^{k+1}\right]_{global},$$

where $\left[\Delta\mathbf{x}_v|_k^{k+1}\right]_{global}$ is the relative motion transformed into the global reference frame.

For the vehicle state defined in Equation 2.25, $\Delta\mathbf{x}_v|_k^{k+n}$ is

$$\Delta\mathbf{x}_v|_k^{k+n} \quad = \quad \begin{bmatrix} \Delta x_v|_k^{k+n} \\ \Delta y_v|_k^{k+n} \\ \Delta\phi_v|_k^{k+n} \end{bmatrix},$$

where $\Delta x_v|_k^{k+n}$ is the relative motion in the direction of the vehicle heading at time step k, $\Delta y_v|_k^{k+n}$ is the relative motion perpendicular to the vehicle heading at time step k and $\Delta\phi_v|_k^{k+n}$ is the orientation change of the vehicle since time step k, as shown in Figure 4.23.

The relative motion is used to form REST observations, therefore the prediction of the relative motion must not depend on the state of the vehicle. This restricts the process model to

$$\mathbf{x}_v(k+1) \quad = \quad \left[\mathbf{f}_{\Delta\mathbf{x}_v}\left[\mathbf{u}(k), \mathbf{v}(k), k\right]\right]_{global} + \mathbf{x}_v(k),$$

where $\mathbf{f}_{\Delta\mathbf{x}_v}$ is not a function of $\mathbf{x}_v$. The specific vehicle model of Section 2.3.3 is of this form. The relative motion over n time steps is

$$\Delta\mathbf{x}_v|_k^{k+n} \quad = \quad \mathbf{f}^n_{\Delta\mathbf{x}_v}\left[\mathbf{u}(k), \ldots, \mathbf{u}(k+n-1), \mathbf{v}(k), \ldots, \mathbf{v}(k+n-1)\right]. \tag{4.15}$$

and its prediction is

$$\hat{\Delta}\mathbf{x}_v|_k^{k+n} \quad = \quad \mathbf{f}^n_{\Delta\mathbf{x}_v}\left[\mathbf{u}(k), \ldots, \mathbf{u}(k+n-1), \mathbf{0}, \ldots, \mathbf{0}\right]. \tag{4.16}$$

The relative motion prediction in Equation 4.16 is in a form that does not depend on the state of the vehicle, this was identified in Section 4.3.4 as necessary. It follows from Equation 4.16 that the prediction of the relative motion must be based only on control inputs, such as those obtained from encoders and gyroscopes.

The control inputs to the vehicle that occur between observations have to be temporarily stored so that the relative vehicle motion between individual observations can be predicted. Storing the control inputs in a buffer circular allows the relative motion of the vehicle during different intervals covered by the buffer to be predicted, and is shown in Figure 4.24, yet avoids the storage requirement growing without bounds. The control inputs are stored rather than a relative motion prediction for each time step, because the predictions would not be in a common reference frame and therefore could not be easily combined when a prediction over several time steps is needed.

It will be important later in the development to know if the errors in predictions of the relative motion over different intervals are correlated. From Equations 4.15 and 4.16 it can be seen that the errors in predictions will be uncorrelated as long as the intervals of the predictions don't overlap, since $\mathbf{v}$ is assumed uncorrelated, $\mathrm{E}\left[\mathbf{v}(i)\mathbf{v}(j)\right] = \mathbf{0}$ for $i \neq j$.

The following section describes how the measures of relative motion developed above are combined with the feature observations to form constructed REST observations.

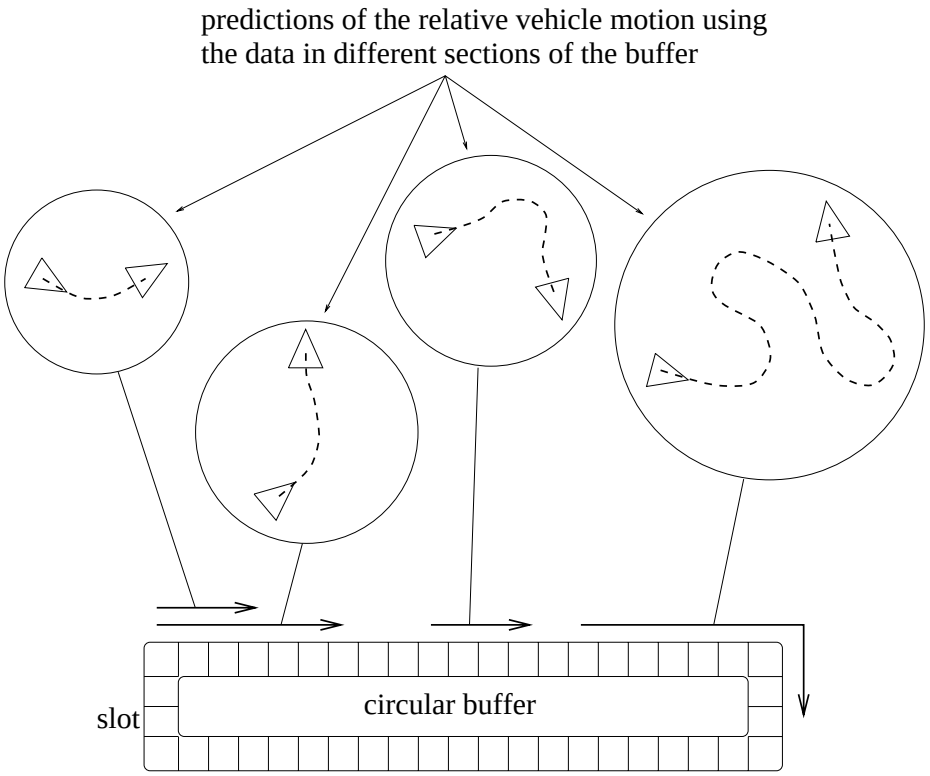

Figure 4.24: The circular buffer

4.3.6 Combining the Predicted Motion with the Observations

To include the relative motion of the vehicle between the observations of p_i at time step k and p_j at time step $k+n$, Equations 4.8 and 4.9 are modified to

$$\begin{aligned} \mathbf{z}_{\mathbf{p}_i}(k) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k), \mathbf{p}_i(k), \mathbf{w}_i(k), k\right] \\ \mathbf{z}_{\mathbf{p}_j}(k+n) &= \mathbf{h}_{\mathbf{x}_v,\mathbf{p}}\left[\mathbf{x}_v(k+n), \mathbf{p}_j(k+n), \mathbf{w}_j(k+n), k+n\right], \end{aligned}$$

and combined as

$$\mathbf{z}_{\mathbf{p}_{ij}}(k+n) = \mathbf{h}_{2\mathbf{p}\Delta\mathbf{x}_v}\left[\mathbf{p}_i(k), \mathbf{p}_j(k+n), \mathbf{w}_i(k), \mathbf{w}_j(k+n), \mathbf{v}|_k^{k+n}, k, k+n\right], \qquad (4.17)$$

where $\mathbf{v}|_k^{k+n} = \{\mathbf{v}(k), \ldots, \mathbf{v}(k+n-1)\}$ is the accumulated process noise from time step k to $k+n-1$ and is associated with the relative motion $\Delta\mathbf{x}_v|_k^{k+n}$ of the vehicle between the observations. Considering the REST $\delta\mathbf{p}_{ij}$, the observation model is modified to

$$\mathbf{z}_{\mathbf{p}_{ij}}(k+n) \quad = \quad \mathbf{h}_{\delta\mathbf{2p}\Delta\mathbf{x}_v}\left[\delta\mathbf{p}_{ij}(k+n), \mathbf{w}(k+n), \mathbf{v}|_k^{k+n}, k, k+n\right], \tag{4.18}$$

where the REST observation noise $\mathbf{w}(k+n)$ depends only on $\mathbf{w}_i(k)$ and $\mathbf{w}_j(k+n)$. The assumption that REST $\delta\mathbf{p}_{ij}$ does not change over the period k to $k+n$ was also used.

The noise term $\mathbf{w}(k+n)$ depends only on $\mathbf{w}_i(k)$ and $\mathbf{w}_j(k+n)$, which are not correlated with $\tilde{\delta}\mathbf{p}_{ij}$, and the noise term $\mathbf{v}|_k^{k+n}$ is the accumulated process noise, which is also not correlated with $\tilde{\delta}\mathbf{p}_{ij}(k+n|k)$. Hence $\mathbf{w}(k+n)$ and $\mathbf{v}|_k^{k+n}$ can be treated as a combined noise, which is not correlated with $\tilde{\delta}\mathbf{p}_{ij}(k+n|k)$. Therefore, Equation 4.18 represents a COIN observation and the update of $\hat{\delta}\mathbf{p}_{ij}(k+n|k)$ is straightforward, involving only the covariance of the combined 'observation' noise and the covariance of the current estimate $\hat{\delta}\mathbf{p}_{ij}(k+n|k)$.

If $\Delta\mathbf{x}_v|_k^{k+n}$ were formed using a vehicle state estimate, then the error in the state estimate would appear in Equation 4.18, possibly correlating the observation error with $\tilde{\delta}\mathbf{p}_{ij}$, and rendering Equation 4.18 a non-COIN observation. The correlations would have to be considered, and the computational advantage of the relative approach would be lost.

The fusion of the feature observations and the predicted relative motion of the vehicle between the observations is now developed for the specific vehicle and observation models of Section 2.3.3.

Denoting the Cartesian coordinates of the first and second feature with respect to the first vehicle position as (x_1, y_1) and (x_2, y_2) respectively, as shown in Figure 4.25, the following expressions are obtained,

$$\begin{aligned} x_1 &= r_1\cos(\theta_1) \\ y_1 &= r_1\sin(\theta_1) \\ x_2 &= \Delta x + r_2\cos(\Delta\phi_2 + \theta_2) \\ y_2 &= \Delta y + r_2\sin(\Delta\phi_2 + \theta_2) \\ d &= \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2} \end{aligned} \tag{4.19}$$

Where (r, θ) is an observation of a feature provided by the sensor and $(\Delta x, \Delta y, \Delta\phi)$ is the change in position and orientation of the vehicle between the two observations. The desired relative distance between the two features is d.

A relative angle is measured by observing three distinct features as shown in Figure 4.26. Denoting the Cartesian coordinates of the first, second and third observed feature as (x_1, y_1), (x_2, y_2) and (x_3, y_3) respectively, the following expressions are obtained. The desired relative angle subtended at

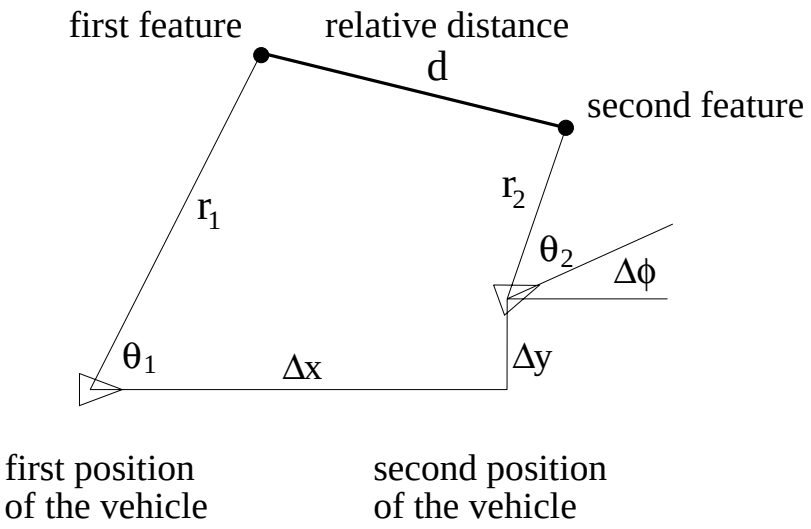

Figure 4.25: Calculating the relative distance

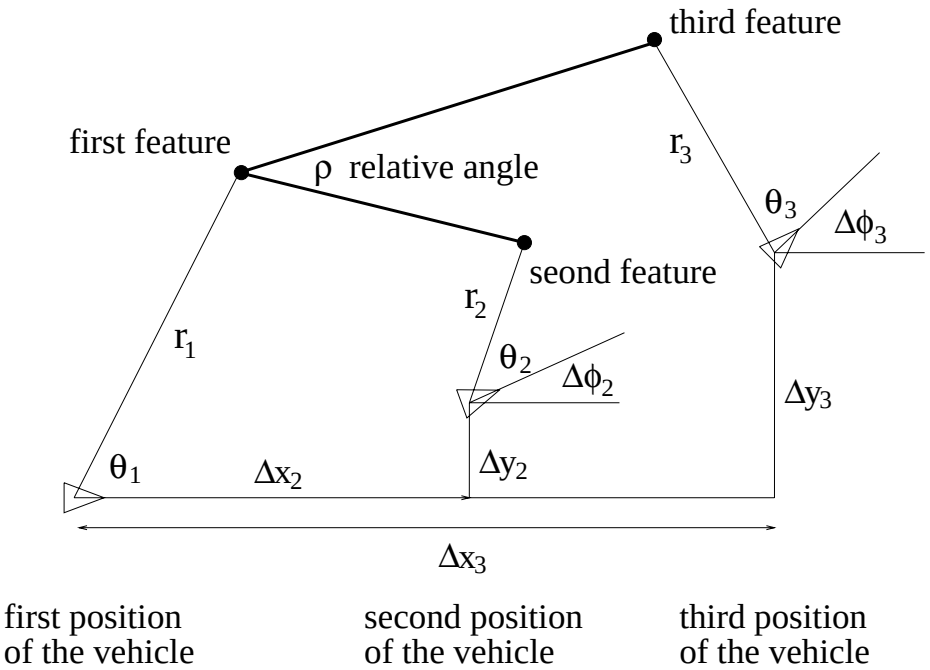

Figure 4.26: Calculating the relative angle

the first feature by the third and second features is ρ,

$$\begin{aligned} x_1 &= r_1 \cos(\theta_1) \\ y_1 &= r_1 \sin(\theta_1) \\ x_2 &= \Delta x_2 + r_2 \cos(\Delta\phi_2 + \theta_2) \\ y_2 &= \Delta y_2 + r_2 \sin(\Delta\phi_2 + \theta_2) \\ x_3 &= \Delta x_3 + r_3 \cos(\Delta\phi_3 + \theta_3) \\ y_3 &= \Delta y_3 + r_3 \sin(\Delta\phi_3 + \theta_3) \\ \rho &= \tan^{-1}(\frac{y_3 - y_1}{x_3 - x_1}) - \tan^{-1}(\frac{y_2 - y_1}{x_2 - x_1}) \end{aligned} \tag{4.20}$$

The observations are stored in the circular buffer described in Section 4.3.5. The circular buffer contains recent observations and the control inputs that were applied to the vehicle between the observations. The motion of the vehicle between observations can therefore be determined as described in Section 4.3.5 and combined with the observations according to Equations 4.19 and 4.20 to produce measurements of relative distance and angle between the features.

As new observations of RESTs are formed, it is important to ensure that the errors in the observations are not correlated with the errors in the current REST estimates, otherwise the observations are non-COIN. Correlations between the errors in subsequent observations of a particular REST must be avoided.

4.3.7 Avoiding Correlations

If two observations, $\mathbf{z}_i$ and $\mathbf{z}_j$, were used to form an observation d_{ij} of the relative distance between two features p_i and p_j, two more observations, $\mathbf{z}'_i$ and $\mathbf{z}'_j$, are needed if a new observation d'_{ij} is to be formed such that the errors in d'_{ij} and d_{ij} are uncorrelated. For the same reason three new observations, one of each feature, are needed in order to form a new COIN observation of a relative angle.

A measurement of a relative distance requires two observations and also requires a prediction of the relative motion of the vehicle between the observations. The error in the prediction of the relative motion $\Delta\mathbf{x}_v|_i^j$ between the two observations $\mathbf{z}_i$ and $\mathbf{z}_j$ will be correlated with the error in the prediction of the relative motion $\Delta\mathbf{x}'_v|_i^j$ between the two other observations $\mathbf{z}'_i$ and $\mathbf{z}'_j$ if and only if the interval covered by $\Delta\mathbf{x}_v|_i^j$ overlaps with the interval covered by $\Delta\mathbf{x}'_v|_i^j$, as described in Section 4.3.5.

Since observations are entered chronologically into the buffer, insisting on a completely new set

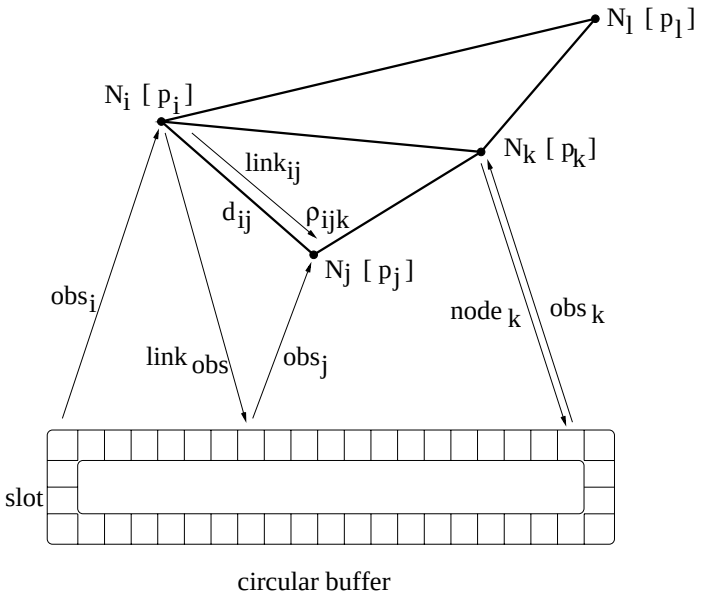

Figure 4.27: A relative map and its buffer

of observations for constructing d'_{ij} assures that the error in $\Delta\mathbf{x}_v|_i^j$ will not be correlated with the error in $\Delta\mathbf{x}'_v|_i^j$. Consequently, the error in d'_{ij} will not be correlated with the error in d_{ij}. Generally, the current estimate of the relative distance between two features p_i and p_j should only be updated when a new observations of p_i and a new observations of p_j are available.

Similar arguments lead to the conclusion that three new observations $\mathbf{z}_i$, $\mathbf{z}_j$ and $\mathbf{z}_k$ of features the p_i, p_j and p_k respectively are necessary to permit the construction of a measurement of the relative angle ρ_{ijk} that will have an error which is uncorrelated to the current estimate of ρ_{ijk}.

The construction of a new measurement of ρ_{ijk} will involve predictions of the relative motion $\Delta\mathbf{x}_v|_j^i$ and $\Delta\mathbf{x}_v|_j^k$, which will have correlated errors if the intervals covered by $\Delta\mathbf{x}_v|_j^i$ and $\Delta\mathbf{x}_v|_j^k$ overlap. Whether the intervals overlap depend on the order in which $\mathbf{z}_i$, $\mathbf{z}_j$ and $\mathbf{z}_k$ were made. However, the correlation can be determined from Equations 4.15 and 4.16, and does not change the fact that the error in the new measurement of ρ_{ijk} is uncorrelated with the error in the current estimate of ρ_{ijk}.

4.3.8 Details of the Relative Filter

A particular feature p_i, represented by the node N_i in the relative map, is typically linked to only a few other features in order to limit the storage requirement and the number of updates that each observation requires. Linking remote features with each other also leads to a large circular buffer, since the vehicle has to travel a considerable distance between the observation of remote features. The large buffer is a disadvantage, because it represents a higher storage requirement.

The advantage of remote links is also doubtful since the variance of the relative distance and relative angle estimates associated with remote features will be very high. The large variance is a direct result of the long distance traveled between the observations needed to construct such estimates.

A link L_{ij} of node N_i to node N_j consists of an estimate and the associated variance of the distance d_{ij} between features p_i and p_j. Associated with a link L_{ij} is a reference $link_{ij}$ that identifies N_j and a reference $link_{ji}$ that identifies N_i. These references allow the nodes N_i and N_j to identify each other, as shown in Figure 4.27.

Each observation held in the circular buffer has a reference that identifies the node in the map with which it has been associated, for example the reference obs_k for the observation of p_k in Figure 4.27. Each feature can have only one observation of itself in the buffer. If a new observation is associated with a particular feature, the previous observation of that feature is removed from the buffer. A node N_i in the map has a reference $node$ which identifies the observation of p_i found in the buffer, or has a void reference if no observation is currently available. An example of such a reference is given by $node_k$ for N_k in Figure 4.27.

For each link L_{ij} the node N_i has a reference $link_{obs}$ that identifies the last observation of p_j that was used to update L_{ij}, or has a void reference if that observation of p_j is no longer available. The reference $link_{obs}$ is compared to $node_j$ to ensure that a new observation of p_j is used in the update of d_{ij}.

The estimate of the relative angle ρ_{ijk}, formed by three features p_i, p_j and p_k, with p_j at the center, is stored with node N_j. Associated with ρ_{ijk} are two references $angle_i$ and $angle_k$ which identify the last observation of p_i and p_k that were used to update the estimate of ρ_{ijk}. The reference $angle_i$ and/or $angle_k$ may be a void reference if the particular observation is no longer in the buffer. The references $angle_i$ and $angle_k$ are compared to $node_i$ and $node_k$ to ensure that new observations of p_i and p_k are used in the update of ρ_{ijk}.

4.3.9 Operation of the Relative Filter

The operation is most easily understood by considering a typical filtering cycle.

Start:

An observation $\mathbf{z}_{new}$ is available of the feature p_{new}. The new observation is gated with features already in the map. If gating is successful, the observation is used to update the map, otherwise it is used to add a new feature to the map.

Gate:

It is desired to test whether p_{new} is equivalent to a known feature. A successful test must pass a relative distance test followed by a relative angle test.

Relative Distance Test

- An observation $\mathbf{z}_1$ is selected from the buffer; the feature and the node associated with $\mathbf{z}_1$ shall be called p_1 and N_1 respectively.
- The relative distance d between p_1 and p_{new} is calculated using $\mathbf{z}_{new}$ and $\mathbf{z}_1$.
- d is tested against all the links of N_1.
- If none of the links are successfully tested, a new p_1 is chosen and the process is repeated until a link gates successfully or no more observations are available in the buffer, in which case $\mathbf{z}_{new}$ could not be associated with any known feature and will be added to the map.
- If a link gates successfully, the feature p_{test} associated with the node N_{test} at the other end of the link is potentially equivalent to p_{new}, and a relative angle test is performed on p_{test}.

Relative Angle Test

- The links of N_1 not leading to N_{test} are examined. A link of N_1 to a node N_2 is sought that is associated with a feature p_2 of which an observation $\mathbf{z}_2$ is still in the buffer.
- The feature p_2 must form the same angle with p_1 and p_{test} as it does with p_1 and p_{new} in order for the angle test to be successful.
- If the angle test fails, p_{new} and p_{test} are not the same feature, and a new p_1 is selected and the distance tests are resumed.
- If the angle test is successful, $\mathbf{z}_{new}$ can be associated with p_{test} and an update can be made. However, it should be verified that no other features pass the distance and angle test. This reduces the possibility of an incorrect association due to ambiguity. The additional effort is limited due to the finite buffer size. During the discussion of the update, p_{test} will be referred to as p_j.

Update:

A new observation of a feature p_j in the map has been identified. The observation is used to calculate a new measurement of the relative distances and angles that involve p_j.

- The links of N_j are examined. The reference $link_{obs}$ of link L_{ji} is checked to identify the last observation of p_i that was used to update L_{ji}. If this is different to the most recent observation of p_i, a new measurement of the relative distance d_ji can be formed and fused with the current estimate, because a new observation of p_j *and* p_i has already been made. All references are updated to reflect the use of the new observations. This process is repeated for all links of N_j.

- The angles formed at N_j are examined. For each angle ρ_{ijk}, with j fixed, the references $angle_i$ and $angle_k$ are checked to identify the last observation of p_i and p_k that were used to update ρ_{ijk}. If both are different to the most recent observation of p_i and p_k, a new measurement of the relative angle ρ_{ijk} can be formed using $\mathbf{z}_{new}$ and the most recent observations of p_i and p_k. The new measurement can be fused with the current estimate of ρ_{ijk}. All references are updated to reflect the use of the new observations. The process is repeated for all angle estimates at N_j.

Adding A New Feature:

The observation $\mathbf{z}_{new}$ of p_{new} failed to gate with any feature in the map. A new node will be created to represent p_{new}. In the discussion below, the new node, feature and observation thereof will be referred to as N_k, p_k and $\mathbf{z}_k$ respectively.

- N_k is joined to those of its nearest neighbours of which there is still an observation in the buffer. For a neighbouring node N_j, a measurement of the relative distance between p_j and p_k is formed using $\mathbf{z}_j$ and $\mathbf{z}_k$, and is entered as the distance estimate for the new link L_{jk}. All the necessary references are initialised.

- Previous links L_{ji} of N_j are examined. If an observation of $\mathbf{z}_i$ of a feature p_i is available in the buffer, a measurement of ρ_{ijk} can be constructed using $\mathbf{z}_i$, $\mathbf{z}_j$ and $\mathbf{z}_k$, and it entered as the angle estimate of ρ_{ijk}. All the necessary references are initialised.

- Angle measurements with N_k at the center are also constructed after more than one link has been made with N_k.

4.3.10 Summary and Discussion

The Relative (REL) filter is a simple Kalman filter. The main difference between the REL filter and the Map Augmented Kalman (MAK) filter is that the REL filter estimates relative distances and relative angles between features, whereas the MAK filter estimates the absolute positions of the features and the vehicle. This difference leads to an important difference in terms of the observation models employed by the two filters.

The observation in the MAK filter is a relative observation of the feature position with respect to the vehicle position. In Section 3.3 it was shown that relative observations lead to correlations between the errors in all state estimates in the MAK filter. It was also shown in Section 3.3 that it is necessary to maintain an explicit representation of these correlations in terms of the covariance matrix, which was identified as a significant storage and computational burden.

In contrast to the MAK filter, the REL filter uses an observation model that allows the estimated states to be observed via COIN observations. The observations of relative distances and relative angles do not depend on the state of the vehicle, as was shown in Section 4.3.1. Consequently, the REL filter can update estimates individually and still achieve consistency. In particular, only the observed states need to be updated with an observation, which has a significant reduction on the complexity of the solution. The REL filter ignores correlations between the error in estimates of different relative distances and relative angles.

The storage requirement for the solution of the SLAM problem using the REL filter increases only linearly with the number of features in the environment and the computational burden does not increase at all.

If the states of interest are not observed in a single observation, it may still be possible to formulate COIN observations of them. This can be achieved if the observation only depends on the change in the vehicle state between the required observations, and is not dependent on the vehicle state itself. If this condition is met, the observations and a prediction of the vehicle motion between the observations can be combined to formulate a measure of the state of interest. The error in this measure will not be correlated to a second measure constructed in a similar way, provided the time intervals used to construct the measures do not overlap, as described in Sections 4.3.4 – 4.3.7. Since the errors are not correlated, the update only involves the observed state, leading to the linear storage and constant computation identified above.

The vehicle state is not used by the REL filter when building a map of the environment. This avoids the correlation problem. In the case where the change in the vehicle state between observations is used, it is necessary to predict the change without using an estimate of the vehicle state, since maintaining a vehicle state estimate would recreate the correlation problem.

The localisation of the vehicle is implicit, since the actual relative observations made by the sensor are associated with nodes in the map, and therefore give a measure of the vehicles position and orientation with respect to the nodes, and hence the map. This directly leads to the localisation of the vehicle with respect to the environment, since the nodes represent the features in the environment.

Since the vehicle will be localised with respect to the features, rather than with respect to an absolute reference frame more commonly used in applications, the REL filter may have a disadvan-

tage in this respect. However, the benefit of a real time solution to the SLAM problem motivates the adoption of a relative framework for applications of AGVs.

4.4 Summary

This chapter examined two alternative algorithms to the SLAM problem. Both alternative algorithms, the BOR filter and the REL filter, have lower memory and computational demands than the MAK filter, allowing a real time implementation.

In particular, for an environment of N features, the storage and computational demands grow as N^2 in the MAK filter. For the BOR and the REL filter, the storage requirement grows as N and the computational demand does not increase. The difference can be illustrated by considering an increase in the number of features in the environment by a factor of 1000. For such an increase, the demand in storage increases by a factor of 1000 and the demand in computation remains unchanged for both the BOR and the REL filters. The same increase in the MAK filter leads to an increase in both storage and computation by a factor of a million.

Both filters achieve this advantage over the MAK filter by avoiding the problem of having to maintain a measure of the correlations between estimate errors. However, the approach to solving the correlation problem is different in the two filters.

The BOR filter operates under the assumption of bounded process and observation noise. This allows the BOR filter to make bounded predictions based on bounded estimates, and also allows the BOR filter to use the observations to form bounded measurements of the states. Since the true state is assumed to lie within the volume of the state space defined by the the bounded estimate, it follows that the true state must also lie within the intersection of two bounded estimates of the same state. The update of the BOR filter can therefore be formulated as the intersection of the current bounded estimate with the bounded measurement obtained from the observation. This update rule is very simple and does not require correlations between state estimate errors to be considered. Consequently, the BOR filter only needs to store a single estimate for the vehicle and each feature, which leads to a linear growth in storage requirement with respect to the number of features. A further consequence of the simple update rule is that only the observed states have to be updated, which leads to a constant computational burden since features are assumed to be observed individually.

The main disadvantage of the BOR filter is that the estimates are given by bounds, which may make subsequent processing difficult. However, if the bounds are sufficiently small the actual position within the bounds may not be important.

The REL filter operates under the same assumptions as the MAK filter but estimates relative

states, such as relative distances and relative angles between features. The observation of a relative state does not depend on the state of the vehicle. Consequently, the update of a relative state estimate involves only the estimate and the new observation; correlations between estimate errors do not have to be considered.

By considering a limited number of relative states involving each feature, the number of states estimated grows linearly with the number of features. Since no correlations have to be considered, the storage also grows linearly and the computation burden does not increase with the number of features.

The main disadvantage of the Relative filter is the absence of absolute position estimates. In particular, the REL filter can only localise the pose of the vehicle relative to the features. However, this disadvantage may not be significant if, for the particular application, a feature-relative solution to the SLAM problem is acceptable.

Chapter 5

Experiments

5.1 Introduction

This chapter presents an experimental evaluation of the simultaneous localisation and map building algorithms developed in the previous chapters. A mobile robot explores a corridor and each filter is used to build a map of the corridor. The corridor consists of a set of point features that can be observed by a sensor mounted on the vehicle. The data is collected and processed off line so that each filter can be examined under identical conditions.

In Section 5.2 the environment is briefly described. This is followed by a description of the sensor in Section 5.3. Section 5.3.1 outlines how the raw sensor data is processed to obtain range and bearing to the features. The vehicle is described in Section 5.4. Preliminary investigations of the collected data and the vehicle and observation models used by the filters are made in Section 5.5. Sections 5.6, 5.7 and 5.8 present the results of the MAK, the BOR and the REL filter respectively, making comparisons of the filters and drawing conclusions. The chapter is summarised in Section 5.9.

5.2 The Environment

All the filters developed in the previous chapters assumed that the environment consists only of point features. While the algorithms do not preclude the use of more complex features, the extraction and processing of such features is considered unnecessary to demonstrate the central theory of simultaneous localisation and map building, and is thus considered beyond the scope of this thesis. Therefore an environment consisting only of point features was employed for the experiments described in this chapter.

A corridor with a staircase was chosen as the experimental scenario, because the staircase was

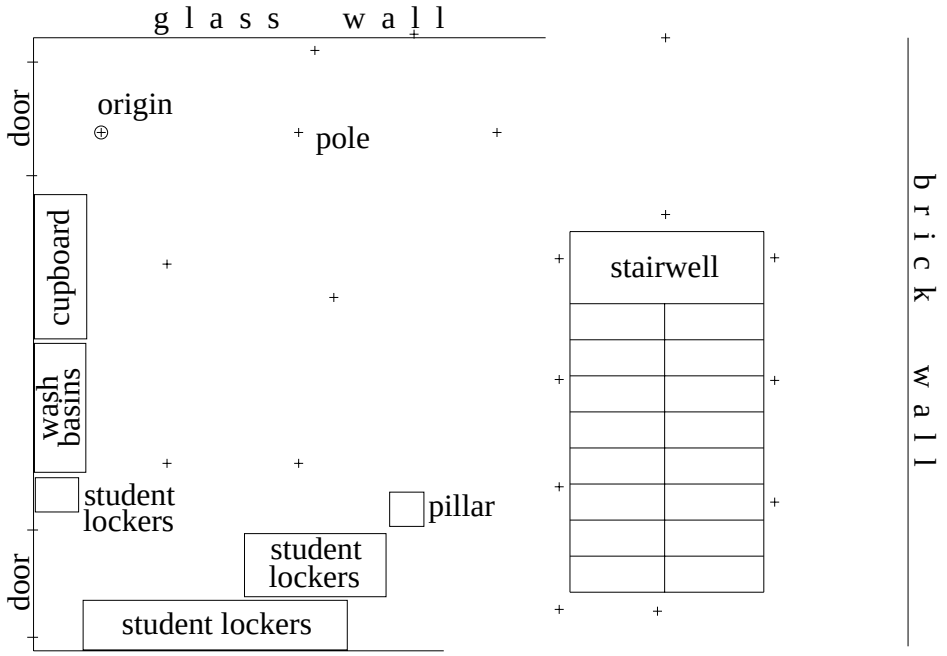

Figure 5.1: Schematic representation of the test environment.

surrounded by railings with vertical struts. The struts were sufficiently thin to be considered as point features on the two dimensional floor. To increase the number of features, and thus create a more realistic environment, poles with a thickness similar to the vertical struts were placed in the corridor. The total number of features was 18.

A schematic representation of the environment is given in Figure 5.1. Photographs of the environment are shown in Figures 5.2 to 5.5.

The poles and the struts were surveyed and their coordinates established within a fixed reference frame, to an accuracy better than 5mm, so that the maps produced by the filters could be evaluated. In the same reference frame a set of waypoints was produced which the vehicle could follow. The waypoints were chosen so as to minimise sharp turns, since experience shows that sharp turns lead to poor predictions.

Figure 5.6 shows the waypoints and the desired path of the vehicle (approximated by line segments joining the waypoints).

Figure 5.2: Photograph of the test environment showing the poles and the vertical struts in the railings. The reflective strips visible below the wash basins and on the pillar were not used.

Figure 5.3: Elevated view of the test environment. Each way point for the vehicle is marked with a '+.'

Figure 5.4: Photograph of the test environment with the railings in the background.

Figure 5.5: Photograph of the test environment showing the floor of the corridor.

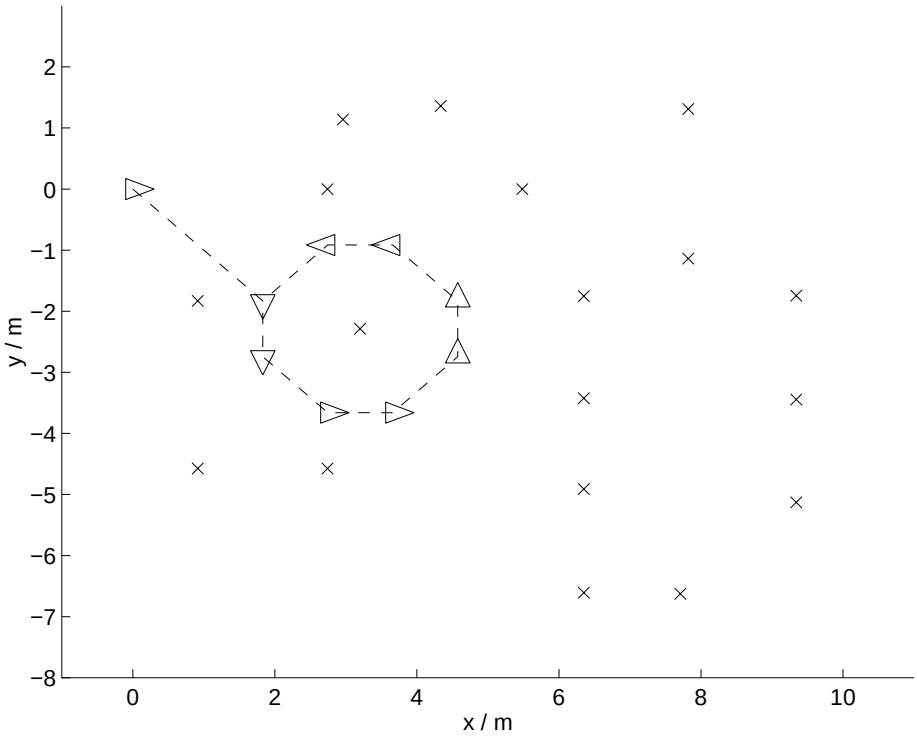

Figure 5.6: The true position of the features and the approximate path of the vehicle.

5.3 The Sensor

The sensor used was a commercial laser rangefinder, shown in Figure 5.7. This sensor scans a semicircle at about 1Hz and returns the range of the closest laser reflection within every degree. A typical scan is shown in Figure 5.8. The figure shows that not only the poles and the struts reflect the laser beam but virtually every other object in the environment. By examining typical scans, it was observed that the desired point features always appear as small clusters. In order for the sensor data to be used by the map building filters, the range and bearing of these small clusters has to be extracted. The development of the extraction algorithm, examined in the next section, is not central to the thesis and was so simple that it did not justify an extensive search of available algorithms.

5.3.1 Extracting Range And Bearing

The feature extraction algorithm relies on the small size of clusters and the free space around them. Thus small isolated features are assumed. Placing the poles far from each other and from other objects in the environment assures that the assumptions are valid.

For the operation of the algorithm, thresholds were defined that describe how much a valid cluster, produced by a pole or a strut, can be spread out and how large the free space has to be

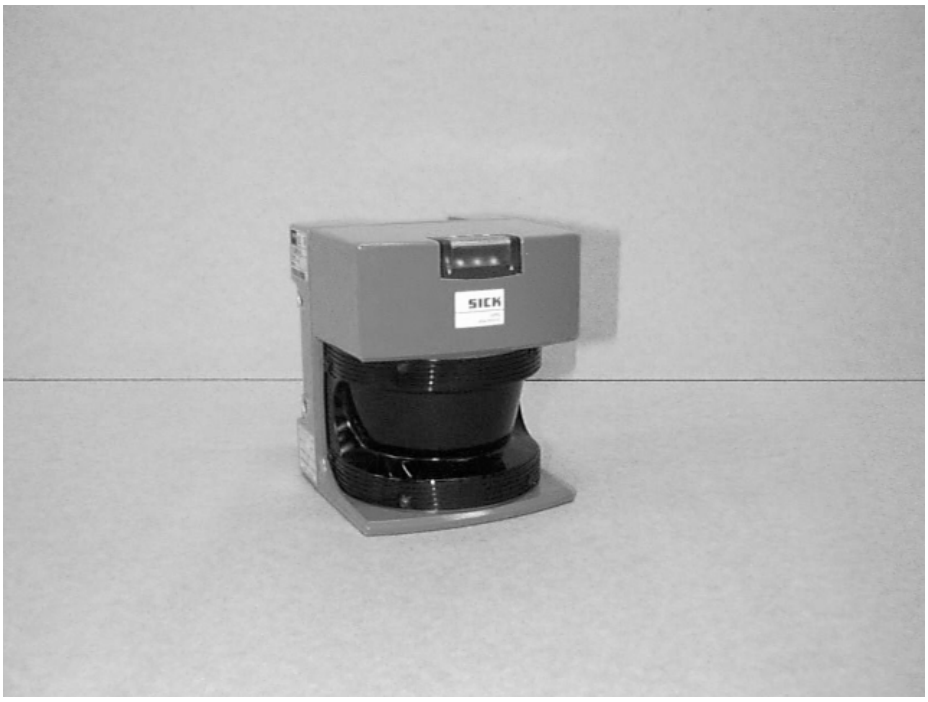

Figure 5.7: The SICK laser rangefinder.

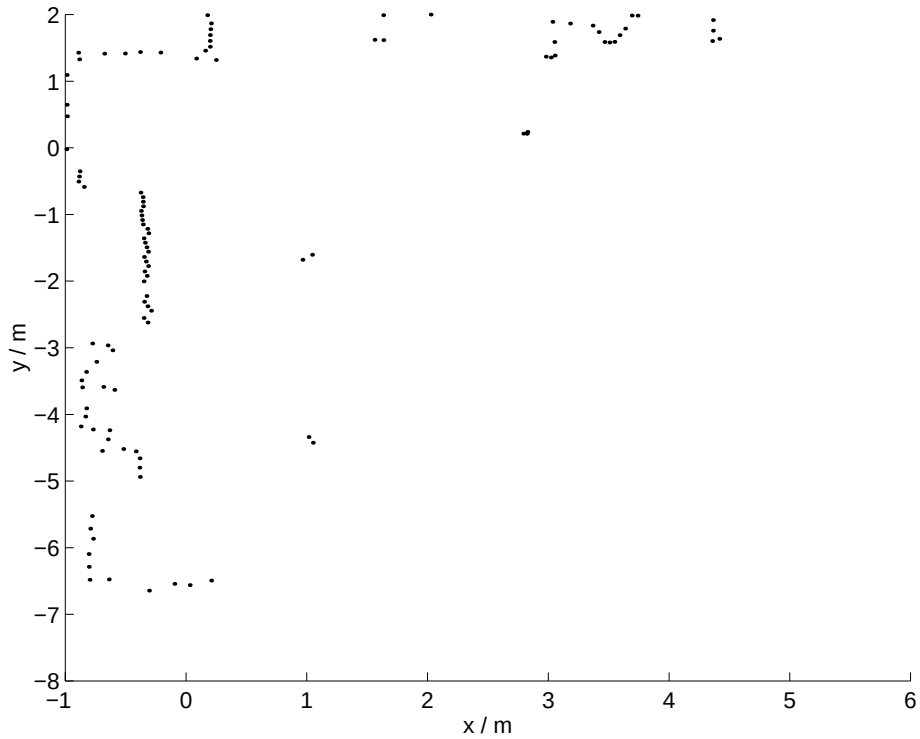

Figure 5.8: A typical scan of the laser rangefinder. The orientation of this scan corresponds to the orientation of Figure 5.1, and the wash basins produce the irregular contour near (-0.75m,-3.75m).

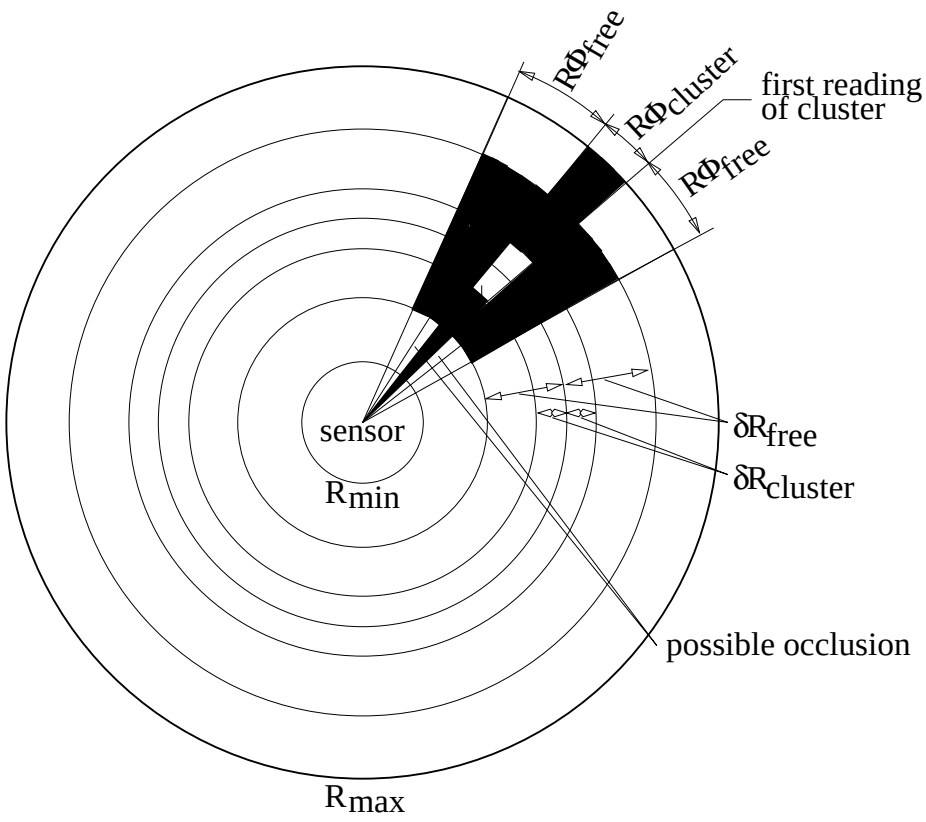

Figure 5.9: Dimensions of a valid cluster.

around the cluster.

Assume that the first return R_i from a feature occurres as the ith range reading in a scan. A subsequent return R_j, still of the same feature, must be within a small error $\delta R_{cluster}$, due to sensor inaccuracy and finite feature size, of R_i. This defines a valid region for the range of subsequent readings of the same feature, as shown in Figure 5.9. Range limits R_{min} and R_{max} are also defined, beyond which the data is considered unreliable and clusters are not considered.

The angular spread of a cluster will depend linearly on the range of the cluster. The further away a feature is from the sensor, the smaller must the difference between the bearing of the beginning of the feature and the bearing of the end of the feature. This can be captured by a threshold on the product of R_i and the number of returns from a feature, since R_i must be a close to the range of the feature and each return represents one degree of sweep. This threshold will be denoted $R\Phi_{cluster}$.

If a return R_j is such that $|R_j - R_i| > \delta R_{cluster}$ or $(j-i)R_i > R\Phi_{cluster}$, R_j cannot belong to the same small cluster as R_i. R_j either indicates the end of the cluster or indicates that the spread of the cluster is too large and the cluster should be ignored.

If R_j is just outside a threshold, the free space around the cluster will be too small to classify

the cluster as isolated. If R_j is separated by substantial free space from the cluster, the cluster is potentially valid. A threshold δR_{free} defines how much free space is required in the range and $R\Phi_{free}$ defines how many readings before the start of the cluster and after the end of the cluster must pass the δR_{free} threshold.

The thresholds $\delta R_{cluster}$ and $R\Phi_{cluster}$ define an upper limit to the spread of a valid cluster and the thresholds δR_{free} and $R\Phi_{free}$ define a lower limit to the amount of free space around the cluster. Since the number of readings expected in a cluster is small, the first reading that falls outside the $\delta R_{cluster}$ and $R\Phi_{cluster}$ thresholds ends the cluster. Spurious readings within a cluster are not considered. A cluster is also rejected if the number of returns in the cluster are below a certain threshold N_{min}. This threshold is used to eliminate spurious readings registering as valid clusters.

The final areas represented by the thresholds are shown in Figure 5.9. It was found that large features could produce a small cluster if they are partially occluded. To eliminate this undesired phenomenon, the beginning and the end of a potential cluster is checked for occlusion. A fixed number N_{ocl} of readings are checked ahead and after the potential cluster. If any of the checked readings has a range smaller than R_i, the range of the first return in the cluster, the cluster was possibly due to occlusion and is disregarded. R_i is chosen for convenience; if the free space test is successful, any of the returns within the cluster would serve as the reference for the occlusion test.

Once a valid cluster is identified, the range and bearing is calculated as the mean range and mean bearing of all the returns within the cluster. This is a reasonably good approximation to the true range and bearing of the feature since the cluster produced by a feature is required to be small. A flow diagram of the extraction algorithm is given in Figures 5.10 and 5.11. Three example scans are shown in Figures 5.12, 5.14 and 5.16. The extracted range and bearings, indicated by dashed lines, are shown in Figures 5.13, 5.15 and 5.17.

The actual thresholds, obtained by maximising successful extractions while insisting on no erroneous extractions, are given in the table below. Using the same performance criteria, N_{min} and N_{ocl} were deduced as 2 and 1 respectively.

R_{min}	R_{max}	$\delta R_{cluster}$	δR_{free}	$R\Phi_{cluster}$	$R\Phi_{free}$
0.01 m	6.00 m	0.20 m	0.40 m	$\frac{180}{\pi}\delta R_{cluster}$	$\frac{180}{\pi}\delta R_{free}$

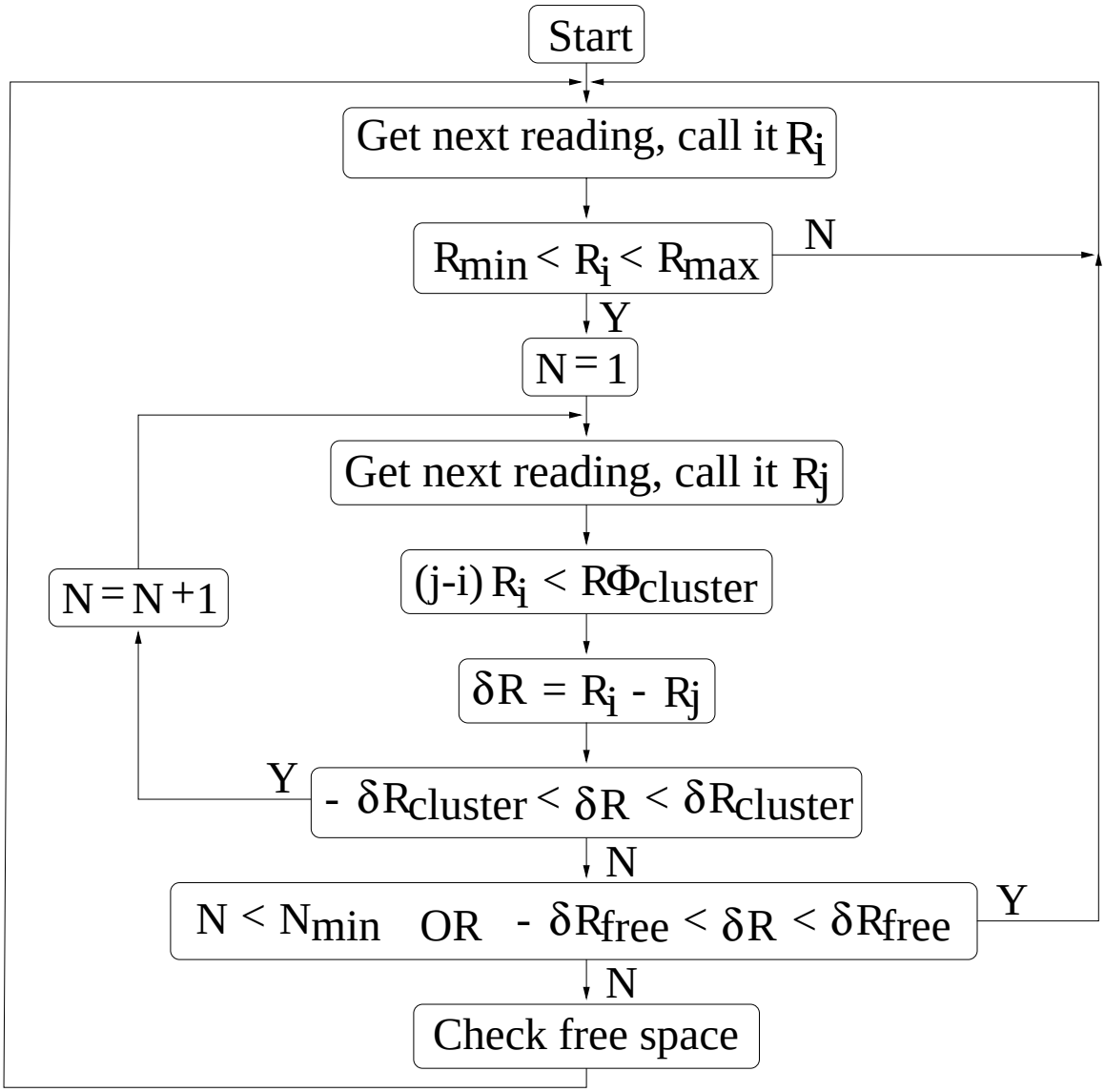

Figure 5.10: Extracting the range and bearing, Flow Diagram I.

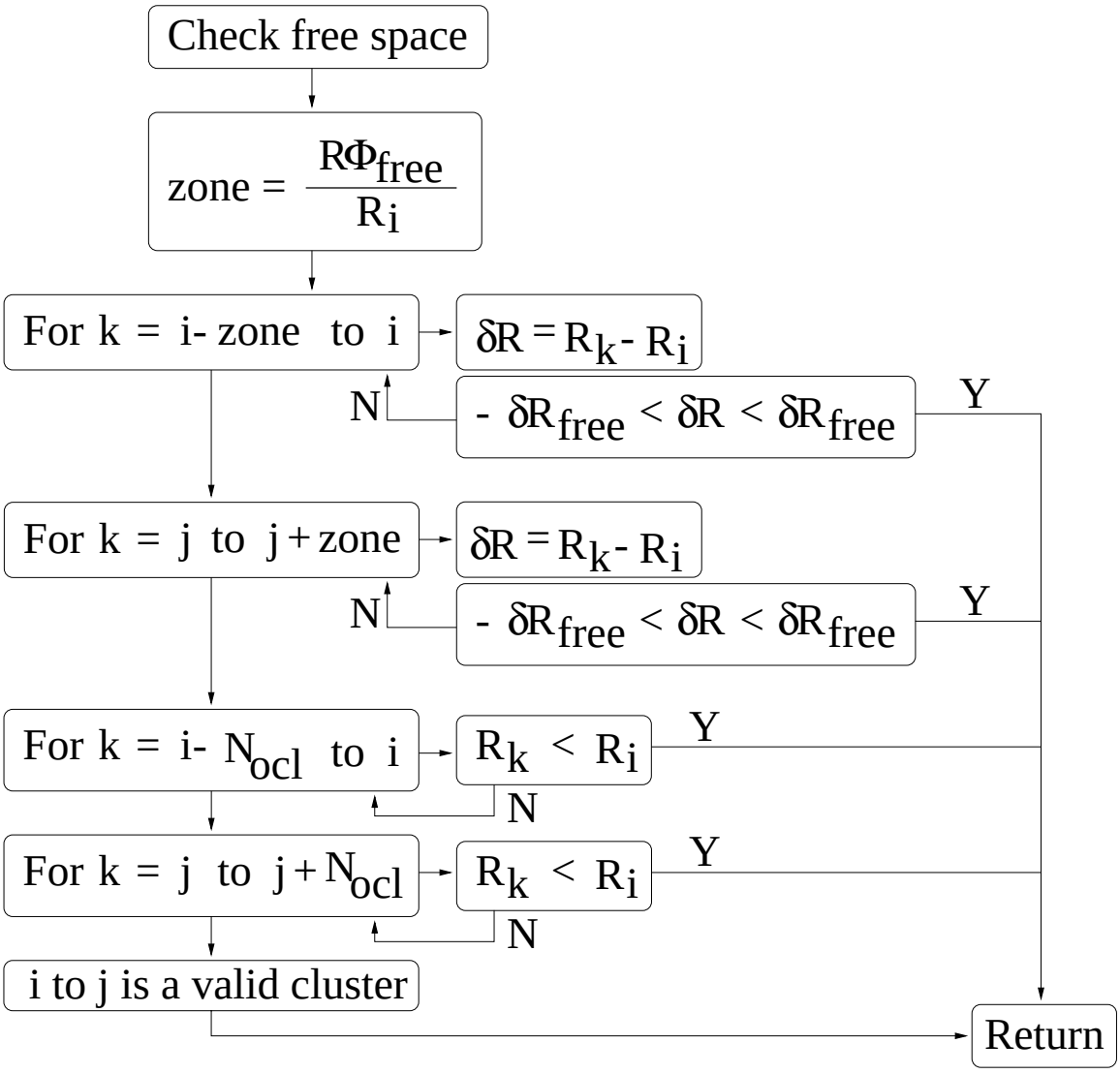

Figure 5.11: Extracting the range and bearing, Flow Diagram II.

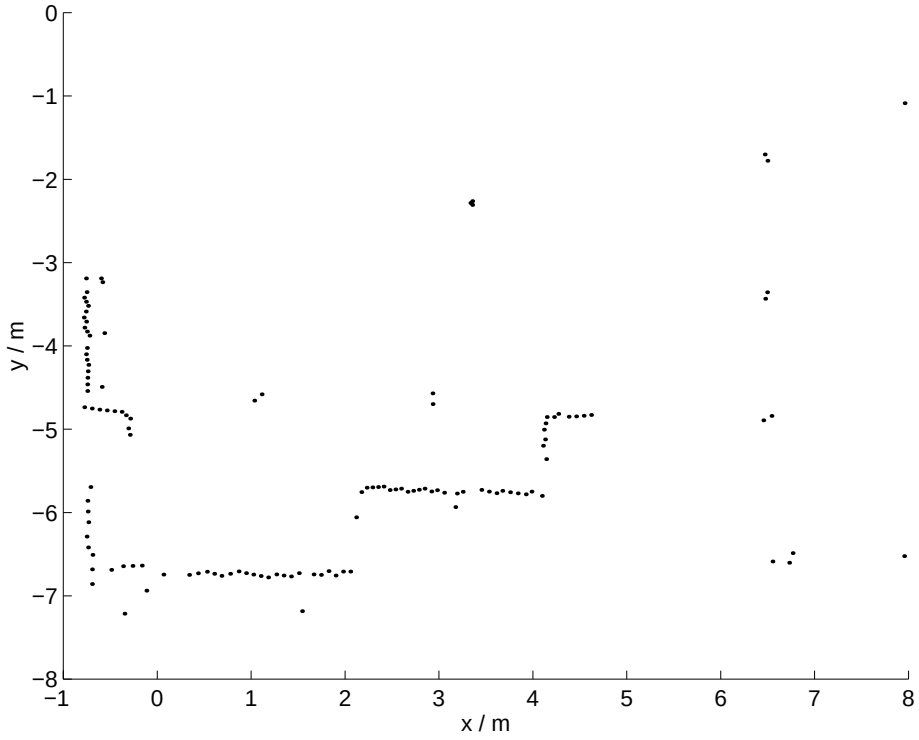

Figure 5.12: Laser data at Timestep 728.

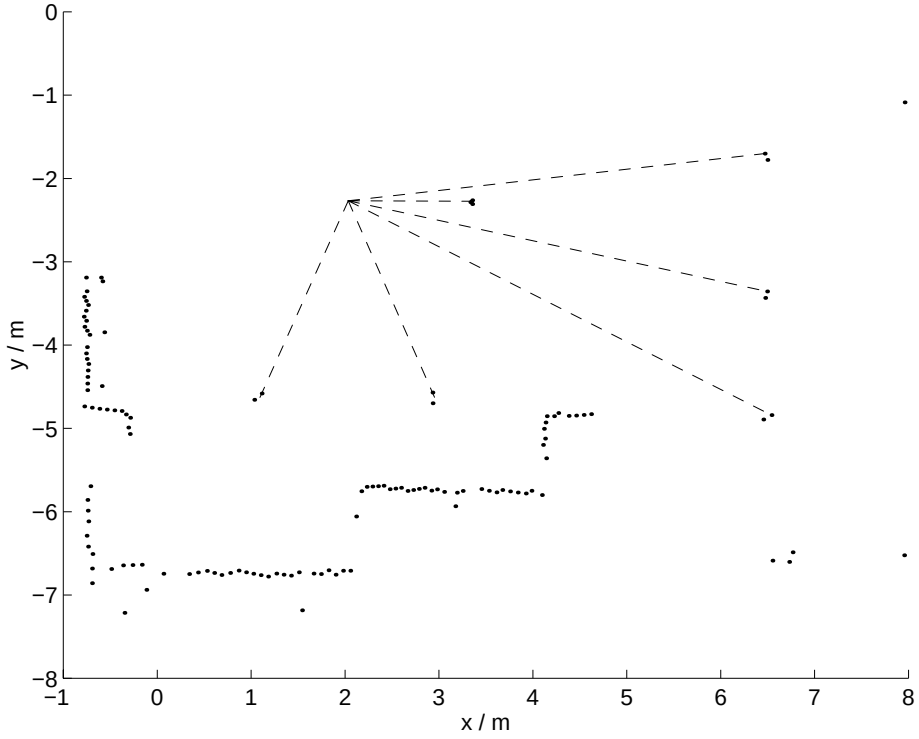

Figure 5.13: Extracted clusters at Timestep 728.

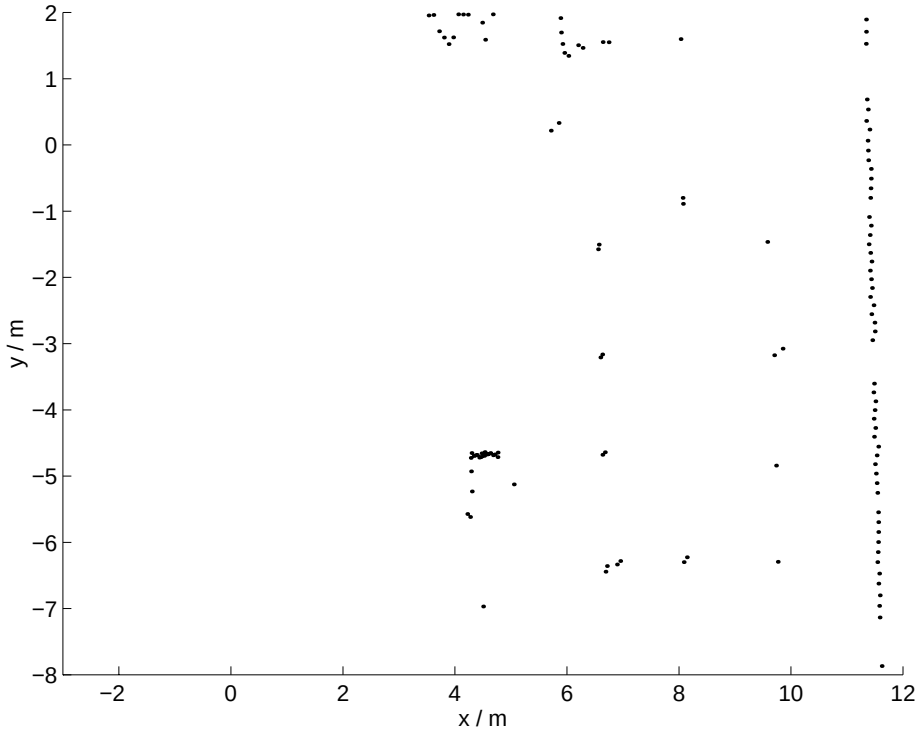

Figure 5.14: Laser data at Timestep 1351.

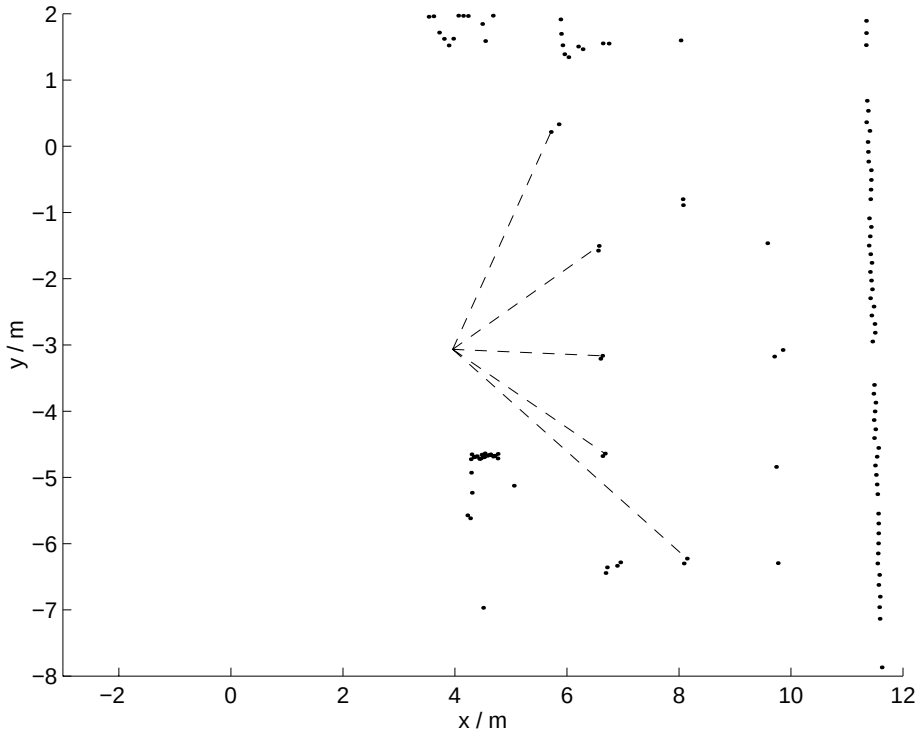

Figure 5.15: Extracted clusters at Timestep 1351.

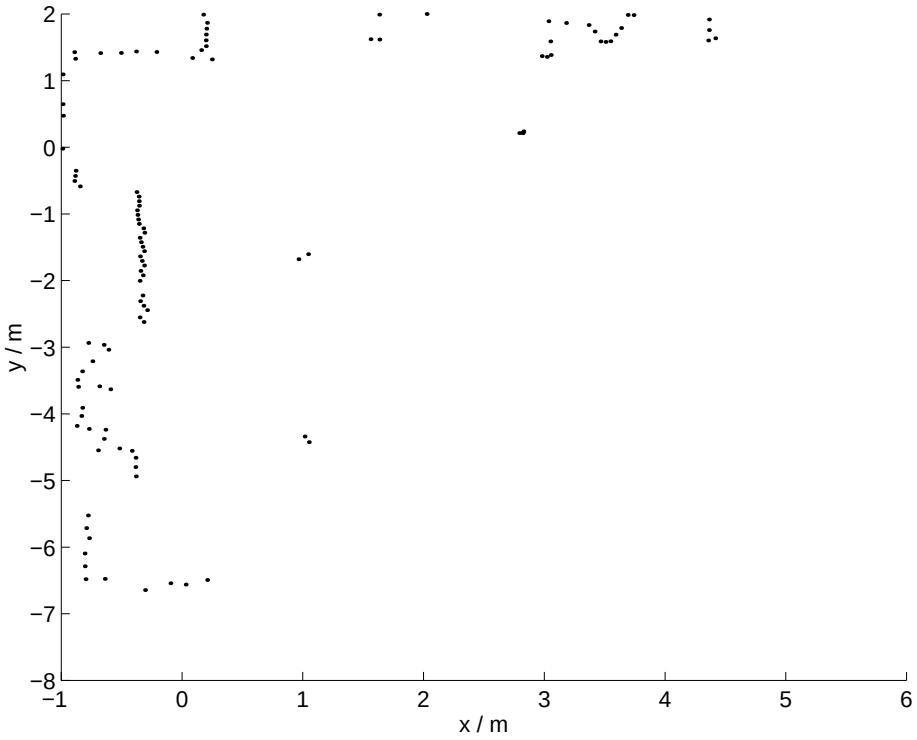

Figure 5.16: Laser data at Timestep 2170.

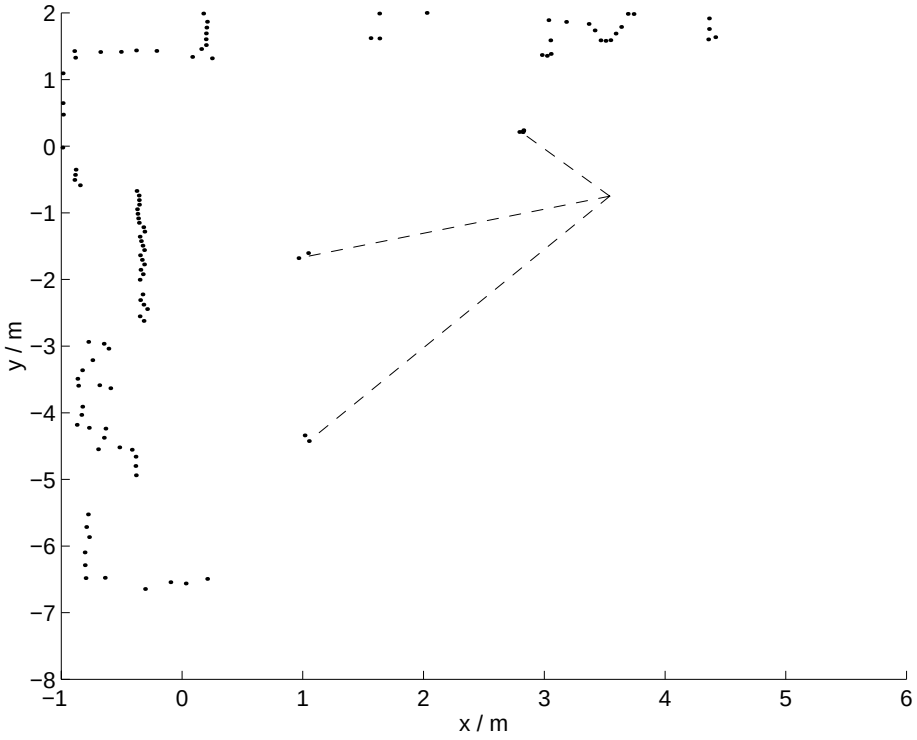

Figure 5.17: Extracted clusters at Timestep 2170.

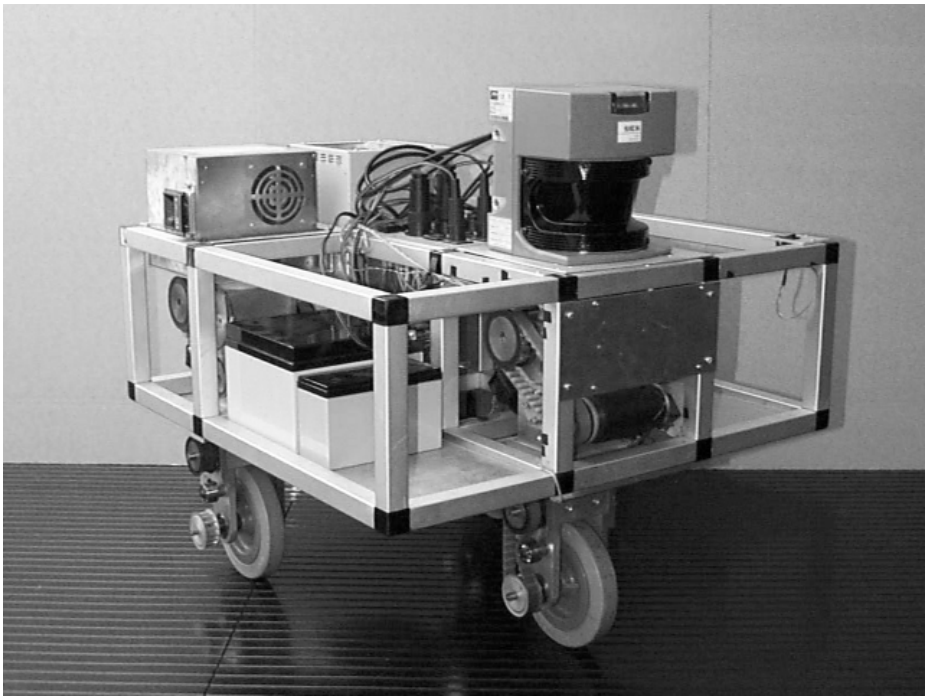

Figure 5.18: The experimental vehicle with the sensor mounted.

5.4 The Vehicle

A three-wheeled indoor experimental robot was used as the test vehicle. The front wheel of the vehicle is driven and steered and the two back wheels can roll freely. The vehicle model for this robot was derived in Section 2.3. Encoders on the front wheel supply measurements of angular velocity and orientation of the wheel. These are used to form predictions of the motion of the vehicle according to the vehicle model. The encoder readings and the sensor observations were logged continuously in timesteps of 32ms. This allowed all map building algorithms to be tested under identical conditions.

Figure 5.19: The experimental vehicle in the test environment with the glass wall in the background.

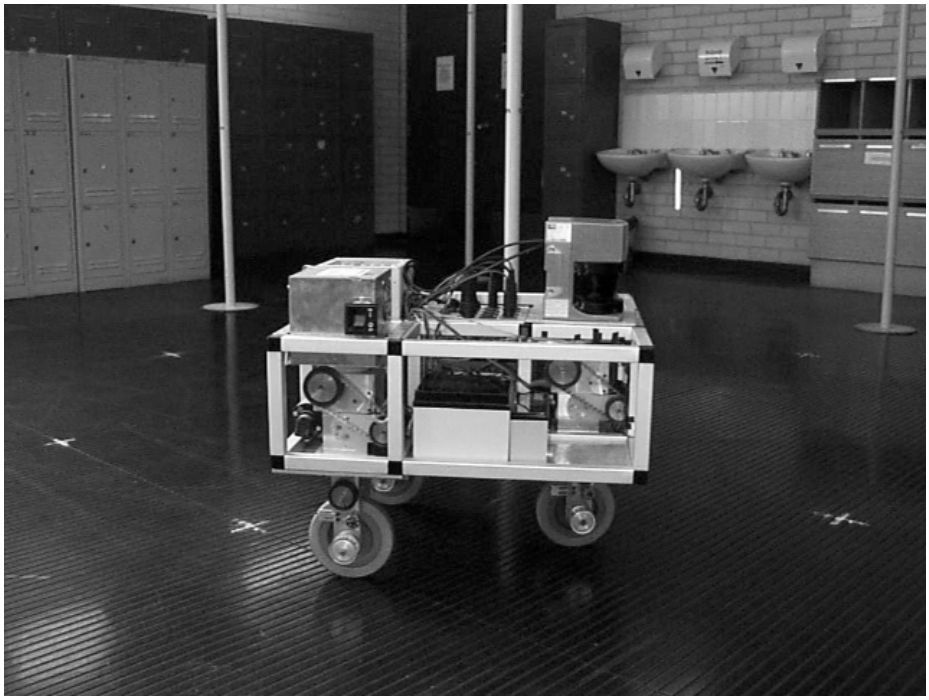

Figure 5.20: The experimental vehicle in the test environment with the wash basins in the background.

5.5 Preliminary Investigations

In order to better understand the performance of the simultaneous localisation and map building filters developed in this thesis, it is necessary to briefly examine the performance of the vehicle and observation models under more controlled conditions. Certain characteristics of the map building results will be traced to the characteristics of the vehicle and the observation model, which are identified in this section. The preliminary investigations are also useful to identify possible experimental design faults, which would otherwise be identified as faults relating to the map building filters.

The first investigation, Section 5.5.1, examines the performance of the vehicle model by developing a Prediction-Only Kalman (PREK) filter. The investigation is then extended in Section 5.5.2 to include the observation model by developing a Known-Features Kalman (KNOK) filter, which localises the vehicle using the feature observations and knowledge of the surveyed feature positions.

5.5.1 The Predictions

Figures 5.21 to 5.30 show the evolution of the vehicle state estimate based only on predictions, as computed by the PREK filter. As expected, the standard deviations of the vehicle estimate diverge during the operation of the PREK filter. This can be seen in Figures 5.21 and 5.22. However, further observations can be made.

The standard deviation of the orientation estimate, Figure 5.21, increases monotonically, but a manoeuvre increases the rate at which the standard deviation increases.

The standard deviation of the x and y position estimates, shown in Figure 5.22, increase as well. The increase in the standard deviations can also be seen when comparing Figures 5.26 and 5.30. However, the increase is not monotonic. This phenomenon is associated with the rotation of the covariance ellipse, which can be seen by comparing Figures 5.27 and 5.28. The rotation of the covariance ellipse can also be seen by considering the correlation coefficient r_{xy},

$$r_{xy} = \frac{\sigma_{xy}^2}{\sqrt{\sigma_{xx}^2 \sigma_{yy}^2}} \tag{5.1}$$

where σ_{xx}^2 is the variance of the estimate in the x coordinate, σ_{yy}^2 the variance in the y coordinate and σ_{xy} is the covariance between the estimates. The correlation coefficient is plotted in Figure 5.23. The periodic change in the coefficient confirms that the covariance ellipse is rotating, and a rotating covariance ellipse is clearly associated with a non-monotonic change in the standard deviation of the x and y position estimates.

Non-monotonic behaviour is also apparent in the determinant of the covariance of the position

estimate, shown in Figure 5.24. The determinant of a covariance matrix is a measure of the volume associated with the covariance, and indicates the total uncertainty in the estimate, as discussed in Section 3.6. The non-monotonic behaviour suggests that the uncertainty in the position estimate may decrease at times. However, the total uncertainty in the vehicle state estimate cannot decrease, since no information is ever added to the vehicle estimate in the PREK filter. Therefore, the decrease in the uncertainty of the position estimate must be offset by an increase in the uncertainty in the orientation uncertainty, leading to a monotonically increasing determinant of the full state estimate covariance, as shown in Figure 5.25

In Figures 5.26 to 5.30, the 2σ bound of the vehicle position estimate, the covariance ellipse, is plotted for different timesteps. It is shown in relation to the true position of the features and is marked with a triangle. The orientation of the triangle follows the mean of the estimate of the vehicle orientation, but the shape and size are fixed.

To accurately verify the predictions, the true position and orientation of the vehicle during the trial would be needed. However, this information was not available. Further investigations relating to the vehicle model and also incorporating the observation model are made in Section 5.5.2.

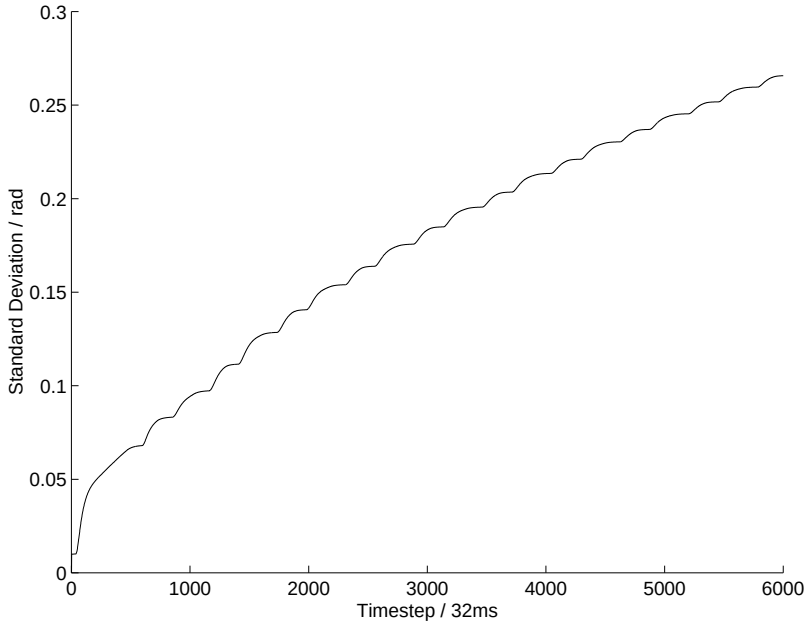

Figure 5.21: The standard deviation of the vehicle orientation estimate is seen to be monotonically increasing. The periods of greater gradient of increase occur during manoeuvres. Both of these are expected characteristics of a PREK filter.

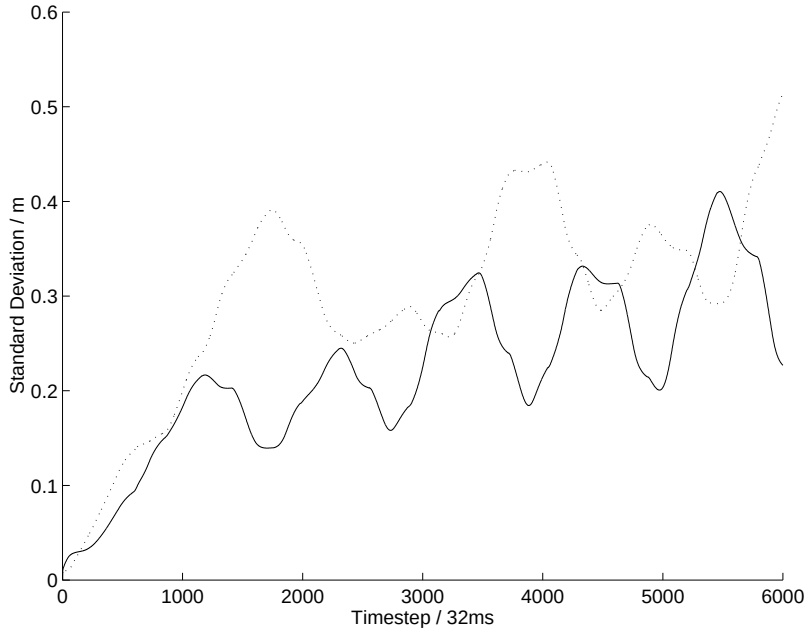

Figure 5.22: The standard deviation of the estimate in the x coordinate (solid line) and the y coordinate (dotted line) are seen to be increasing and oscillating out of phase, indicating the change in shape of the position covariance ellipse.

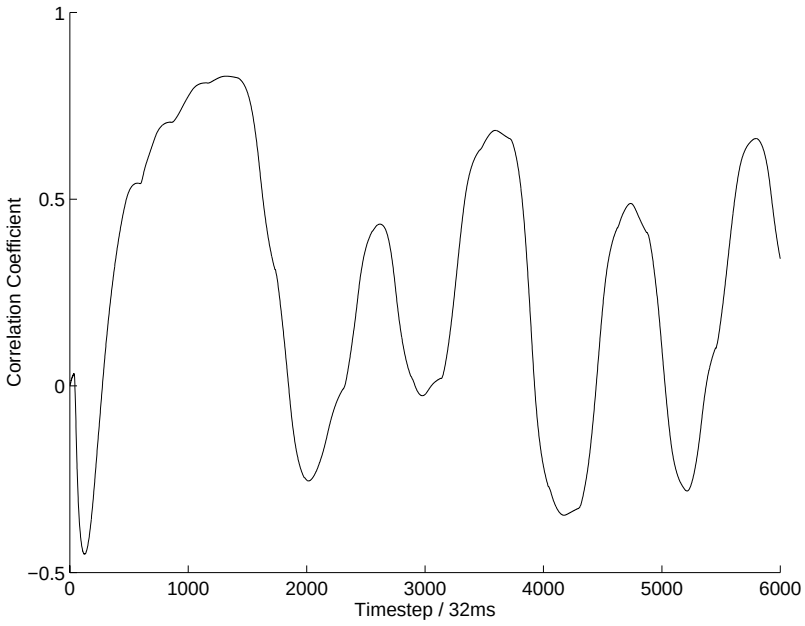

Figure 5.23: The correlation coefficient of the variance of the predicted position is seen to be oscillating, passing through zero several times. This shows that the major axis of the covariance ellipse is changing its orientation.

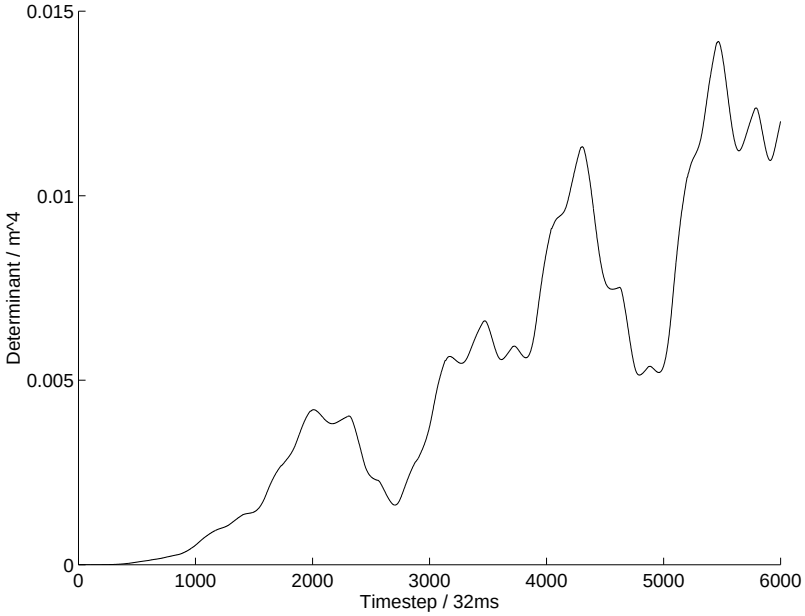

Figure 5.24: The determinant of the position covariance matrix is also increasing. However, the increase is non-monotonic. This occurs because the position covariance ellipse may decrease in area, signifying an increase in the certainty of the vehicle position.

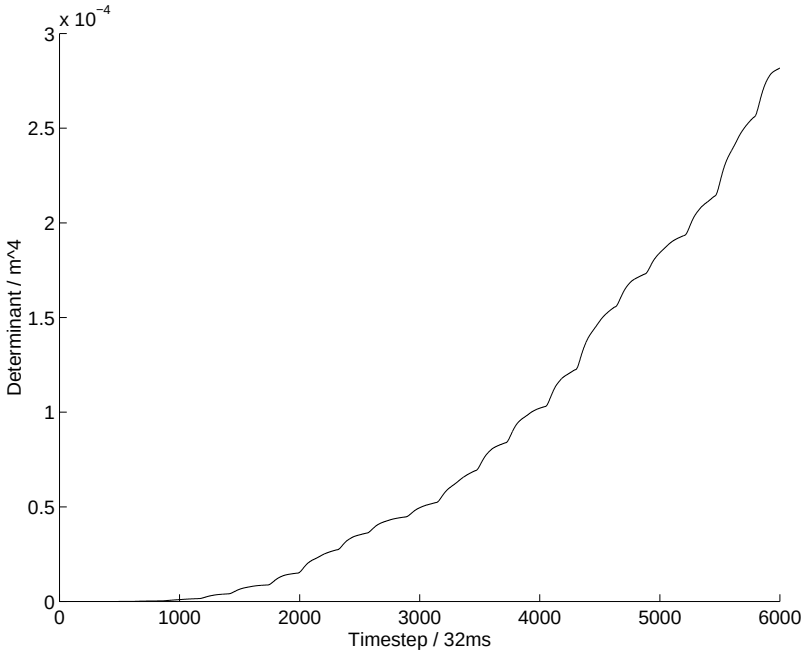

Figure 5.25: The determinant of the state covariance matrix is monotonically increasing, showing that the total uncertainty in the state estimate is increasing.

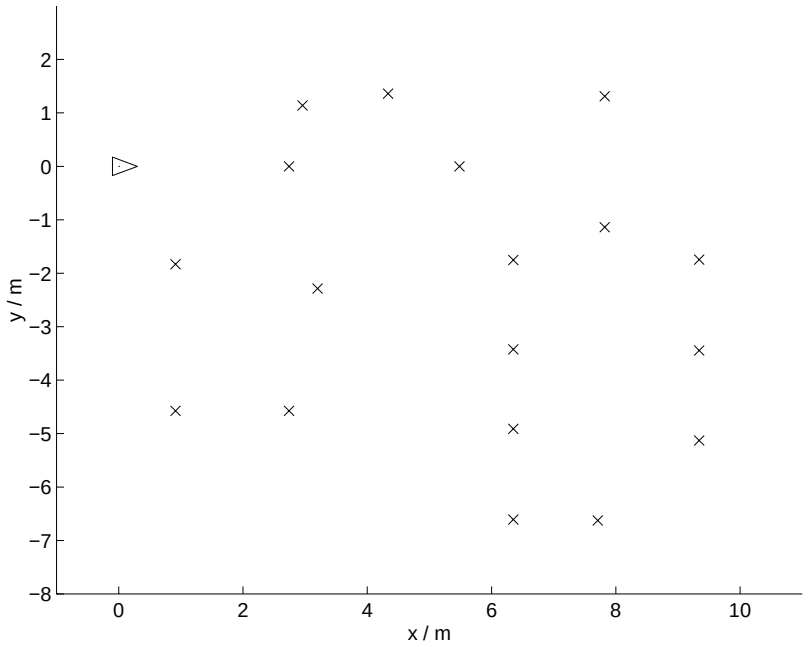

Figure 5.26: Timestep 1, the initial vehicle position estimate and the true position of the features is shown. The covariance ellipse is too small to be seen.

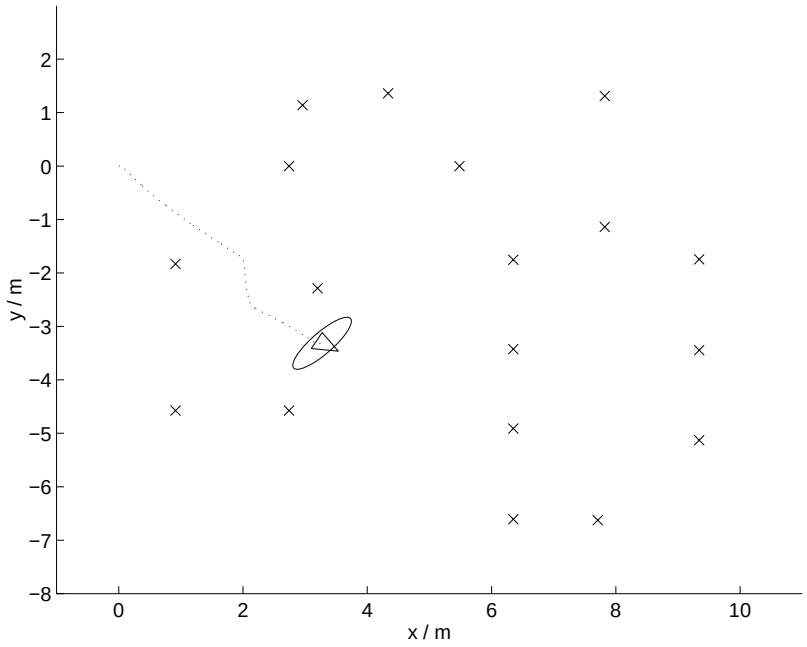

Figure 5.27: Timestep 1199, the covariance ellipse has increased in area. The dotted line is the trace of the vehicle path as predicted by the vehicle model.

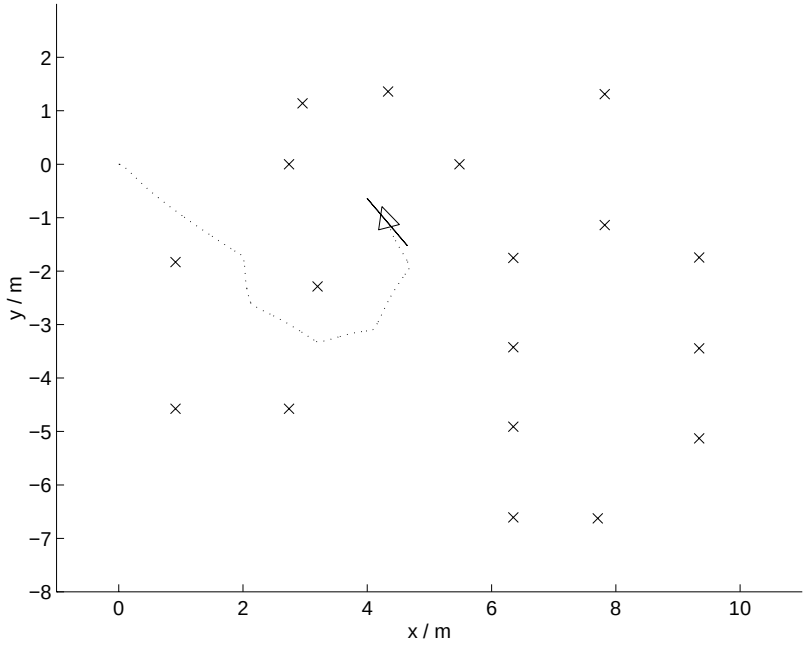

Figure 5.28: Timestep 2000, the shape and orientation of the covariance ellipse indicates a high negative correlation between the x and the y coordinate. No satisfactory explanation of the high correlation was found, but no implementation errors were found either.

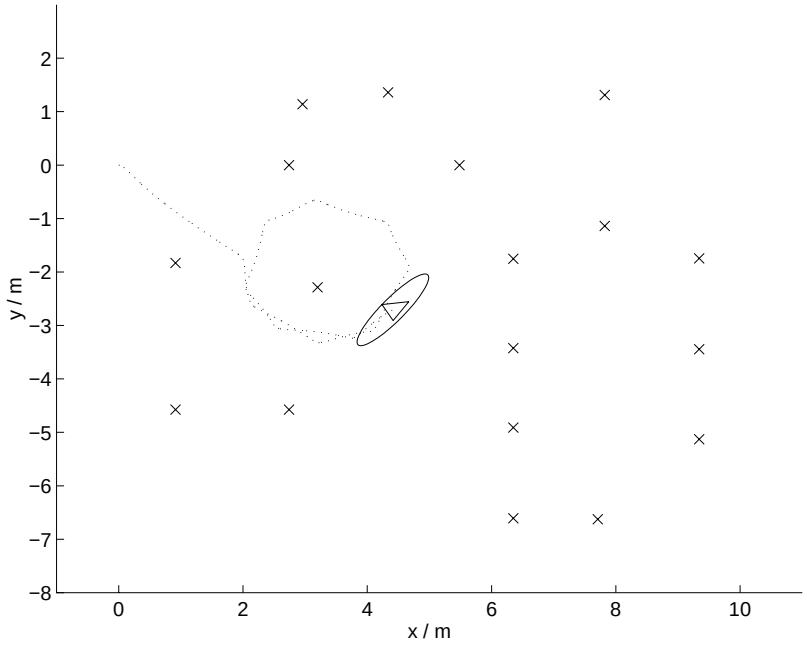

Figure 5.29: Timestep 3687, the standard deviations of the estimate have increased further. The slight rotation of the robot path away from the desired path was observed during the logging process, and crude tuning of the PREK was used to approximate this rotation.

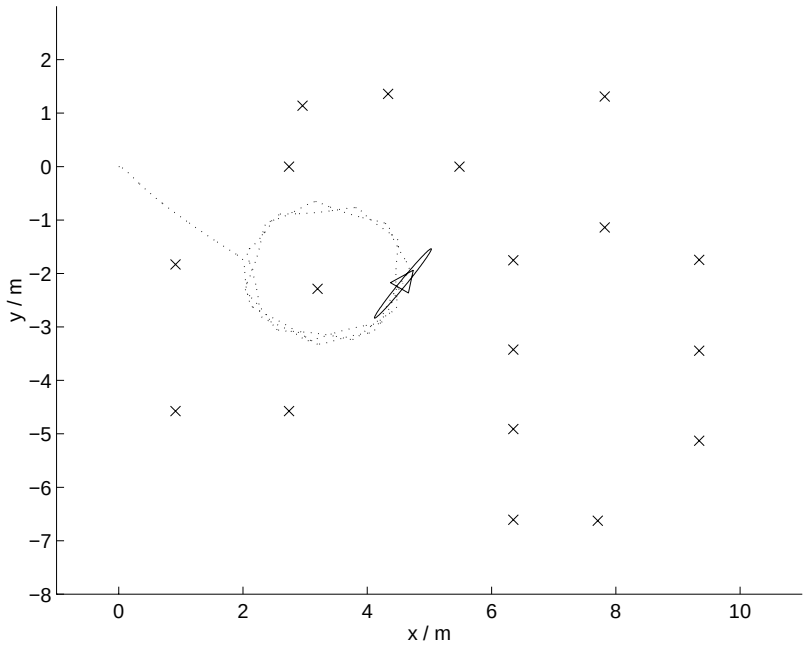

Figure 5.30: Timestep 6000, the estimates after two and a half circuits (equivalent to just over three minutes of actual vehicle operation) are shown.

5.5.2 Using the Surveyed Feature Positions

The best test of the overall experimental design is to survey the position of the features and use the known features as beacons in a Kalman filter to localise the vehicle. The Kalman filter based on known features is referred to as a KNOK filter. Localisation based on known feature positions is a well understood problem and the application of the Kalman filter to the solution of this problem is extensively documented in the literature [28]. The test implicitly verifies the operation of the vehicle model, the sensor model and the feature extraction algorithm, under the same conditions as would occur during a map building trial. The test cannot guarantee that the experimental design is perfect, but it can indicate the presence of problems not specifically related to the map building algorithms, and thereby avoid some false conclusions.

Figures 5.31 to 5.36 show the evolution of the vehicle state estimate for a KNOK filter. Comparing Figures 5.31 – 5.33 with Figures 5.21 – 5.24, it can be seen that the KNOK filter prevents the vehicle state estimate standard deviations from diverging.

The standard deviation of the vehicle orientation is very sensitive to manoeuvres; this can be seen by the inability of updates to decrease the standard deviation during a manoeuvre, as shown in Figure 5.31. The change in the shape of the position covariance ellipse during the experiment is again evident in Figures 5.32 and 5.33. Figure 5.32 also shows that the standard deviation of the x and y position estimates move out of phase, a phenomenon observed already in the PREK filter.

Oscillations with a period of about 2250 timesteps can be seen in the standard deviation of the orientation estimate, Figure 5.31, and the determinant of the position covariance, Figure 5.33, with the peaks occurring approximately at timesteps 2250 and 4800. Corresponding peaks can also be seen in the standard deviation of the x coordinate estimate, although they are not so prominent.

A period of 2250 timesteps is equivalent to one circuit, and therefore suggests that some regions along the vehicle path are better suited for updates, probably due to the particular feature constellation. The particular feature constellation leads to better information propagation from the features to the vehicle. The results suggest that the constellation has a greater effect on the standard deviation of the orientation estimate than on the standard deviation of the position estimate.

Finally, the innovation plots shown in Figure 5.34 and 5.35 indicate that the KNOK filter is not operating optimally. Correlations and biases are evident, as is the fact that the both innovation standard deviations are too conservative.

It was decided that all filters employed in the tests should use the same filter parameters in order to provide a comparable basis for the experiment. Tuning the two filters discussed so far and the map building filters examined below is a formidable task. Optimal filter performance is difficult to achieved due to unmodelled effects such as the finite size of the features; the errors injected

state	initial mean	initial standard deviation
x_v	0.0 m	0.01 m
y_v	0.0 m	0.01 m
ϕ_v	0.0 radians	0.01 radians

Table 5.1: Variance of the initial vehicle state estimate.

noise source	standard deviation
velocity V	0.10 m/s
steer angle γ	0.20 radians
range r	0.10 m
bearing θ	0.05 radians

Table 5.2: Variance of the process and observation noise.

by the feature extraction algorithm; the grooves in the corridor floor, as shown in Figure 5.5; and imperfections in the vehicle frame and wheels. All unmodelled effects invalidate the assumptions of the Kalman filter, leading to the poor performance. A more detailed exposition on models and tuning filters can be found in [46].

The final values of the filter parameters are given in the Tables 5.1 and 5.2.

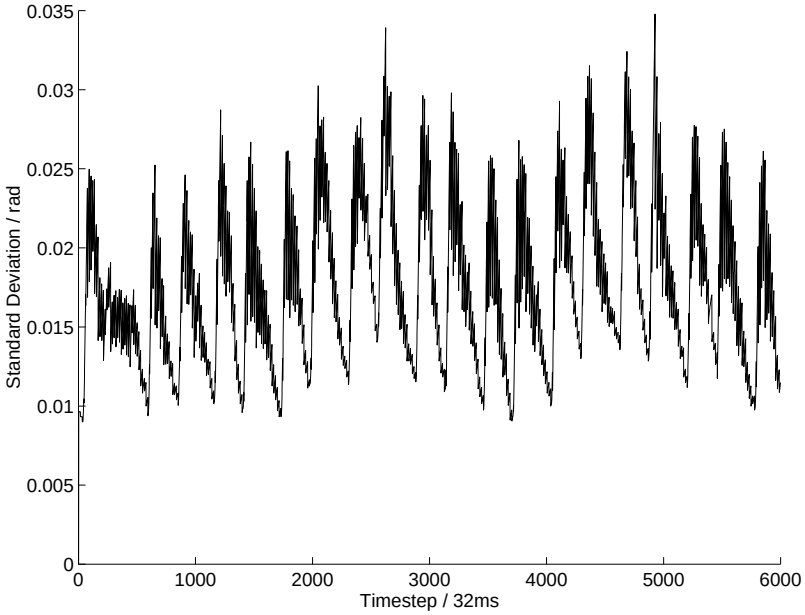

Figure 5.31: The standard deviation of the vehicle orientation estimate is oscillating, with the peaks in the deviation occur during manoeuvres. However, it exhibit no long-term divergence. A lower frequency oscillation, with a period of about 2250 timesteps, is also evident and is explained in the text.

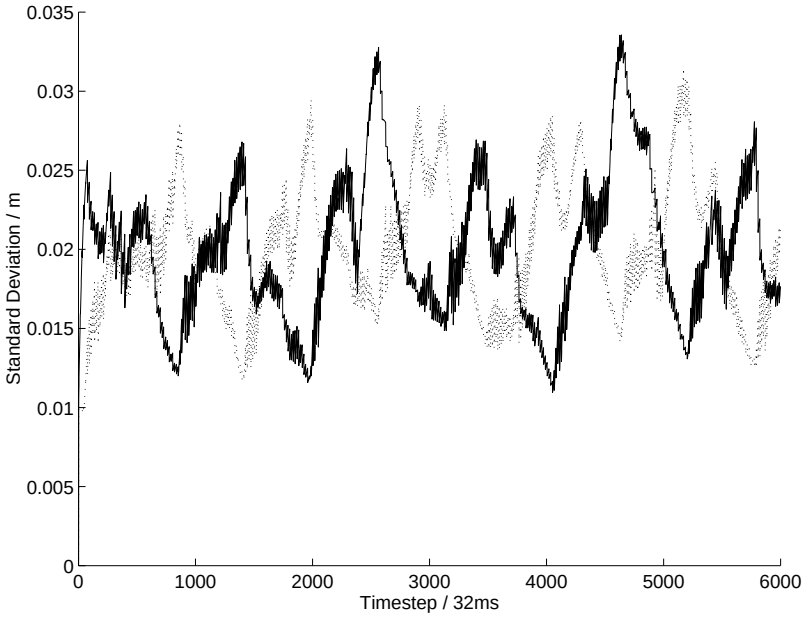

Figure 5.32: The standard deviation of the estimate in the x coordinate (solid line) and the y coordinate (dotted line) are oscillating out of phase, indicating a change in shape of the position covariance ellipse.

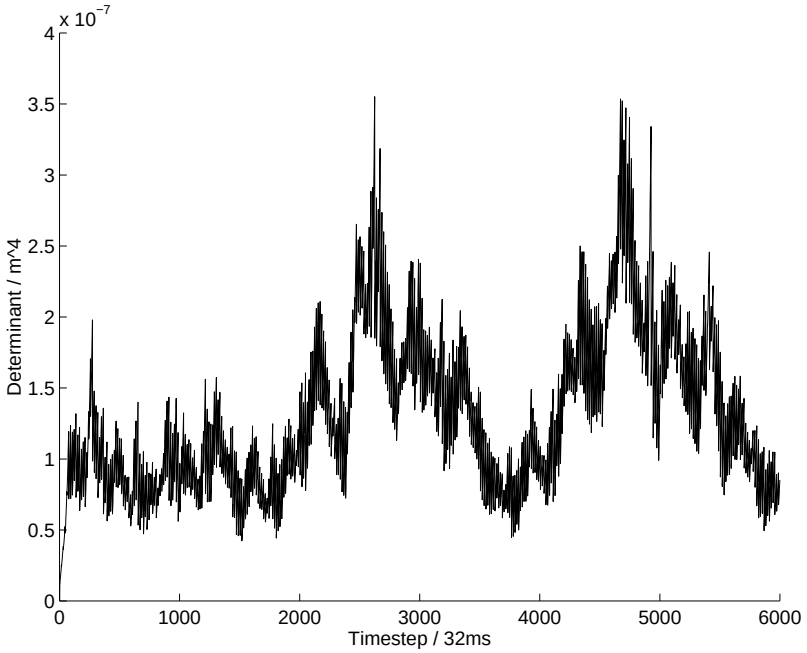

Figure 5.33: The determinant of the position covariance matrix has a low frequency component with a period of about 2250 timesteps, in phase with the oscillation of the standard deviation of the vehicle orientation estimate. The determinant is also non-diverging.

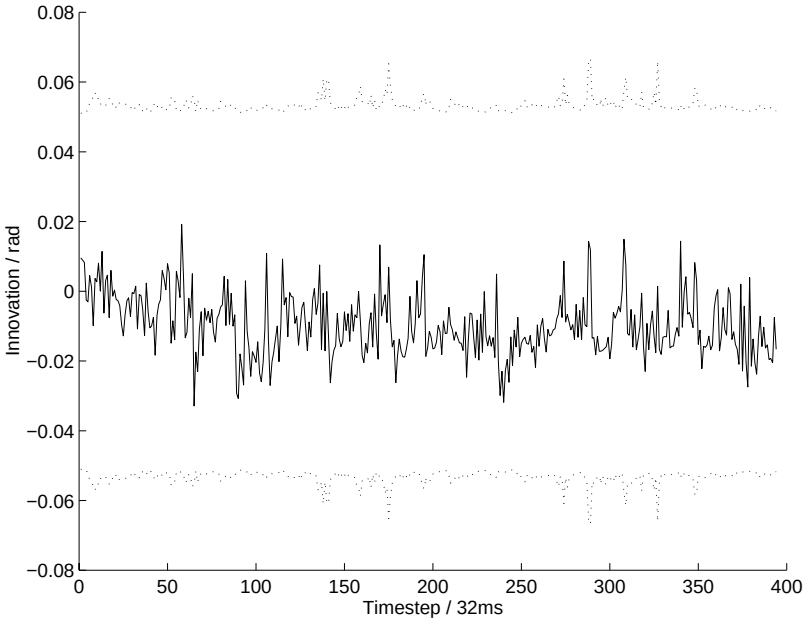

Figure 5.34: The innovation in the bearing is evidently biased and correlated, and its standard deviation is over estimated.

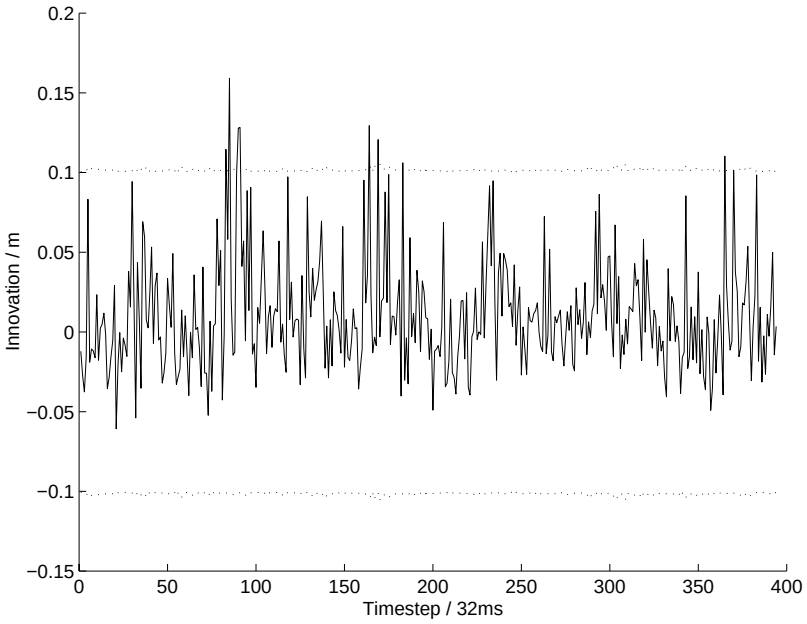

Figure 5.35: The innovation in the range, like the innovation in the bearing, is evidently biased and correlated, and its standard deviation is also over estimated.

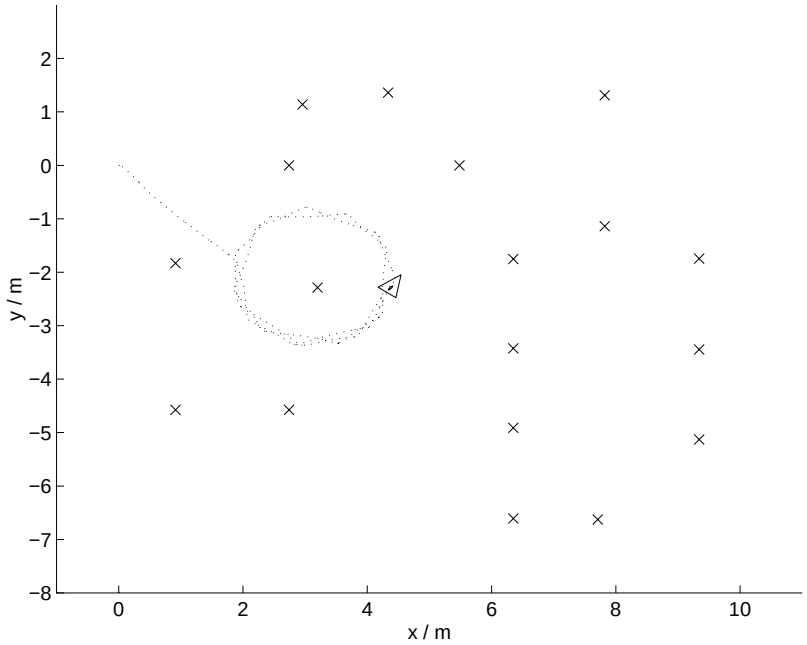

Figure 5.36: Timestep 6000, the rotation in the vehicle path is again evident, but the covariance ellipse is smaller than in the PREK filter.

5.6 The Map Augmented Kalman Filter

This section examines the experimental results of the MAK filter, and uses the insights gained in Section 5.5 to interpret the results. Section 5.6.1 examines the operation of the MAK filter, and compares it to the operation of PREK and KNOK filters. The evolution of the map constructed by the MAK filter is examined in Section 5.6.2 and the conclusions are summarised in Section 5.6.3.

5.6.1 The Operation of the Map Augmented Kalman Filter

The MAK filter was run over exactly the same data set as the PREK and the KNOK filters. The first point to note is that the standard deviation of the vehicle estimates is always larger in a MAK than in a comparable KNOK filter, as seen by comparing Figures 5.37 and 5.38 to Figures 5.31 and 5.32. This is expected since the information contained in the surveyed feature positions is not available to the MAK filter. The MAK filter has to construct estimates of the feature positions as the vehicle observes them, and thus observation information must be "spread" amongst a larger number of states.

The construction of the estimates by the vehicle results in a definite pattern in the map. As can be seen in Figures 5.42 to 5.51, the features initialised by the vehicle while it is near the origin, the starting point of the vehicle, have smaller covariances than features initialised when it is further away. This is due to the accumulation of process noise by the vehicle as it travels to remote features, and has a significant effect on the operation of the MAK filter. This will be discussed later in this section.

The most striking similarity between the operation of the MAK and the KNOK filter is the behaviour of the standard deviation of the orientation estimate, Figures 5.31 and 5.37. In both filters the standard deviations exhibit a very strong sensitivity to manoeuvres, leading to a high frequency oscillation in phase with the manoeuvres. A further similarity between the standard deviations is the 2250 period oscillation, identified in the KNOK filter as a result of particular feature constellations. The oscillations in the two filters are in phase. The only significant difference in the standard deviations is in their mean values; a mean of 0.025rad for the MAK filter compared with a mean of 0.023rad for the KNOK filter. The main factors affecting the behaviour of the orientation estimate in the MAK and the KNOK filters seem to be the same.

A difference between the MAK and the KNOK filter can be found in the phase shift of the 2250 period harmonic in the determinants. Whereas in the KNOK filter the peaks occurred at timesteps 2550 and 4800, the peaks in the MAK filter occurred at timesteps 1250, 3500 and 5750. This can be seen by comparing Figures 5.33 and 5.39. The difference suggests that, in the case of the position estimate, the main factors affecting the behaviour of the MAK and the KNOK filters may be quite

different. However, in both filters the factors are linked to the position of the vehicle along its path, as suggested by the 2250 period oscillations.

The second difference concerns the extent to which the 2250 period oscillation is visible. Examining the determinants in Figures 5.39 and 5.33 again, it can be seen that the amplitude of the 'noise' on the harmonic in the case of the MAK filter is well below 10% of the amplitude of the harmonic, whereas in the case of the KNOK filter it is about 50%. Therefore the 2250 oscillation is more dominant in the MAK filter than in the KNOK filter.

Further, the oscillation develops at timestep 0 in the MAK filter, but only develops after approximately timestep 1300 in the KNOK filter. There is also a significant decay in the amplitude of the determinant in the MAK filter, with a peak of $6.9 \times 10^{-6} m^4$ at timestep 1250, but a peak of only $4.8 \times 10^{-6} m^4$ at timestep 3500. The 2250 period oscillation is very dominant in the standard deviation of the position estimates in the MAK filter, Figure 5.38, but is hardly visible in the KNOK filter, Figure 5.32.

The second difference is therefore a more dominant 2250 period harmonic in the position estimate of the vehicle, suggesting that, in the case of the position estimate, the vehicle's position along its path has a greater effect on the MAK filter than on the KNOK filter.

A possible explanation to the differences in the KNOK and the MAK filter is that the vehicle orientation estimate is sensitive to manoeuvres and the constellation of features observed by the vehicle, whereas the vehicle position estimate is primarily sensitive to the confidence in the position estimate of the features observed by the vehicle. This hypothesis will now be used to explain the observations, starting with the vehicle orientation estimate.

The constellation of the features and the manoeuvres of the vehicle were the same for the MAK and the KNOK filter. Therefore, the standard deviation of the orientation estimates are comparable, as indicated by Figures 5.31 and 5.37. The difference in the confidence of the feature positions, known in the case of the KNOK filter but estimated in the case of the MAK filter, leads only to a very small difference, a mean of 0.025rad compared with a mean of 0.023rad, in the standard deviation of the orientation estimate.

The suggested hypothesis can also be used to explain observations relating to the position estimate. It was suggested that the vehicle position estimate is primarily sensitive to the confidence in the feature estimates. Therefore, the higher confidence of the MAK filter in the estimates of features near the origin, indicated by the smaller covariance ellipses in Figures 5.42 to 5.51, results in better position updates when the vehicle is in that area. This occurs approximately at timesteps 2700 and 4950, which in turn leads to the local minimums in the standard deviation of the vehicle position estimate, Figure 5.38, and the determinant, Figure 5.39, at those timesteps. The local maximums occur when the vehicle is at its furthest point, at approximately timesteps 1250 and 3500. Each

time the vehicle revisits the furthest point, the confidence in the features has increased due to the updates, and therefore the peak in the determinant and the peak in the standard deviation of the position estimates decay in the case of the MAK filter.

The phase shift between the standard deviation of the x and y position estimates has already been observed in the PREK and the KNOK filter, and is therefore probably an artifact of the predictions and not a result of the simultaneous localisation and map building.

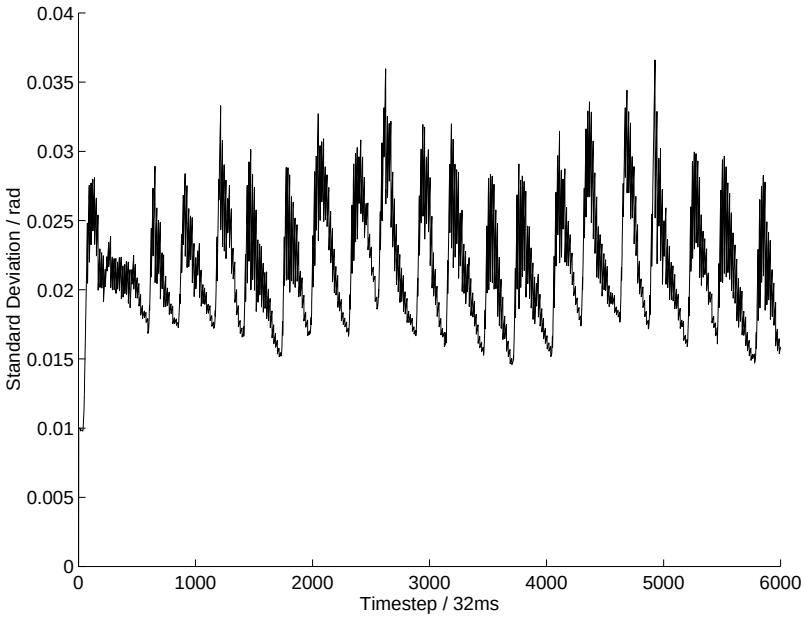

Figure 5.37: The standard deviation of the vehicle orientation estimate is very similar to that in the KNOK filter. However, the mean deviation is slightly larger in this case (0.025rad compared with 0.023rad). Oscillations with a period of about 2250 timesteps are again evident.

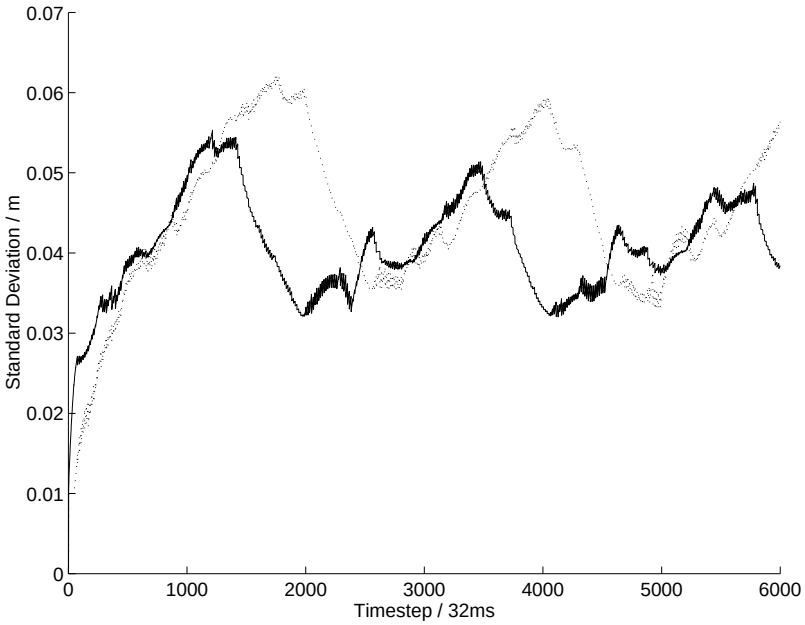

Figure 5.38: The standard deviations of the vehicle position estimate are quite different to those of the KNOK filter. The mean deviations are about twice as larger in the MAK filter than in the KNOK filter, and exhibit a 2250 timestep oscillation.

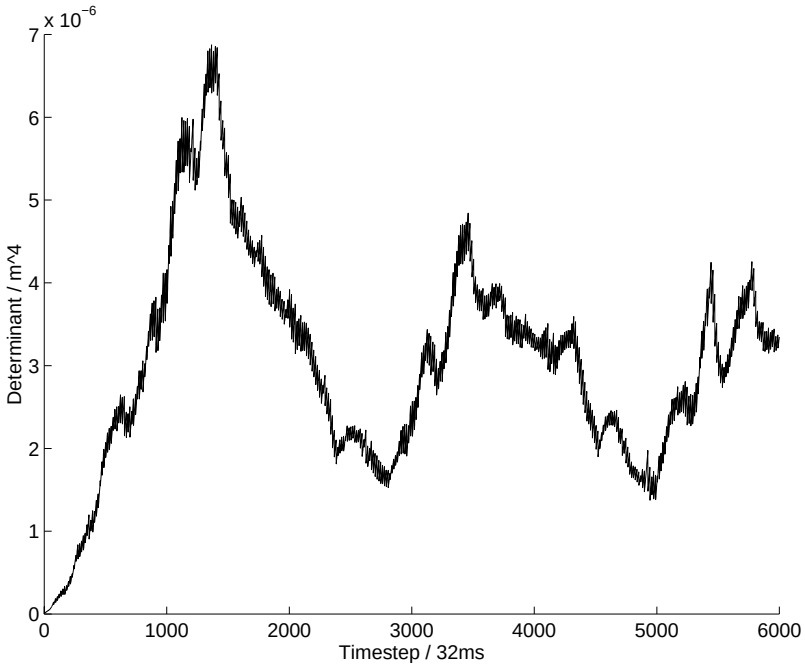

Figure 5.39: The determinant of the position covariance matrix exhibits the 2250 timestep oscillation, but the mean is more than an order of magnitude larger than in the KNOK filter ($3.0 \times 10^{-6} m^4$ compared to $1.6 \times 10^{-7} m^4$), and phase shifted.

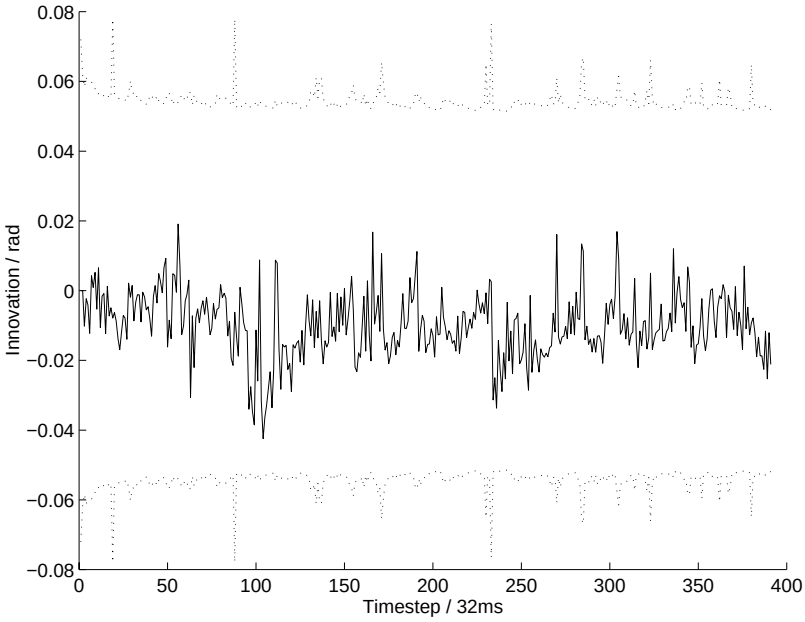

Figure 5.40: The innovation in the bearing is comparable to that of the KNOK filter, it is correlated and biased. The standard deviation of the innovation is over estimated.

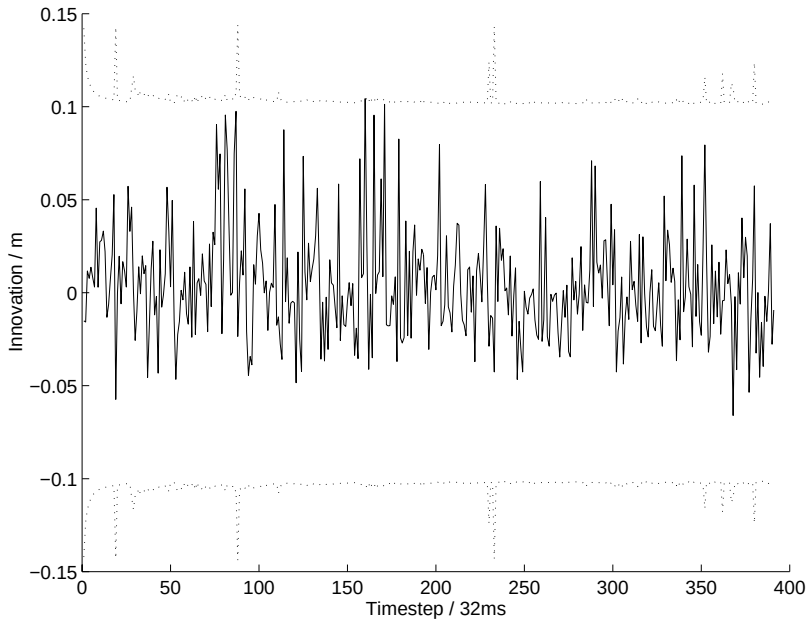

Figure 5.41: The innovation in the range is again comparable to that of the KNOK filter.

5.6.2 The Evolution of the Map

In the Figures 5.42 to 5.51 the evolution of the map produced by the MAK filter is documented. The true position of the features is marked with a 'x' and the 2σ (second standard deviation) covariance ellipse for the vehicle and the feature position estimates are shown, centered on the mean. The estimated location of the vehicle is the marked with a triangle, and the orientation of the triangle follows the mean of the estimate of the vehicle orientation, but the shape and size of the triangle is fixed. The mean of a position estimate, located at the centre of its covariance ellipse, is not marked by a separate symbol. This avoids cluttering the figures with symbols.

If the filter is operating consistently, 95% of the time the true position should fall within the covariance ellipse. Although some feature position estimates have errors well beyond the second standard deviation, as shown in Figures 5.42 to 5.51, most feature position estimates show errors consistent with their standard deviations. The serious disagreements can be traced to failures of the feature extraction algorithm on features such as shown in Figure 5.57, and to the suboptimal operation of the MAK filter, suggested by the innovations shown in Figures5.40 and 5.41.

Figures 5.52 to 5.55, show the evolution of the estimate of the x and y coordinate of a particular feature, located at (at 2.742,-4.575m). In each figure, the error of the estimate is plotted with the standard deviation of the estimate, demonstrating the consistency of the MAK filter.

The standard deviations of the features estimates initialised when the vehicle is close to its

starting point are smaller than the standard deviations of the feature estimates initialised further from the starting point. This is can be seen, for example, when comparing the initialised covariance ellipses in Figures 5.43 and 5.48. The phenomenon is due to the accumulation of process noise by the vehicle as it travels further from the starting point. All standard deviations decrease with time as more updates are made, but those of estimates of features near the starting point are always lower than those of remote features.

The evolution of standard deviations of all feature estimates is shown in Figure 5.56. All standard deviations are seen to be monotonically non-increasing, as was derived in Section 3.5. Figure 5.56 also suggests there is a lower limit to the standard deviation of the feature estimates. This is in agreement with the derivations in Section 3.7.

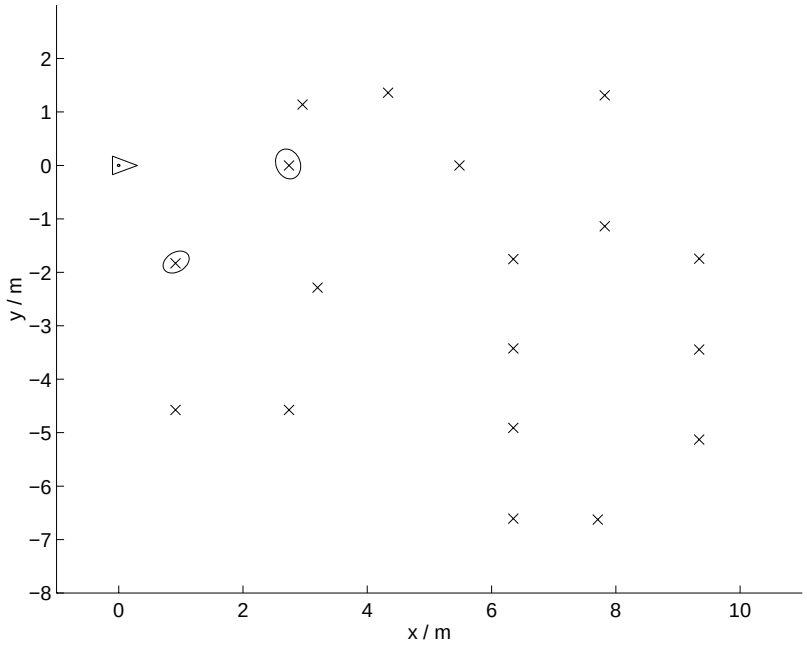

Figure 5.42: Timestep 1, after the first laser scan, feature estimates have been initialised for the features located at (0.914m,-1.830m) and (2.742m,0.000m).

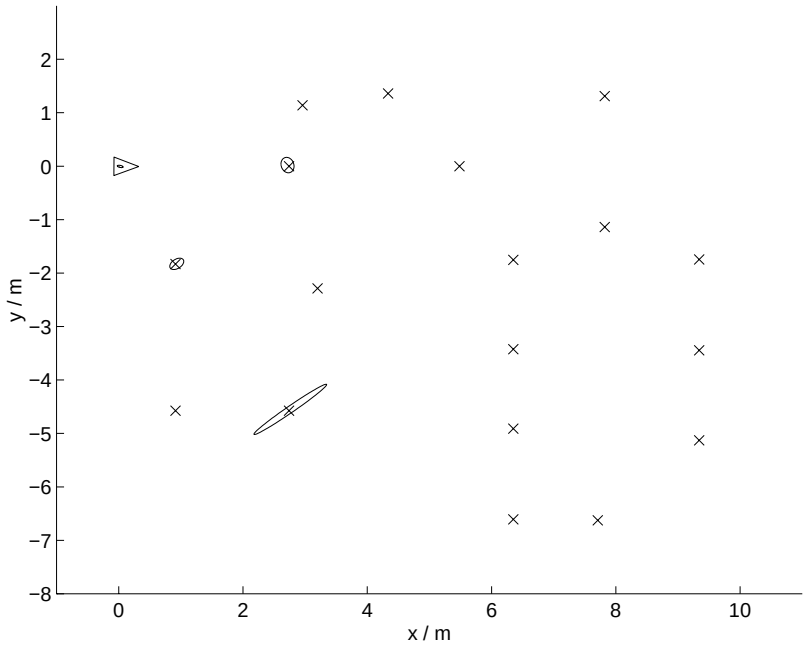

Figure 5.43: Timestep 46, a feature estimate has been initialised for the feature located at (2.742m,-4.575m). The uncertainty in the bearing of the observation has a greater effect with increasing range. This leads to a covariance ellipse with the major axis perpendicular to the line of sight. The first two estimates have been updated and their covariance ellipses have decreased in area.

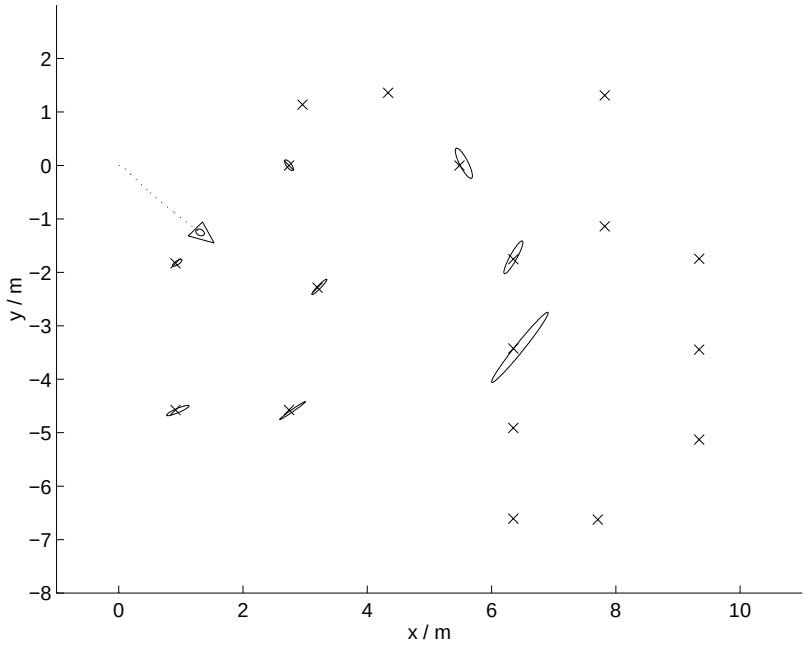

Figure 5.44: Timestep 410, the covariance ellipse of the vehicle position estimate has increased significantly. This and the large range account for the large covariance ellipse initialised at 6.350m,-3.424m).

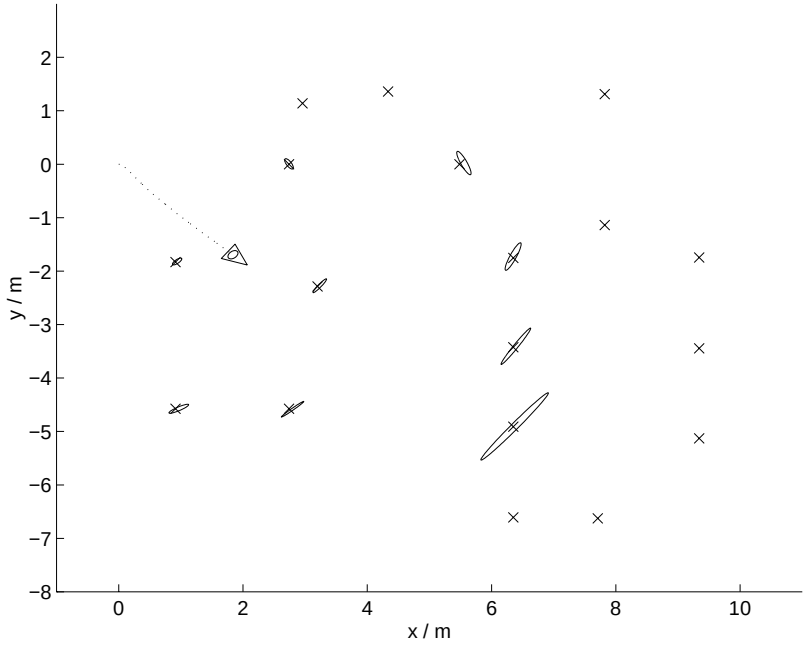

Figure 5.45: Timestep 577, only one covariance ellipse does not cover the true position of the estimated feature, located at (5.484m,0.000m), but the true position is close to the edge of the ellipse. Therefore, the MAK filter seems to be operating consistently. A new initialisation occurred at (6.350m,-4.911m).

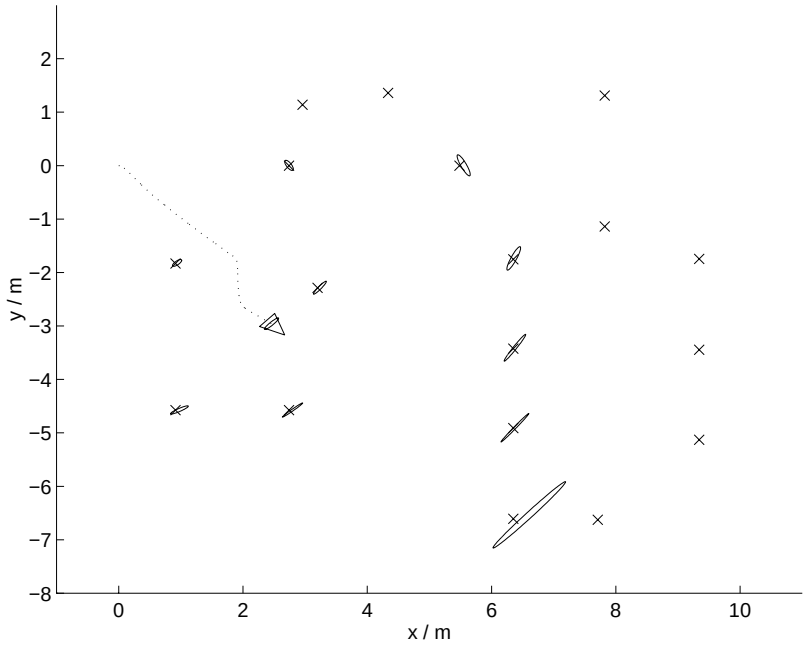

Figure 5.46: Timestep 987, the first serious error of a position estimate has occurred. The true position of the feature is located at (6.31m,-6.63m). However, the feature is particularly difficult to estimate since it is a double strut of the railing, as shown in Figure 5.57, and cannot be treated as a point feature.

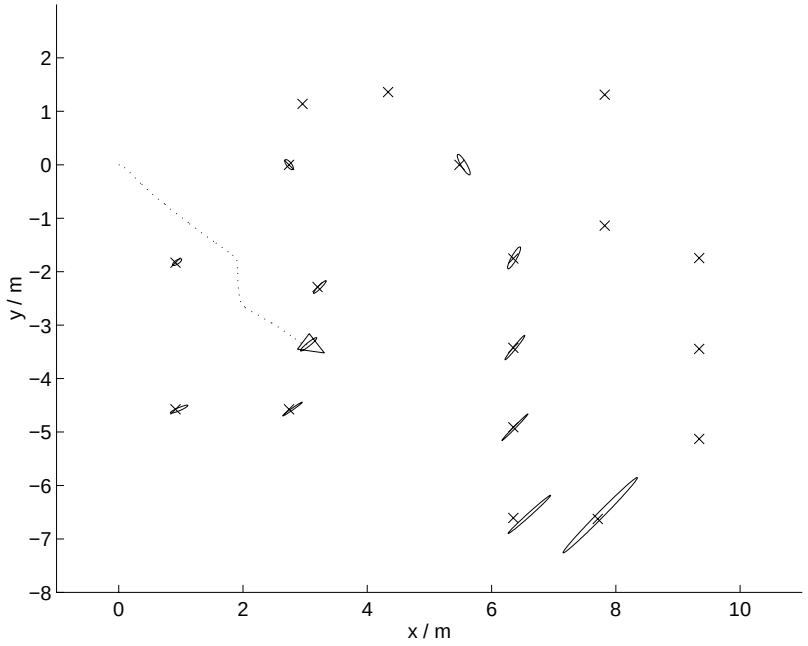

Figure 5.47: Timestep 1199, the true position of the next initialised feature, located at (7.710m,-6.625m), again falls within the estimated covariance ellipse, suggesting the consistent operation of the MAK filter.

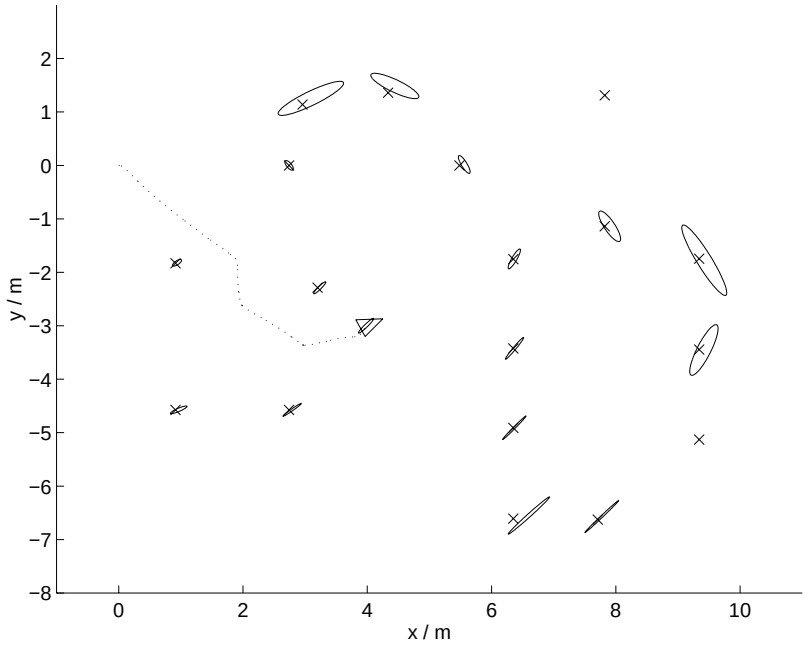

Figure 5.48: Timestep 1488, a large number of features have been initialised during the last 100 timesteps due to the rotation of the vehicle opening up a new area to the forward looking sensor. The most recent is at (9.342m,-1.745m).

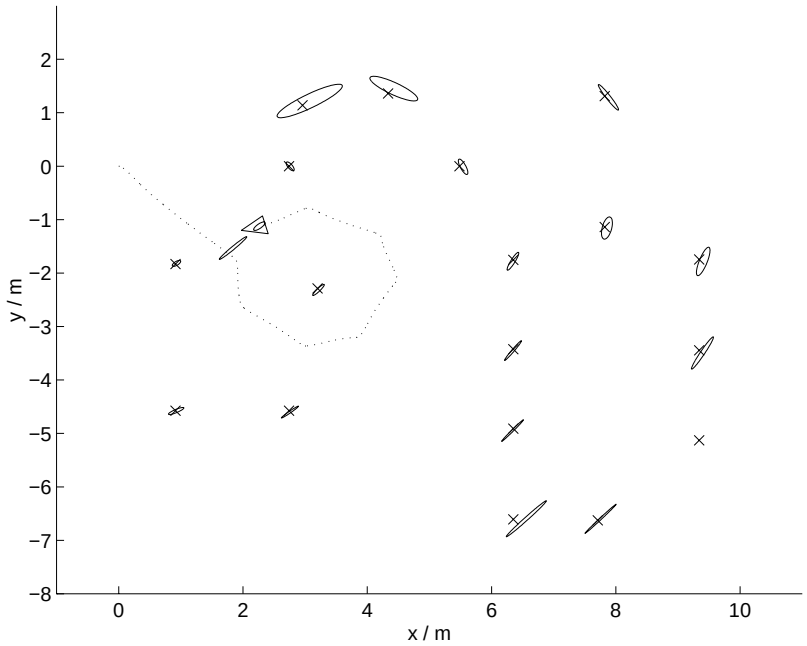

Figure 5.49: Timestep 2549, a spurious observation leads to a 'ghost' feature at (1.840m,-1.531m). The observation may have been caused by the umbilical cord used during the logging process. The vehicle has almost completed one circuit.

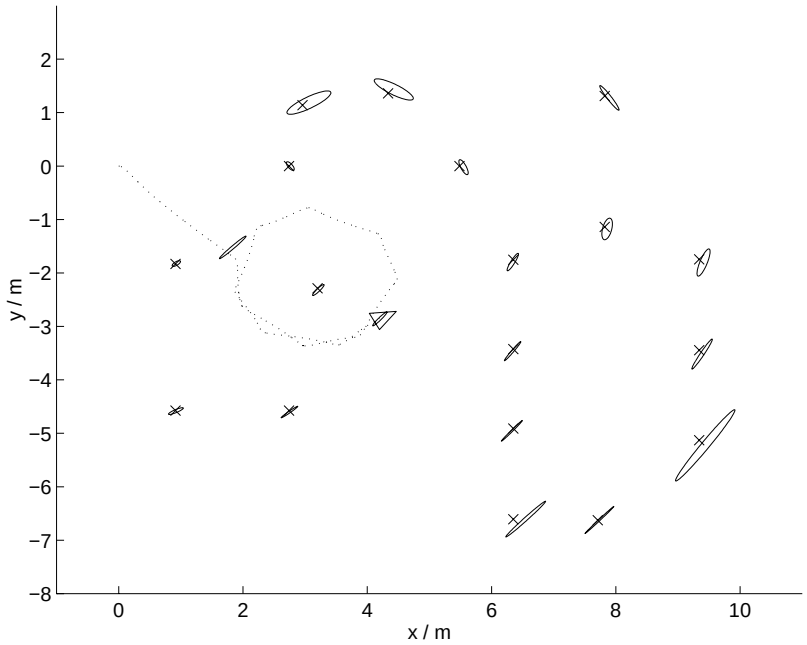

Figure 5.50: Timestep 3687, a feature at (9.342m,-5.128m) has been initialised which was missed during the first circuit.

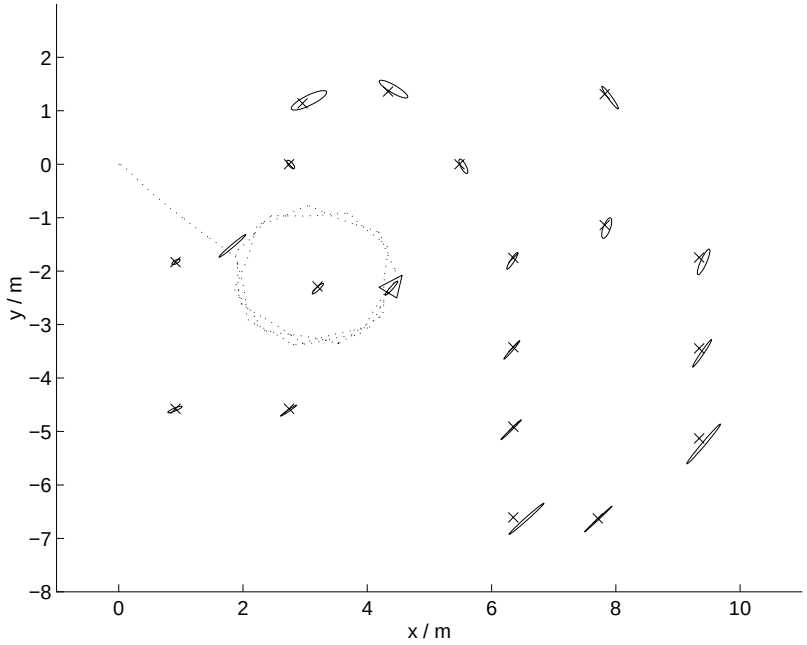

Figure 5.51: Timestep 6000, the end of the experiment has been reached. The estimates to the far right seem to be biased, however, this is not surprising as it was pointed out that the filter is not operating optimally. All feature estimates have been updated since their initialisation.

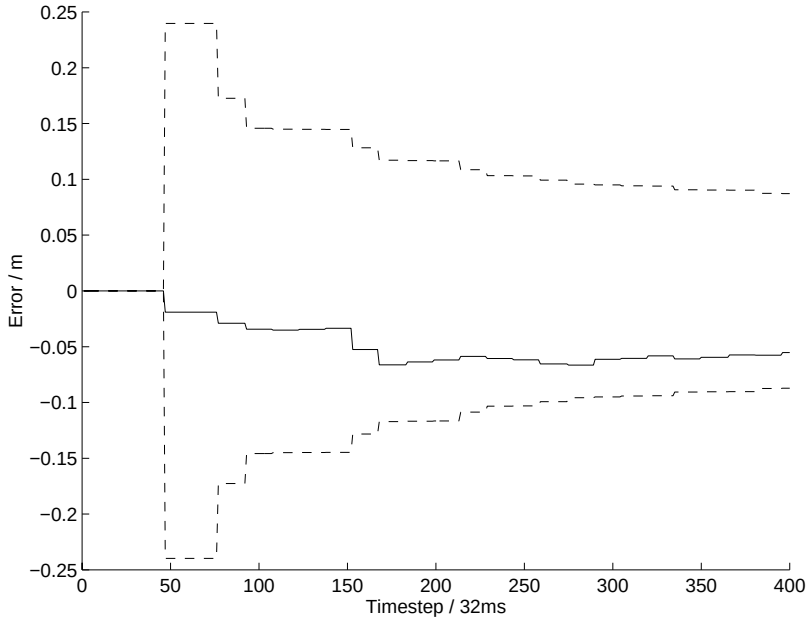

Figure 5.52: Initial stages of the error and standard deviation of the estimate of the x-coordinate of a feature (at 2.742,-4.575m).

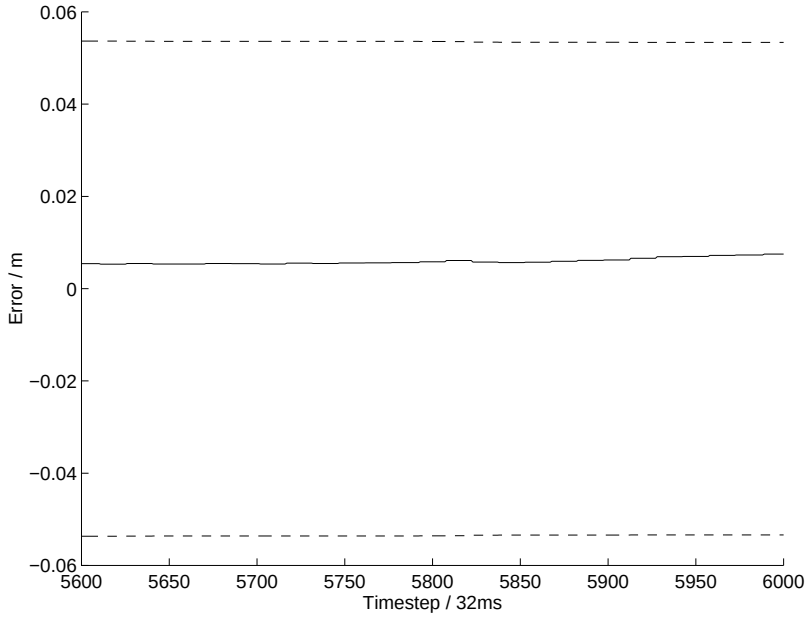

Figure 5.53: Final stages of the same error and standard deviation. The important feature of the error is that it is consistent with its variance. The bias is unfortunate, but it does not signify inconsistency.

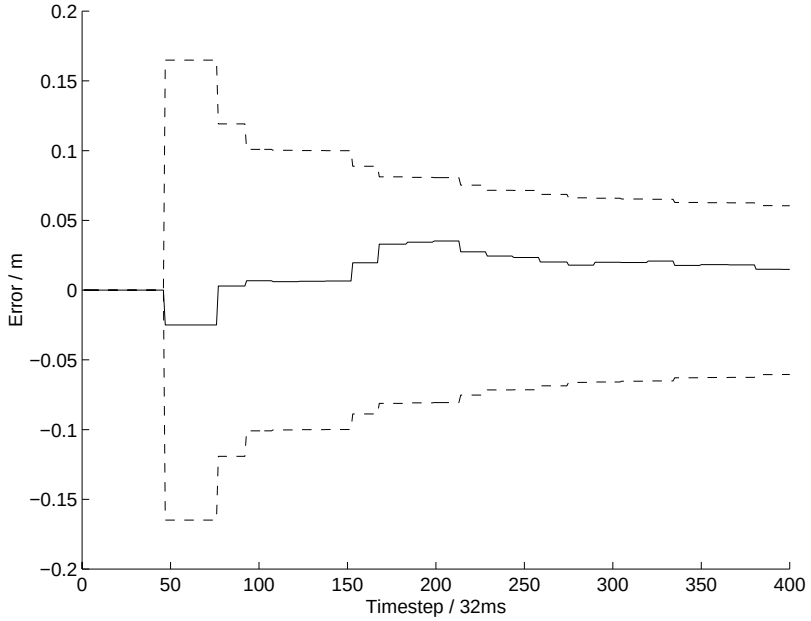

Figure 5.54: Initial stages of the error and standard deviation of the estimate of the y-coordinate of a feature.

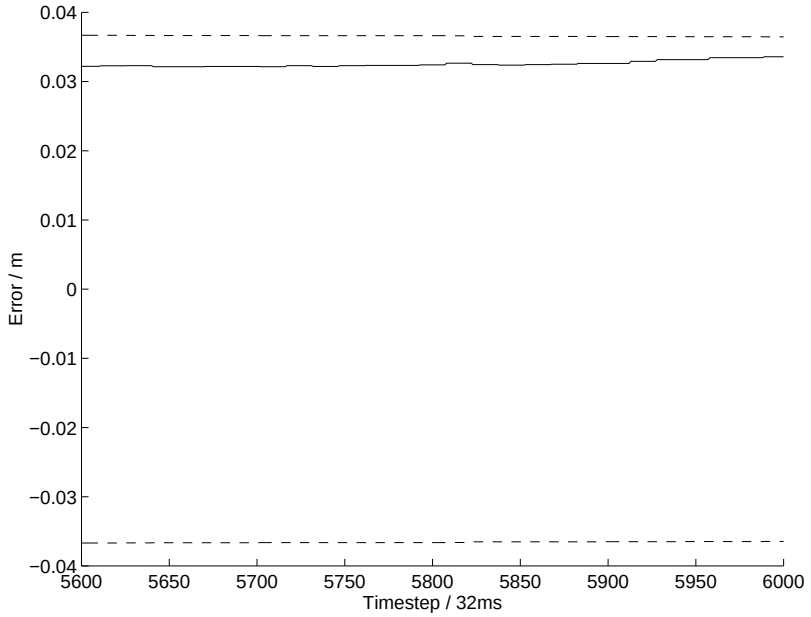

Figure 5.55: Final stages of the same error and standard deviation.

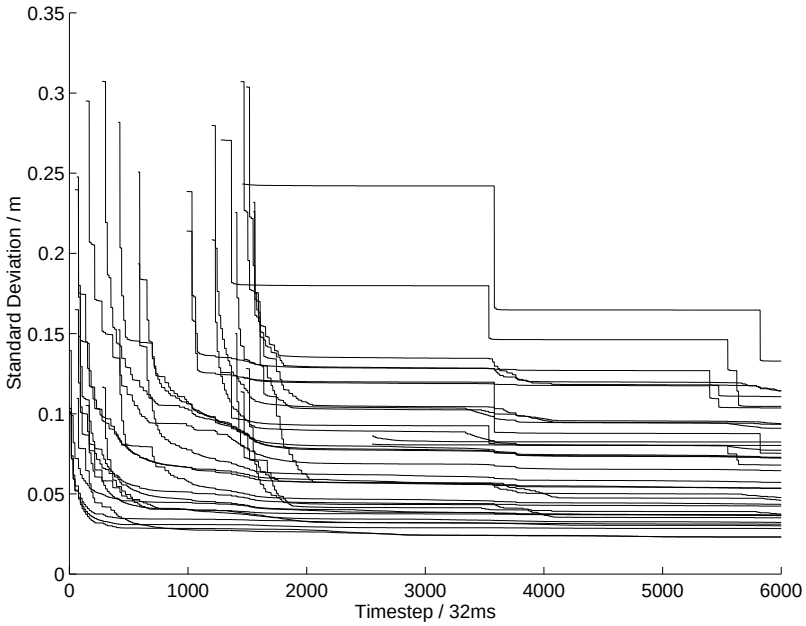

Figure 5.56: The standard deviations of all the feature estimates are monotonically non-increasing and exhibit a lower limit.

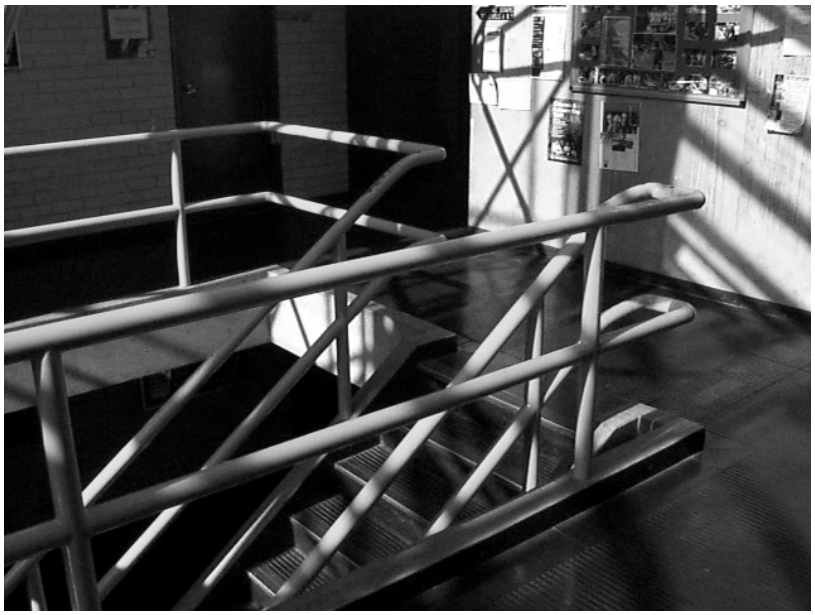

Figure 5.57: A double strut of the railing. The extraction algorithm fails on this feature, producing a range and a bearing that do not correspond to a single strut.

5.6.3 Conclusions

From the results in this section it can be concluded that the MAK filter provides a solution to the SLAM problem. The feature estimates are, considering the imperfections of the experiment, consistent and convergent, and the vehicle is also successfully localised.

The performance of the MAK filter, measured by the covariance of the estimates, depends significantly on the location in the environment. The feature estimates far from the starting point of the vehicle have lower covariances than the feature estimates near the starting point. This is associated to the accumulation of process noise by the vehicle as it traveled to initialise estimates of remote features.

Linked to the covariance of the feature estimates is the covariance of the vehicle estimate. In areas where the feature estimate covariance is low, the vehicle estimate variance will also be low. This follows from the fact that the usefulness of the vehicle state update depends on the variance of the observed feature.

The variance of the feature estimates was observed as monotonically non-increasing, as derived in Section 3.5. In particular, it was observed that the variance of all feature estimates decreased, and as a consequence the localisation of the vehicle improved.

The limit derived in Section 3.7 was also confirmed as a lower limit on the variance of the feature estimates and an initial increase in the variance of the vehicle estimate which was never recovered.

The consistency in the feature estimates suggests that the vehicle state estimate errors are also consistent with their standard deviations, however no direct verification of the errors in the vehicle estimates was available.

5.7 The Bounded Region Filter

This section examines experimental results from the BOR filter. Section 5.7.1 describes the operation of the filter. The evolution of the map constructed by the BOR filter is examined in Section 5.7.2 and conclusions are presented in Section 5.7.3.

5.7.1 The Operation of the Bounded Region Filter

The estimates of the BOR filter have no standard deviations associated with them. The quality of a BOR filter estimate is therefore measured by the size of the bounds on the estimate. The smaller the bound on an estimate, the better the quality of the estimate; a small bound on a BOR filter estimate corresponds to a small standard deviation in a MAK filter estimate. The total uncertainty of a bounded estimate can be measured by the volume enclosed by the bounds, or the area in the two dimensional case. The volume or area of a bounded estimate corresponds the the determinant of the covariance matrix of a probabilistic estimate.

The BOR filter was run on the same data set as the MAK filter. Figures 5.59 and 5.58 show the evolution of the bounds of the vehicle state estimates. The bounds in the orientation estimate and the x and y position estimates all exhibit a low frequency oscillation with a period of about 2250 timesteps. Such an oscillation was already encountered in the KNOK and the MAK filters and is linked to the quality and constellation of the observable features. The constellation of the features is the same in the MAK as in the BOR filter, and the quality of the feature estimates is distributed over the environment in the BOR filter in a similar way to the distribution in the MAK filter, as described in Section 5.7.2. This suggests that the 2250 timestep oscillation in the BOR filter also arises due to the quality and constellation of the observable features.

The bounds on the x and y vehicle position estimates in the BOR filter evolve similarly to the standard deviations of the x and y position estimate in the MAK filter, with peaks at approximately timesteps 1250, 3500 and 5750. The peaks correspond to the vehicle being at its furthest form the starting point, where the quality of the feature estimates is the worst, as argued in Section 5.7.2. The usefulness of the update of the vehicle position estimate depends on the quality of the observed features. Consequently the quality of the vehicle position estimate follows the quality of the feature estimates in the vicinity of the vehicle, as shown in Figures 5.59 and 5.58, and as is also the case in the MAK filter.

The evolution of the vehicle orientation estimate and the evolution of the vehicle position estimate are very similar, as can be seen in comparing Figures 5.58 and 5.59. The quality of the vehicle orientation estimate also follows the quality of the estimates of the features in the vicinity of the vehicle. This is in strong contrast to the evolution of the standard deviation of the vehicle orientation

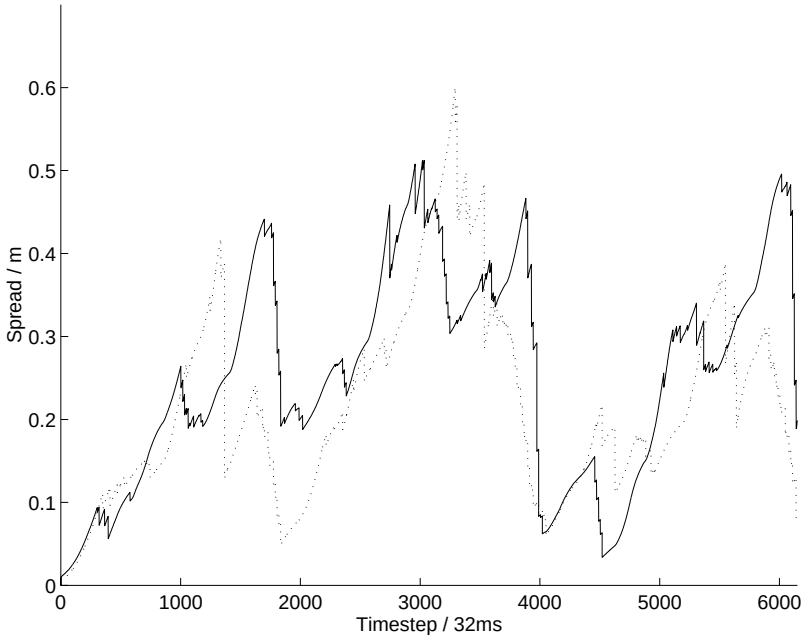

Figure 5.58: Evolution of the spread of the position estimate is dominated by an oscillation of approximately 2250 timesteps.

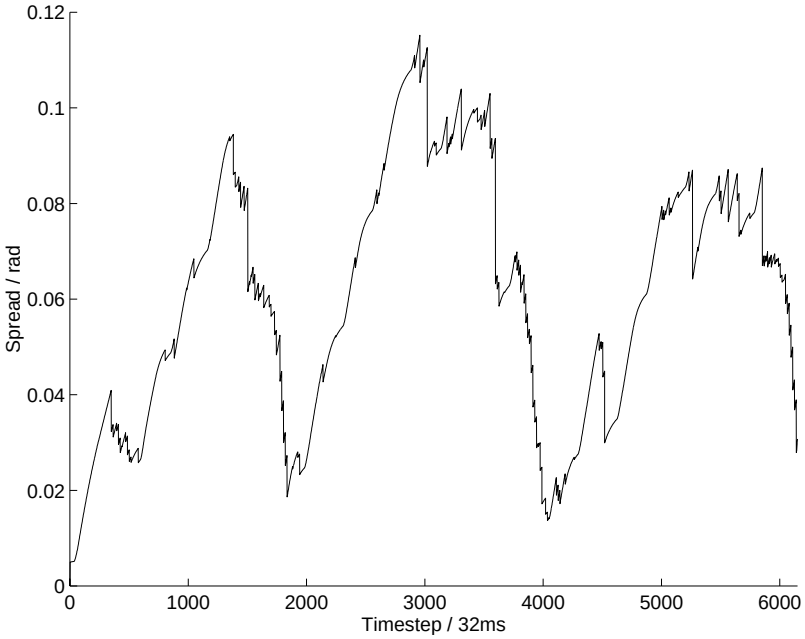

Figure 5.59: Evolution of the spread of the orientation estimate is dominated by an oscillation of approximately 2250 timesteps.

estimate in the KNOK and MAK filters. These were affected mainly by the manoeuvres of the vehicle and the particular feature constellation observed by the vehicle, but were not affected by the regional quality of the features estimates.

The importance of the quality of feature estimates in the case of the BOR filter may be due to the very poor quality of the position estimate of the feature at (6.350m,-6.607m) and its neighbours, as can be seen in Figures 5.68 to 5.74. Observations made in that region may fail to be associated uniquely with a feature and consequently cannot be used in an update. Even if the observation is associated uniquely, the update may still prove to be insignificant due to the poor quality of the feature estimates. The difference in the quality of the estimate of features near the origin and remote to the origin leads to the evolution of the orientation estimate shown in Figure 5.59.

The feature constellations along the path of the vehicle, which have a visible effect on the performance of the orientation update in the KNOK and the MAK filter, have no visible effect in the BOR filter. This is probably due to the fact that the extreme differences in the quality of the feature estimates dominate the orientation update in the BOR filter.

Since the orientation estimate is not significantly affected by manoeuvres, the prediction stage of the BOR filter is inferior to the prediction stage of the PREK, KNOK and MAK filters. The effect of the process noise increases in the PREK, the KNOK and the MAK filters during a manoeuvre, this is a result of the Jacobians used to transform the process noise into vehicle state space. In the BOR filter, the process noise is not transformed but directly added during a prediction, and therefore leads to an inferior prediction stage.

The inferior prediction stage does not entirely account for the poor performance of the BOR filter. A further factor affecting the performance of the BOR filter is the distribution of the process noise. The BOR filter assumes that the process noise is bounded, but makes no assumptions about the distribution of the noise within those bounds. However, the distribution does play an important role in the operation of the BOR filter.

Consider a very simple BOR filter in which a feature is observed by a stationary vehicle whose position is known perfectly. In order to update the position estimate of the observed feature, the BOR filter only intersects the current estimate with each new observation. Given that the observation noise is bounded and the bounds have been correctly set in the BOR filter, the BOR filter will progressively improve its estimate as new observations are made. However, if the noise distribution of the observation has a large mass near zero, most of the observations will be very close to the true feature position, but the BOR filter must still apply the initially specified maximum bounds. This may potentially lead to very inefficient updates and a slow convergence as shown in Figure 5.60, and identified as one of the disadvantages of BOR filters in Section 4.2.1.

The effect of the noise distribution can also be seen in a simple simulation. The problem is reduced

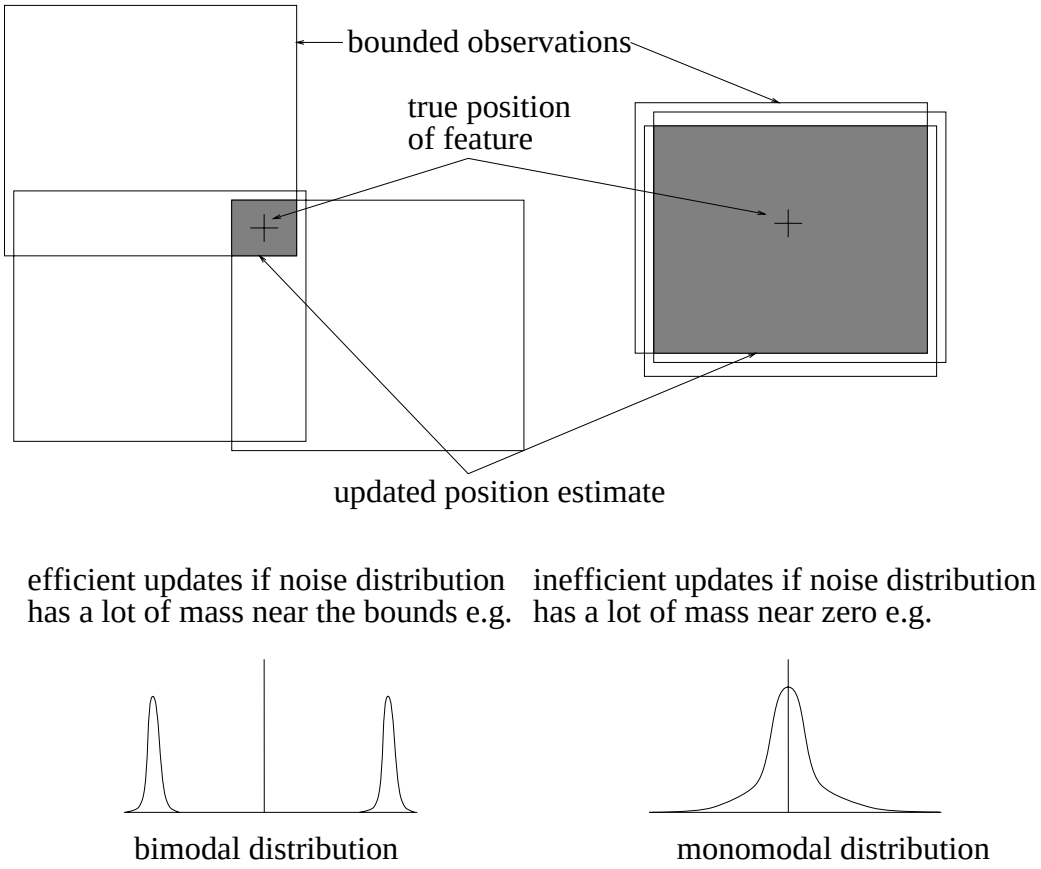

Figure 5.60: The effect of the noise distribution in a BOR filter.

to one dimension and two different noise distributions are considered. The first noise distribution considered is a truncated monomodal distribution, and a histogram of the actual noise injected is shown in Figure 5.61. The evolution of the BOR filter estimate is shown in Figure 5.62. The second noise distribution is a truncated bimodal distribution, with a concentration of mass near the lower bound on the noise distribution and another concentration near the upper bound. A histogram of the actual noise injected in shown in Figure 5.63, and the evolution of the BOR filter estimate is shown in Figure 5.64.

It is evident that the BOR filter estimate converges much more rapidly in the truncated bimodal case than in the truncated monomodal case. A noise distribution with large probability mass near the extreme bounds will lead to a more efficient BOR filter than a noise distribution with large probability mass near zero.

It is more than likely that the actual noise encountered during the experiment is bounded, but it is not as likely that the noise distribution is favourable to the BOR filter. The unfavourable noise distribution not only leads to inefficient updates, but may potentially cause the filter to diverge.

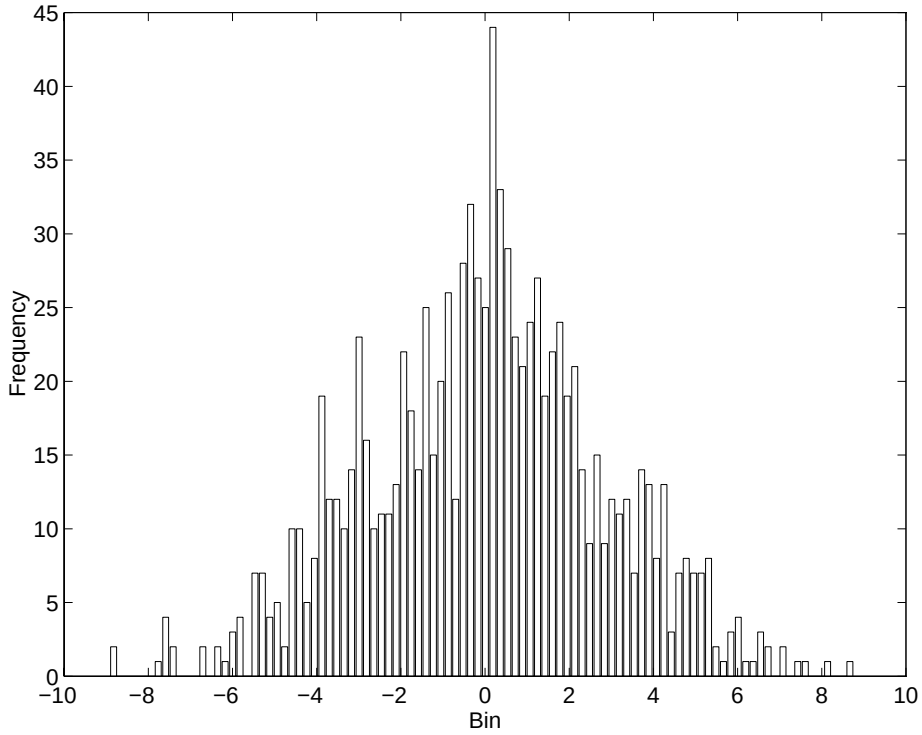

Figure 5.61: Histogram of the actual noise, consisting of 1000 samples, injected in the truncated normal distribution case.

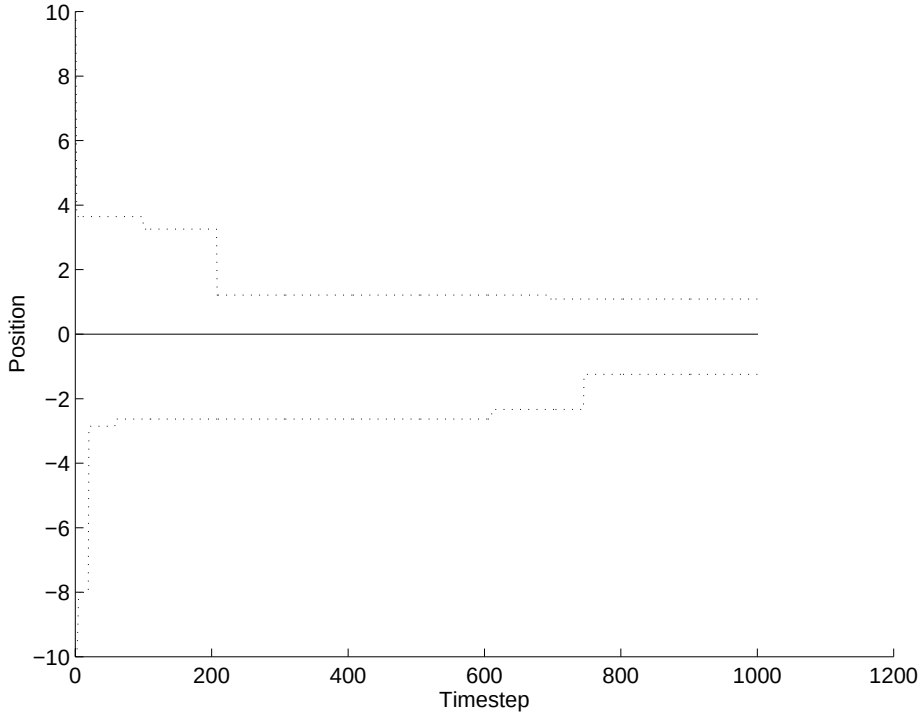

Figure 5.62: Evolution of the estimate in the truncated normal distribution case. The true position is at zero and the upper and lower bounds of the estimate are shown by the dotted lines.

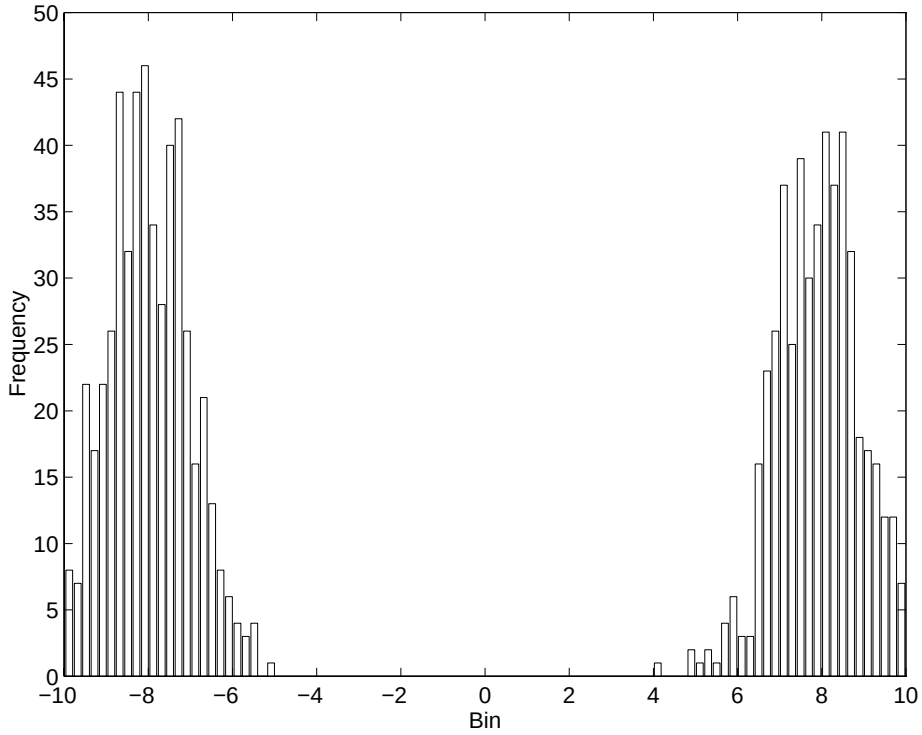

Figure 5.63: Histogram of the actual noise, consisting of 1000 samples, injected in the truncated bimodal distribution case.

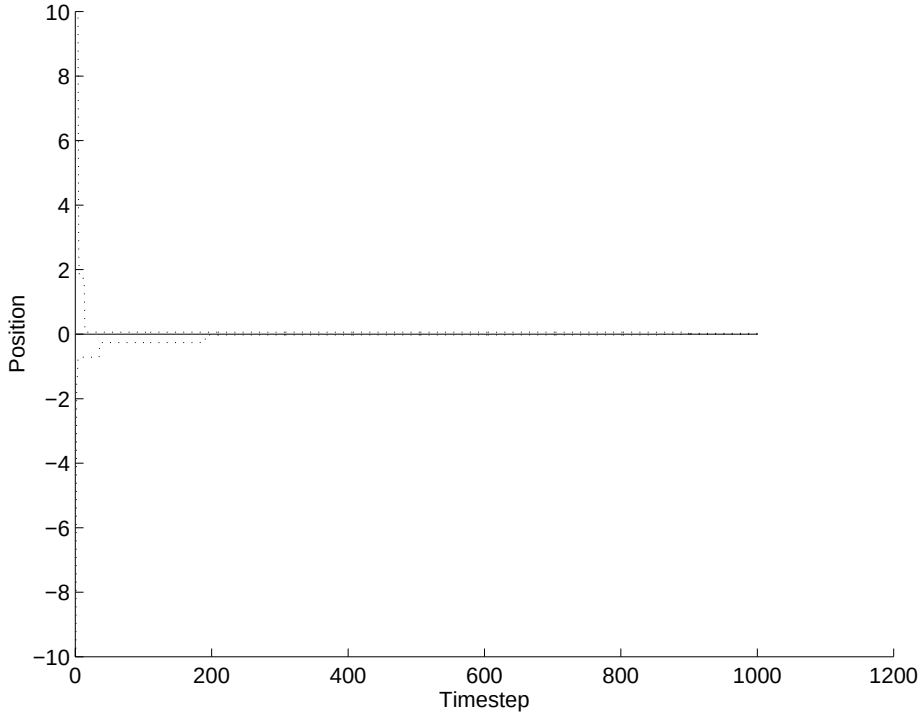

Figure 5.64: Evolution of the estimate in the truncated bimodal distribution case. The true position is at zero and the upper and lower bounds of the estimate are shown by the dotted lines.

state	initial minimum bound	initial maximum bound
x_v	-0.005 m	0.005 m
y_v	-0.005 m	0.005 m
ϕ_v	-0.005 radians	0.005 radians

Table 5.3: Bounds on the initial vehicle state estimate.

noise source	bound
velocity V	$\pm$ 0.002 m/s
steer angle γ	$\pm$ 0.007 radians
range r	$\pm$ 0.0175 m
bearing θ	$\pm$ 0.0155 radians

Table 5.4: Bounds on the process and observation noise.

This situation arises if the predictions increase the bounds on the estimates so rapidly that the inefficient updates cannot control the increase.

The only way to stop the filter diverging is to reduce the assumed bounds on the process noise and/or on the observation noise. Reducing the assumed process noise bounds leads to a slower increase in the estimate bound during the prediction and reducing the observation noise bounds leads to a more rapid decrease in the estimation bounds during the update. However, reducing the assumed noise bounds beyond their true bounds leads to inconsistency.

Unfortunately, this was the situation in the actual experiment. It was possible to construct a map based on the collected data using the BOR filter, but the filter was not consistent. The initial estimate of the vehicle state, and the process and observation noise bounds finally chosen are given in Tables 5.3 and 5.4. Comparing the chosen bounds and the variances chosen for the same parameters, given in Tables 5.1 and 5.2, shows the inconsistency.

It is very likely that the bounds on the process noise and the observation were violated during the experiment. This is also evident in the map that was produced by the BOR filter, which is examined in Section 5.7.2.

5.7.2 The Evolution of the Map

The Figures 5.65 to 5.74 document the evolution of the map produced by the Bounded Region filter. The true positions of the features are again marked with an 'x' and the rectangular position bounds of the vehicle and feature estimates are shown. A triangle marks the estimated vehicle position. The orientation of the triangle follows the center of the orientation bound of the vehicle, but the shape and size of the triangle is fixed.

Similar to the MAK filter, the quality of the feature estimates is best at the origin, near the starting point of the vehicle, and deteriorates with distance from the starting point. This can be seen in Figures 5.65 to 5.74 by comparing the bounds on the estimate of features far from the origin and the bounds on the estimate of features close to the origin. The deterioration of the quality of the feature estimates has been linked in the MAK filter to the accumulation of process noise by the vehicle as it travels to remote features. This was described in Section 5.6. Similar arguments apply in the BOR filter. As the vehicle is traveling, the accumulation of process noise leads to an increase in the bounds on the vehicle estimate. Since the initialisation of the feature estimates depends on the vehicle estimate, it follows that a vehicle estimate with a large bound will lead to an initialised feature estimate with a large bound. Similarly, a small bound on the vehicle estimate leads to a small bound on the initialised feature estimate. This explains the distribution in the quality of the feature estimates within the map.

When the vehicle returns to an area of the environment, it is able to improve on the quality of the feature estimates in that area. This can be seen by comparing Figures 5.72 and 5.74, where the bounds on the feature estimates in the vicinity of the vehicle can be clearly seen to have decreased, equivalent to an improvement in the quality of the feature estimates. This is similar to the evolution observed in the map constructed by the MAK filter. However, the significant difference is that the BOR filter only updates the observed features, whereas the MAK filter can potentially update unobserved features via the correlations, as described in Section 3.3.

An artifact of the rectangular bounded region used by the BOR filter is that an observation that does not align well with the coordinate axis leads to a poor quality feature initialisation, equivalent to a large area of the initialised position bound. This can be seen in Figure 5.70, and is an unfortunate property of the particular bounds used by the BOR filter.

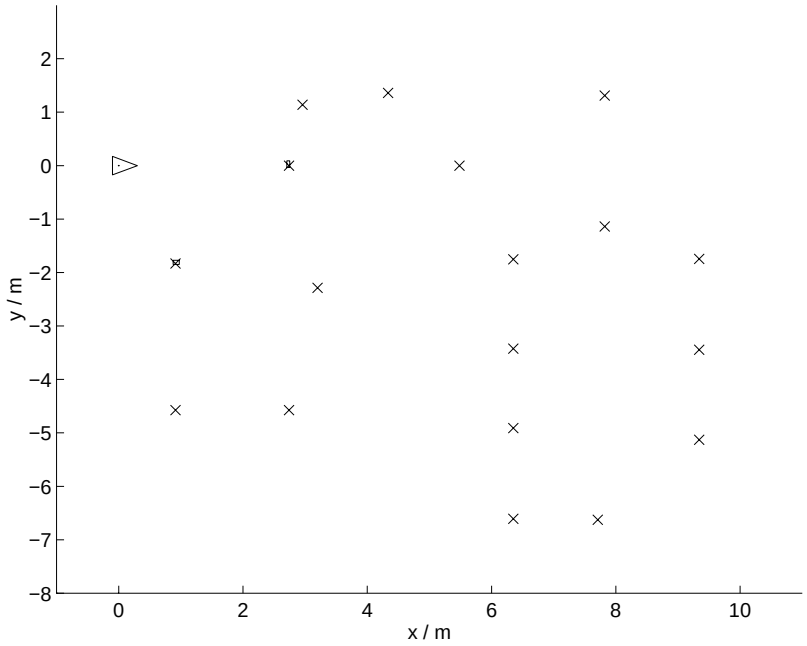

Figure 5.65: Timestep 1, estimates for the features at (0.914m,-1.830m) and (2.742m,0.000m) are initialised on the first laser scan.

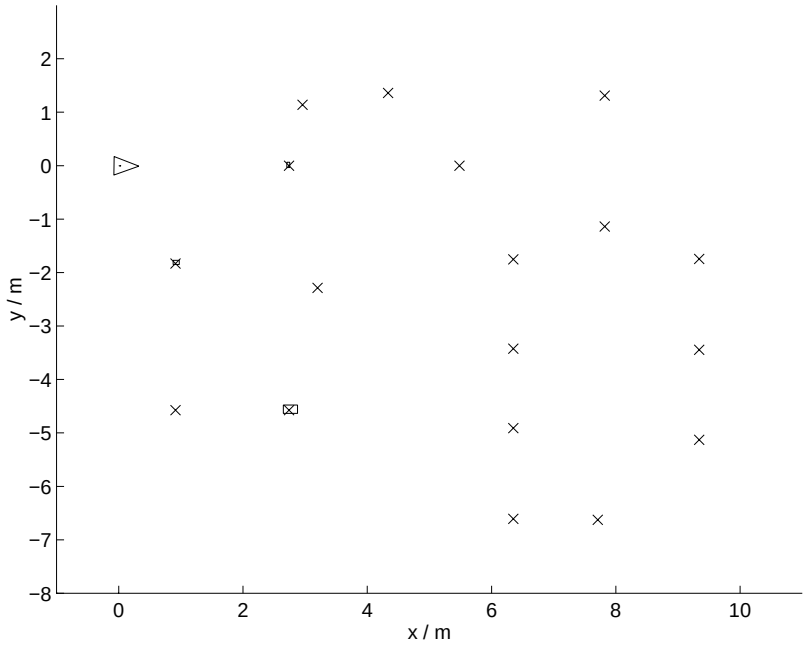

Figure 5.66: Timestep 46, a feature estimate is initialised at (2.742m,-4.575m) with a larger rectangle. Since the third feature is more distant, the bearing uncertainty becomes dominant and requires a larger rectangle as the initial estimate.

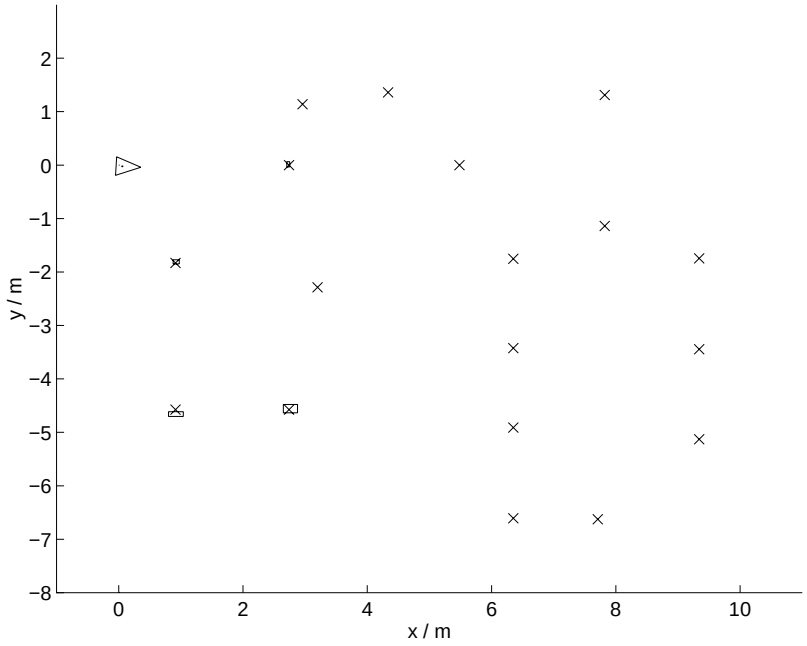

Figure 5.67: Timestep 61, the fourth feature initialised, at (0.914m,-4.575m), is definitely inconsistent, suggesting that the bounds were set too tight.

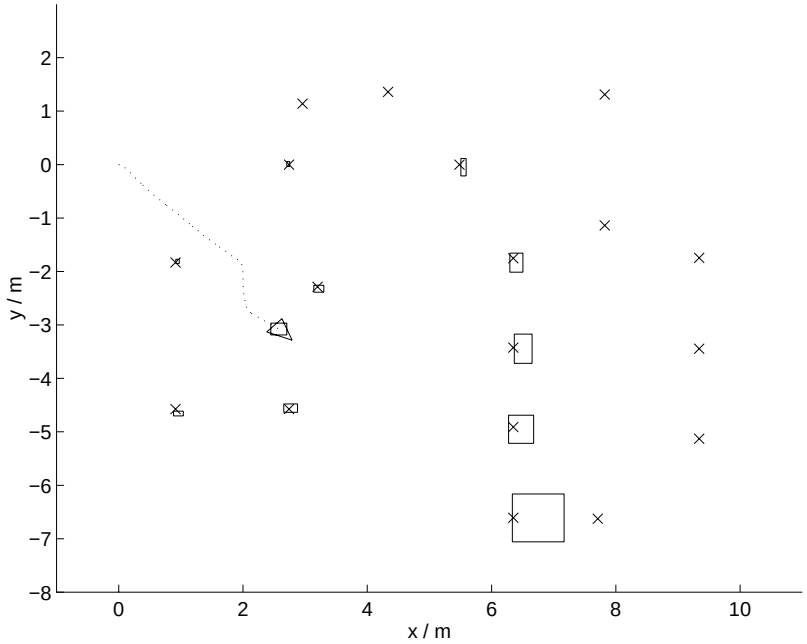

Figure 5.68: Timestep 987, the vehicle position bound has increased significantly. This leads directly to the large feature estimate initialised at (6.350m,-6.607m).

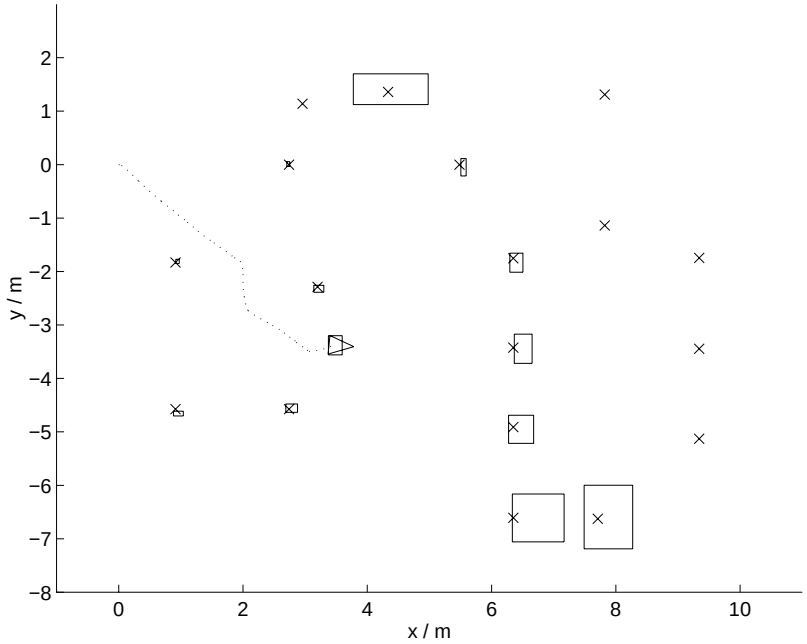

Figure 5.69: Timestep 1275, the line of sight to the feature initialised at (4.330m,1.360m) is well aligned with y axis. Therefore uncertainty in the bearing, which is very significant at a long range, only affects the bound in the x axis.

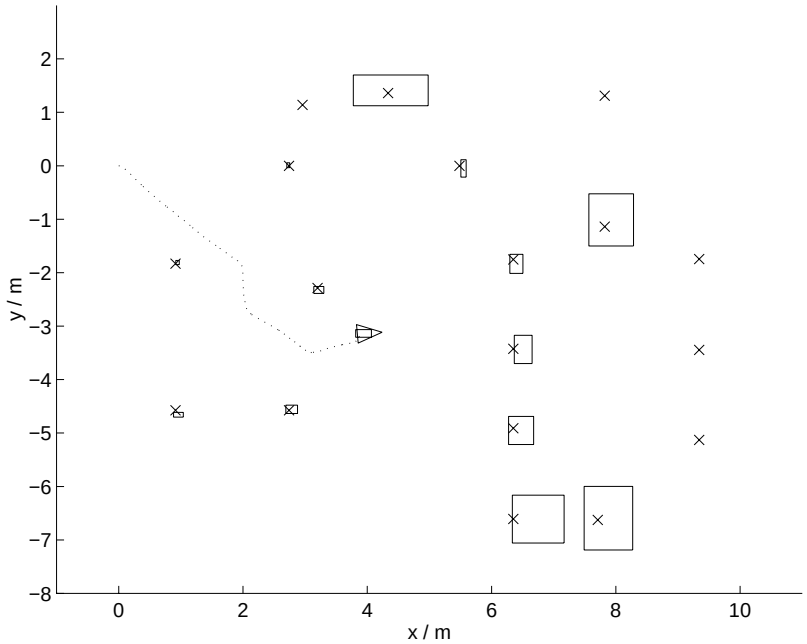

Figure 5.70: Timestep 1397, the line of sight to the feature initialised at (7.820m,-1.139m) is close to a line of slope 1. Therefore the uncertainty in the bearing affects the bound in the x and the y axis.

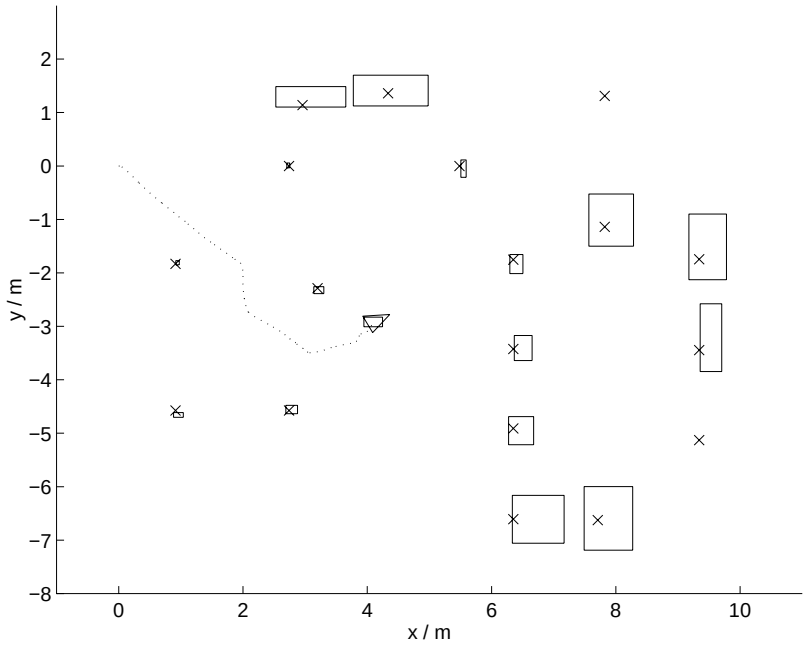

Figure 5.71: Timestep 1488, some estimates have become inconsistent with the true feature positions, while others are initialised with a very large bound, as in (9.342m,-1.745m). Therefore, it is unlikely that the BOR could have been tuned further.

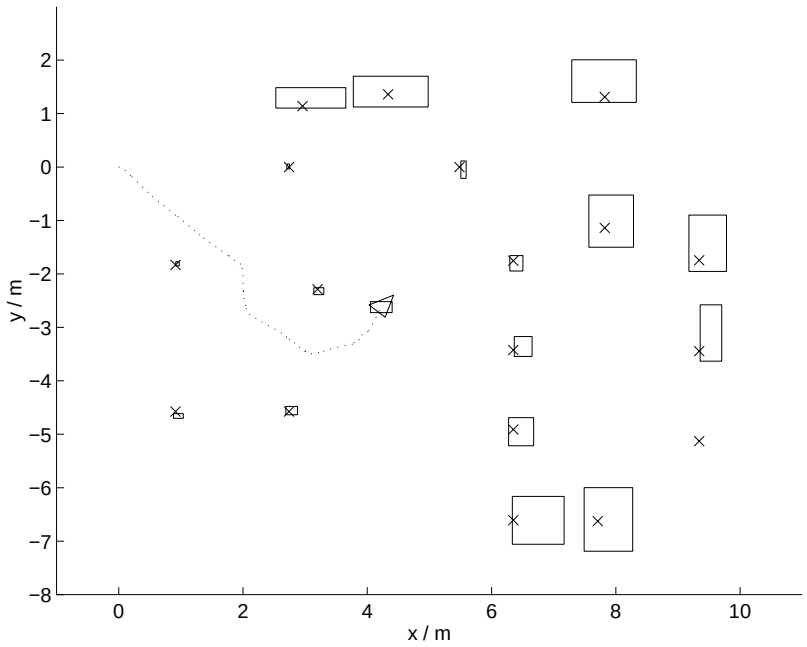

Figure 5.72: Timestep 1548, the estimates show that the bounds of a feature estimate increases with the distance traveled by the vehicle before the feature is initialised. A new feature has been initialised at (7.822m,1.310m).

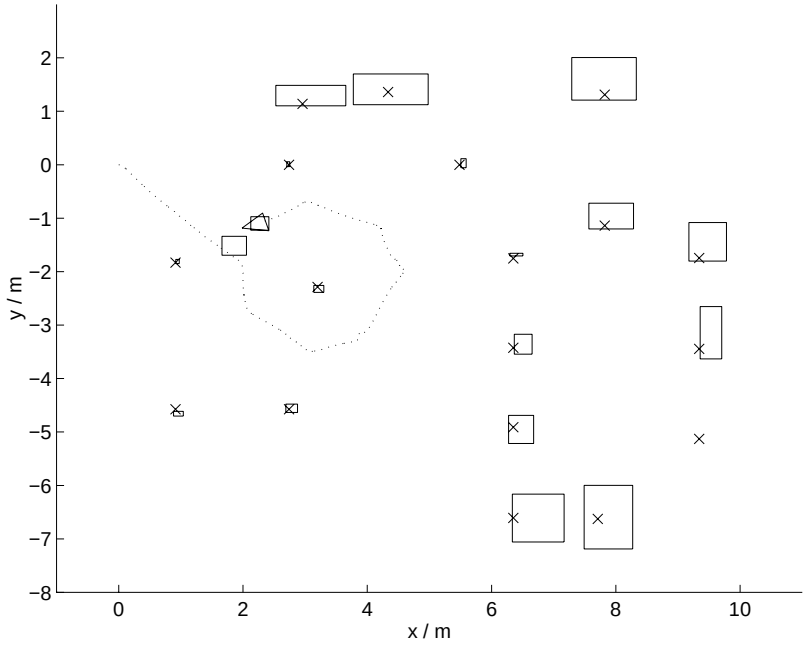

Figure 5.73: Timestep 2549, the BOR filter also initialises a 'ghost' estimate as the MAK filter did. This is expected since exactly the same data was used for both filters.

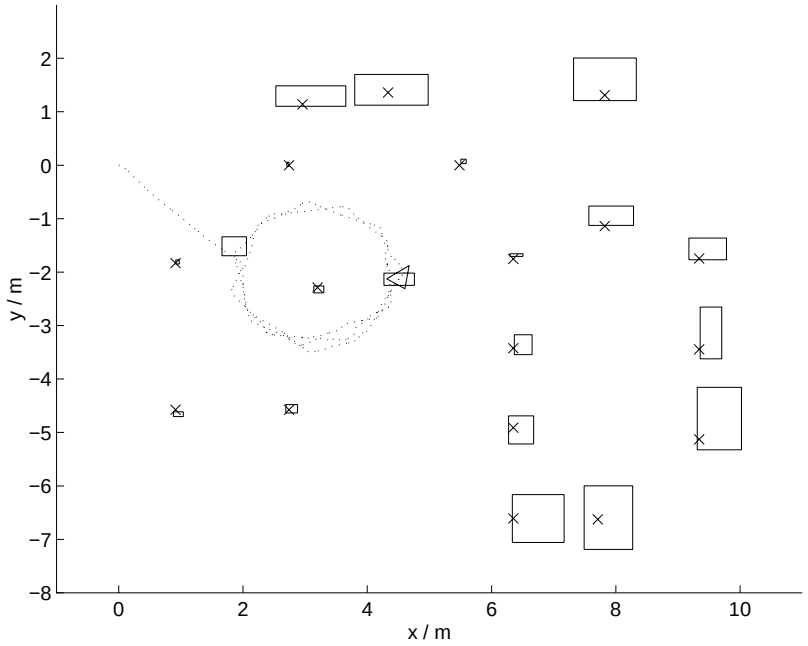

Figure 5.74: Timestep 6000, the end of the experiment has been reached. A large number of the feature estimates are not consistent with the true feature positions.

5.7.3 Conclusions

The BOR filter has very low computational demands. However, the experiments showed that even if the assumptions of bounded noise are satisfied, the filter may diverge. The problem is most likely associated with the distribution of the process and observation noise. The efficiency of the BOR filter was shown to be poor if a large probability mass of the noise distribution is concentrated near zero.

It is generally believed [66, 46] that the Gaussian distribution is the best approximation of the noise distribution. However, the assumption of bounded process and observation noise is justified by considering the physical sources of the noise, such as the slippage of the wheels or the bias in the laser rangefinder timers, and arguing that the noise sources are not expected to produce noise of infinite magnitude. This leads to the truncated Gaussian distribution.

The truncated Gaussian distribution satisfies the assumptions of the BOR filter, and therefore the BOR filter will operate consistently. However, the large concentration of probability mass of the truncated normal distribution near zero leads to very inefficient updates. This inefficient update combines with the predictions to produce a possibly diverging vehicle state estimate. The divergence of the BOR filter in the experiment was only avoided by reducing the estimated noise bounds, and therefore rendering the filter inconsistent.

The BOR filter could not be applied successfully to the experimental data.

5.8 The Relative Filter

This section examines the experimental results of the Relative (REL) filter. Section 5.8.1 concentrates on the operation of the Relative filter and compares it to the operation of the MAK filter described in Section 5.6.1. Section 5.8.2 examines the evolution of the map constructed by the REL filter.

5.8.1 Operation of the Relative Filter

The REL filter constructs a map of relative distances and angles between the observed features. The relative distances and angles are the states the Relative filter estimates; this is analogous to the positions of the features that the MAK filter estimates. However, the REL filter does not maintain an estimate of the vehicle state. As described in Section 4.3.1, the vehicle is localised based on the relative observations to features that have been associated with the nodes in the relative map. The accuracy of the localisation of the vehicle, relative to the features, depends only on the quality of the feature observations, and is not examined in the experiment. The same data set was used for the REL filter as was used for the PREK, the KNOK, the MAK and the BOR filters.

In the REL filter, the standard deviation of each state estimate is a monotonically non-increasing function of the timestep. This was derived in Section 4.3.3, and can be seen in Figures 5.75 to 5.78 for a specific relative distance and a specific relative angle estimate. The standard deviations appear to be consistent with the errors, this can be seen by the fact that the errors fall within the positive and negative bounds given by the standard deviations. Consistency was also found in the standard deviation of all relative distance and angle estimates when the error of the estimate was examined.

The same relative distance and angle estimates examined in Figures 5.75 to 5.78 were also extracted from the MAK filter so that the performance of the Relative filter could be compared to that of the MAK filter. The estimates from both filters are shown together in Figures 5.79 and 5.80 for the relative distance, and in Figures 5.81 and 5.82 for the relative angle. The key to the figures is given in the table below.

Figure 5.79 shows that the initial error in the estimates of the two filters is indistinguishable, and there is no significant difference between the standard deviations of the estimates. Towards the end of the experiment, a noticeable difference does emerge, as shown in Figure 5.80. Although there is a significant difference in the errors, the two filters have similar standard deviations, 0.017m compared to 0.020m, suggesting that the expected errors are similar. In the case of the relative angle comparison, the MAK filter is definitely superior to the Relative filter. This can be seen by comparing the final standard deviation of 0.024rad for the MAK filter and the final standard deviation of 0.0079rad for the Relative filter, as shown in Figure 5.82.

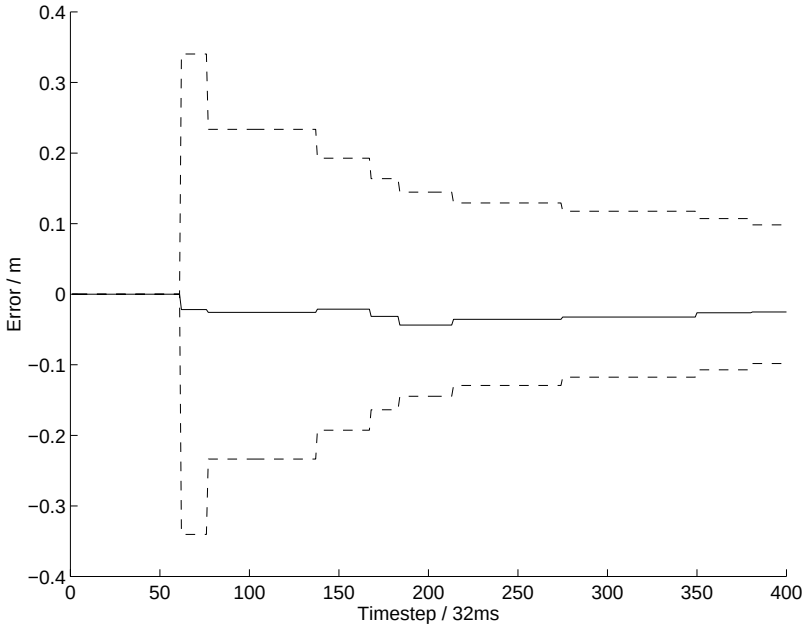

Figure 5.75: Initial stages of the error and the standard deviation of the estimate of the relative distance between a feature at (2.742m,-4.575m) and another feature at (0.914m,-4.575m). Before the relative distance estimate is initialised, the error and its standard deviation are both plotted as zero.

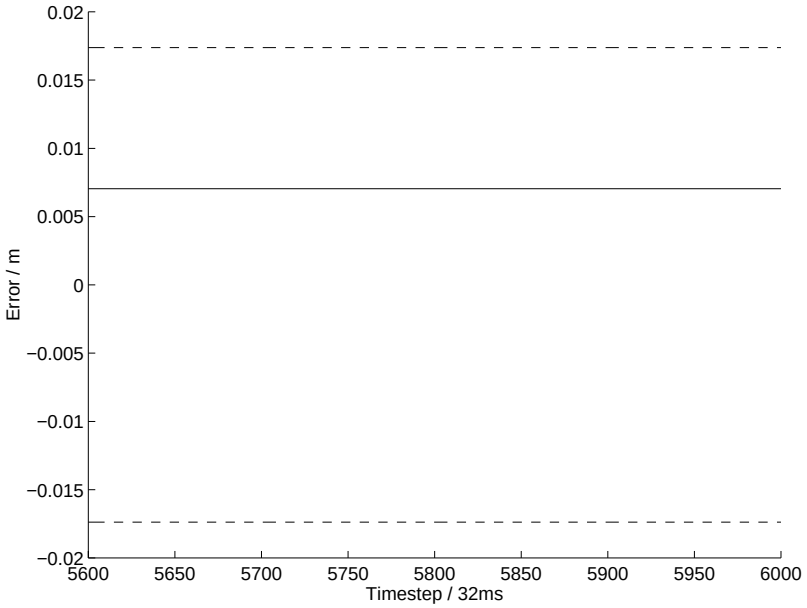

Figure 5.76: Final stages of the same error and standard deviation.

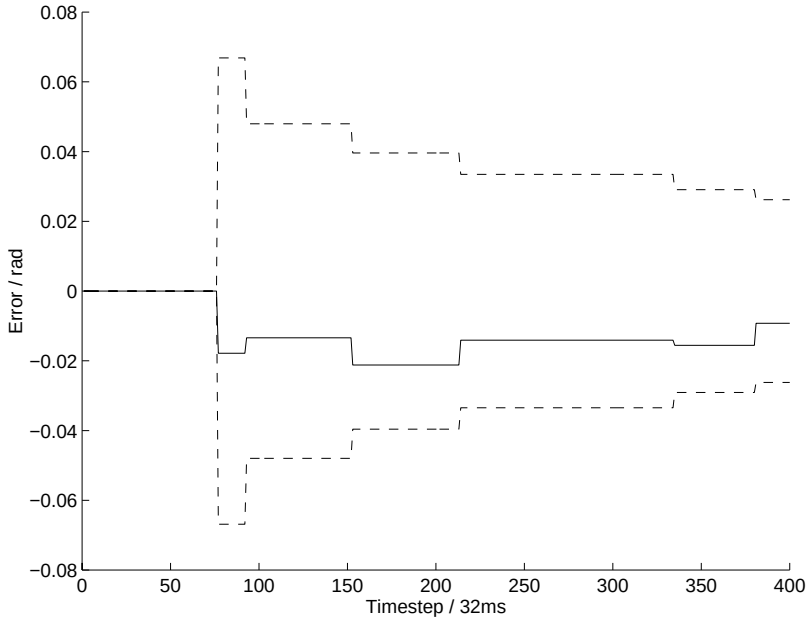

Figure 5.77: Initial stages of the error and the standard deviation of the estimate of the relative angle formed by features at (3.200m,-2.287m), (2.742m,-4.575m) and (0.914m,-1.830m).

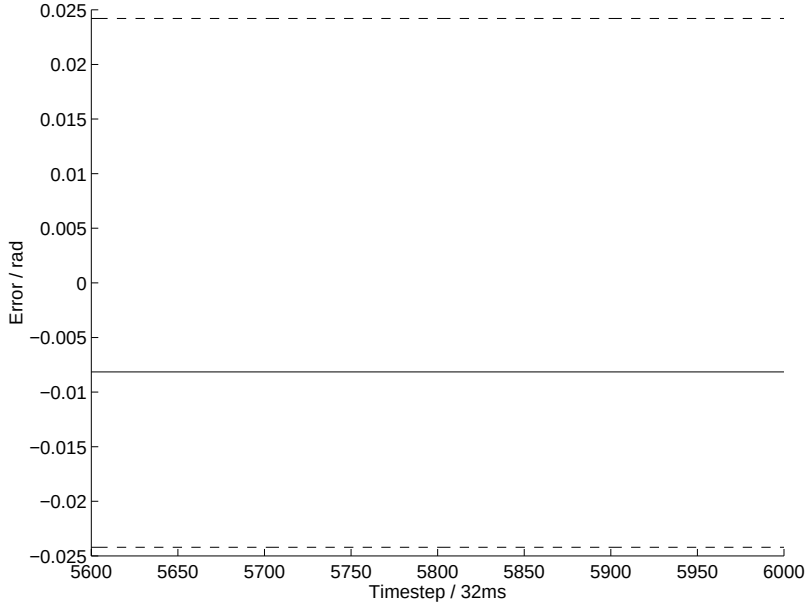

Figure 5.78: Final stages of the same error and standard deviation.

Extracting the appropriate relative distance and angle estimate form the MAK filter, it was found that MAK filter always produced estimates with standard deviations that were smaller than the standard deviations of the estimates obtained by the Relative filter. This is expected, since the REL filter maintains less information than the MAK filter.

—	error of the Relative filter estimate
- · -	error of the Augmented State Kalman filter estimate
- -	standard deviation of the Relative filter estimate
· · ·	standard deviation of the MAK filter estimate

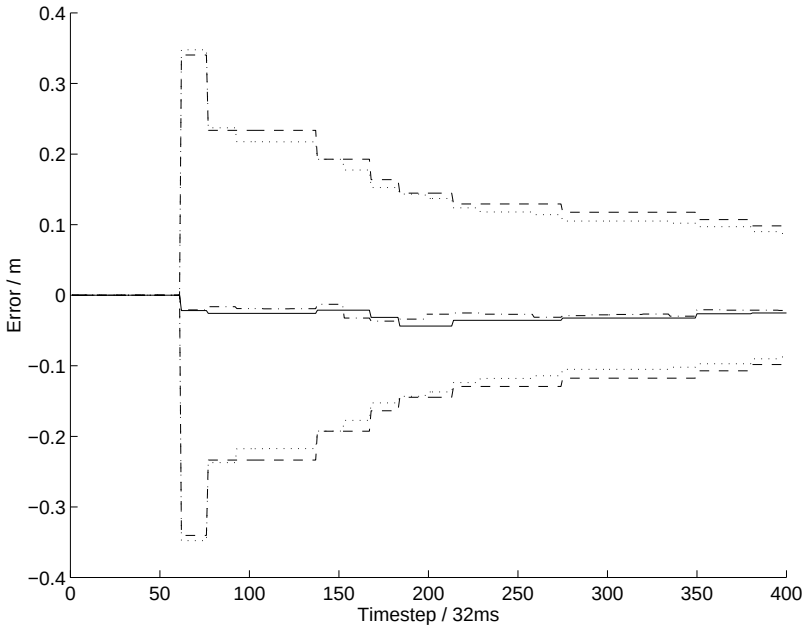

Figure 5.79: Initial stages of the error and the standard deviation of the estimate of the relative distance between a feature at (2.742,-4.575m) and another at (0.914m,-4.575), plotted for the REL and the MAK filter.

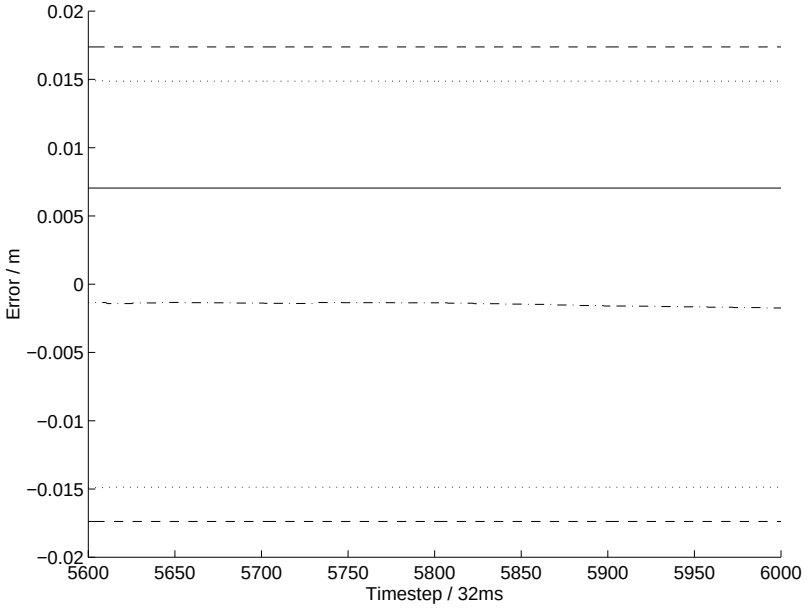

Figure 5.80: Final stages of the same errors and standard deviations.

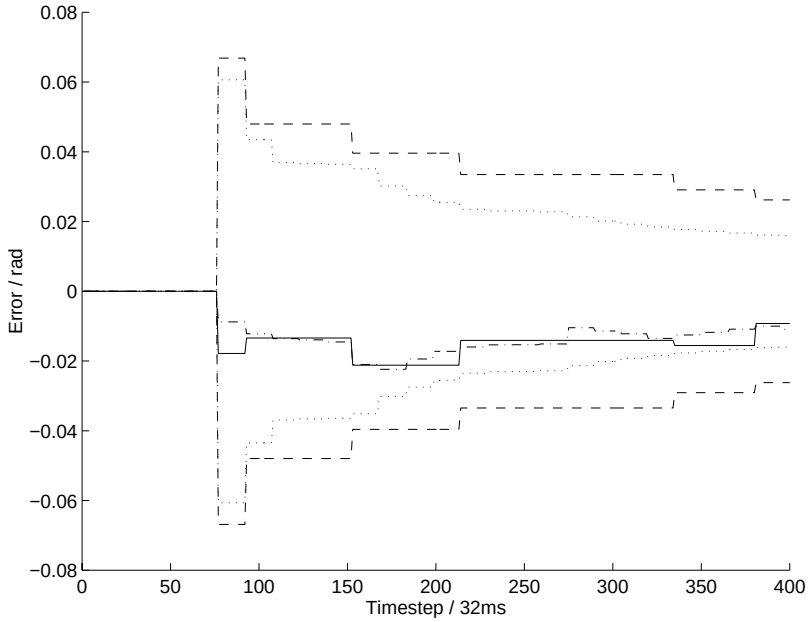

Figure 5.81: Initial stages of the error and the standard deviation of the estimate of the relative angle formed by features at (3.200m,-2.287m), (2.742m,-4.575m) and (0.914m,-1.830m), plotted for the REL and the MAK filter.

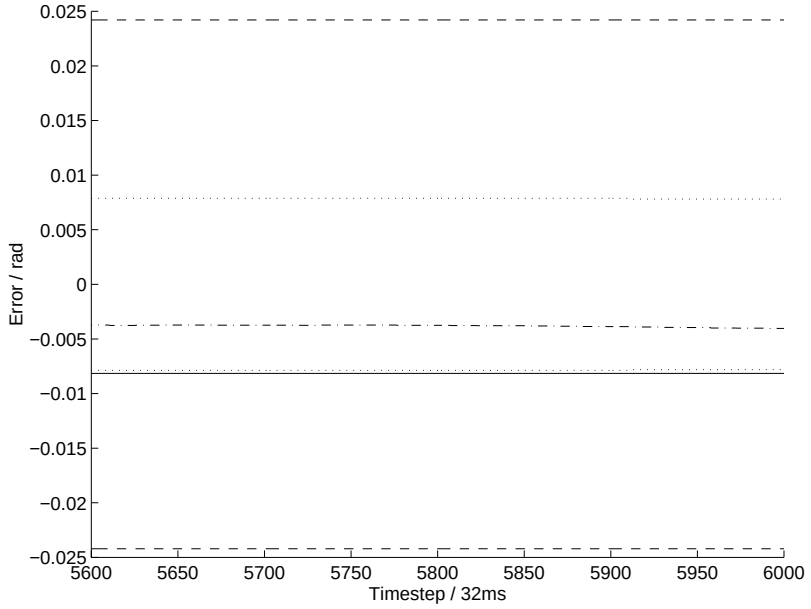

Figure 5.82: Final stages of the same errors and standard deviations.

5.8.2 The Evolution of the Map

In the Figures 5.83 to 5.92 the evolution of the Relative map is presented. The estimates of the relative distances are drawn as line segments connecting two nodes, with the thickness of the line segment inversely proportional to the variance of the estimate. Thick line segments represent a low standard deviation of the distance estimate. The standard deviation of the first relative distance estimate, formed between features at (0.914m,-1.830m) and (2.742m,0.000m), was 0.157m after initialisation, represented by the thin line segment in Figure 5.83. At the end of the experiment, the same estimate had a standard deviation of 0.016m, represented by the thick line segment in Figure 5.92.

The true positions of the features are marked with a 'x'. To draw the relative map, the absolute positions of the nodes were calculated by using the first observation to define the absolute coordinates of the first node and propagating the absolute position to all nodes via the links. Due to the errors in the links, errors in the relative distance and angle estimates, the absolute position estimates deteriorate with each propagation. The absolute position estimates were obtained only for the purpose of plotting the evolution of the map, and do not form part of the relative map.

The quality of the relative map, measured by the standard deviation of the estimates, is not necessarily worse in regions far from the origin, the starting point of the vehicle, than in regions close to the origin. The quality of a certain region of the relative map only depends on the number of observations that were used to update that region. The more observations that are made, the more updates are possible and the better will be the quality of the map in the region. This is the reason for the difference in the quality of the relative distance estimates shown by the thickness of the lines in Figure 5.92. The distance of the region from the origin is not significant. This is a direct consequence of the fact that relative states are estimated.

The deterioration of the map quality with the distance form the origin was observed in the MAK and the BOR filter and was attributed to the accumulation of process noise by the vehicle. In the case of the Relative filter, no estimate of the vehicle is maintained and therefore the distance traveled by the vehicle is not significant. It only gains significance if a relative distance or angel estimate is constructed for a group of features that were not observed simultaneously. In that case, the accumulation of process noise between the observations would affect the quality of the estimate, therefore the motion of the vehicle between the observations of the features in the group would be significant. However, the motion of the vehicle before the group was observed would still be irrelevant.

The sensitivity of the REL filter to the assumed variance of the process and observation noise, the tuning parameters in the filter, was lower than the sensitivity of the MAK and the BOR filter

to the parameters. Certain parameter settings produced gating failures in the MAK and the BOR filters, while no gating failures were found in the REL filter for the same set of parameters. This is probably associated with the fact that the MAK and the BOR filter used the absolute position of feature estimates to gate, whereas the REL filter used relative positions.

In the MAK and the BOR filters, the estimate errors are propagated from the vehicle to the features, from the features to the vehicle, and from one feature to another feature. In the REL filter, the estimate errors are not propagated from one relative state estimate to another relative state estimate. Therefore, the relative state estimates are not as sensitive to the process and observation noise as the position estimates in the MAK and the BOR filters. Hence the REL filter is more stable than the MAK and the BOR filters with respect to the process and observations noise.

In Figure 5.89, the feature added at (2.958m,1.140m) has not been linked to one of its closest neighbours at (2.742m,0.000m), but has been linked with a remote feature at (6.350m,-1.753m). This is not agreement with the strategy described in Section 4.3.8. The reason for this is that there was no observation in the buffer of the feature at (2.742m,0.000m) when the new feature at (2.958m,1.140m) was initialised. This could have been prevented by a larger buffer, however at the cost of an increase in the storage and the computational burden. A better strategy is to initialise the missing relative distance estimate when an observation of both features is available. However, the current implementation of the REL filter does not initialise relative distance estimates between features already in the map.

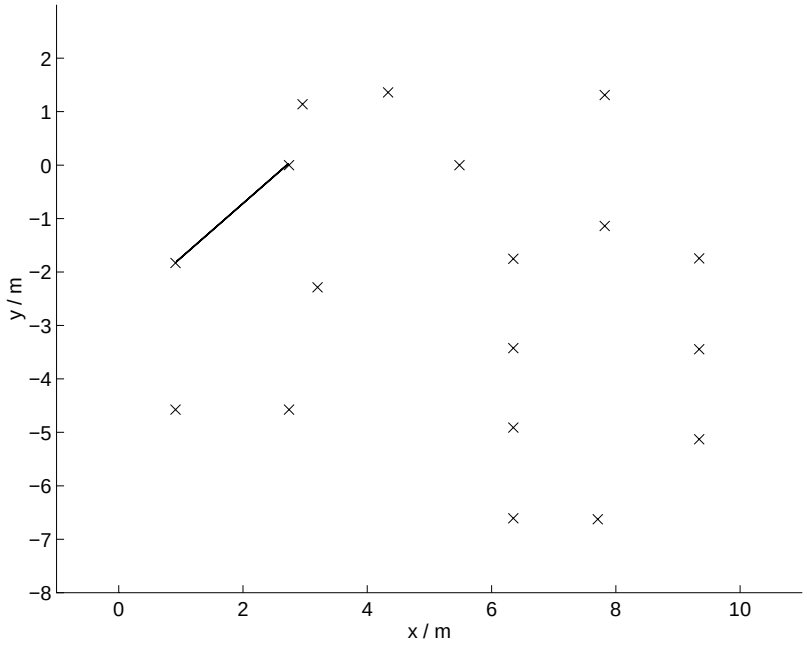

Figure 5.83: Timestep 1, only after two distinct features have been observed, located at (0.914m,-1.830m) and (2.742m,0.000m), is the relative map initiated.

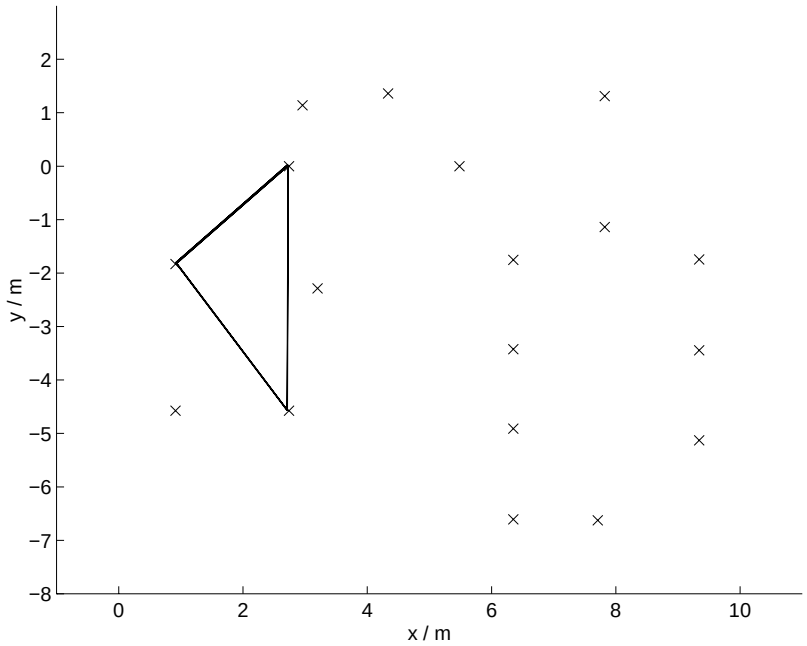

Figure 5.84: Timestep 46, a third feature is observed, located at (2.742m,-4.575m), and a fully connected map is formed.

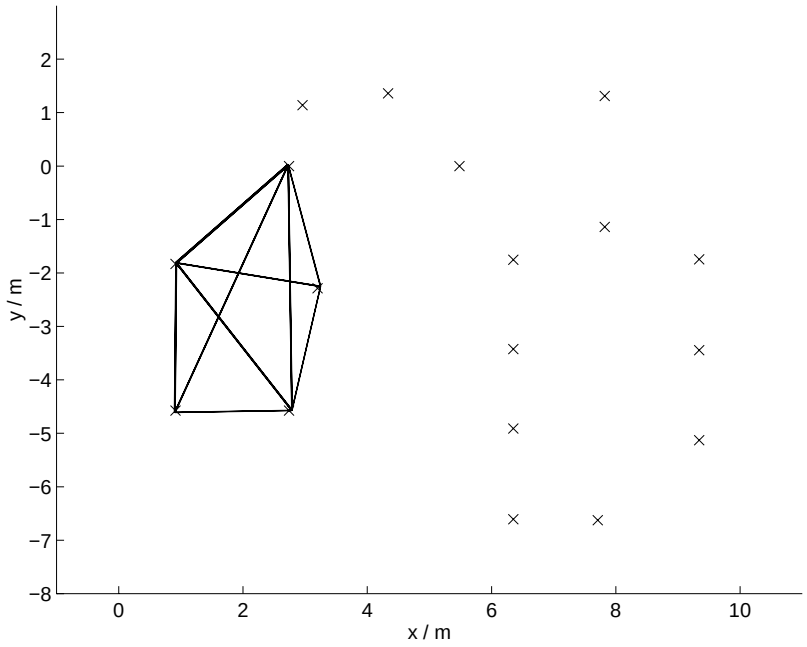

Figure 5.85: Timestep 76, the fifth feature, located at (3.200m,-2.287m), is connected only to the three closest neighbours. The relative map is now so large that not all possible links are made.

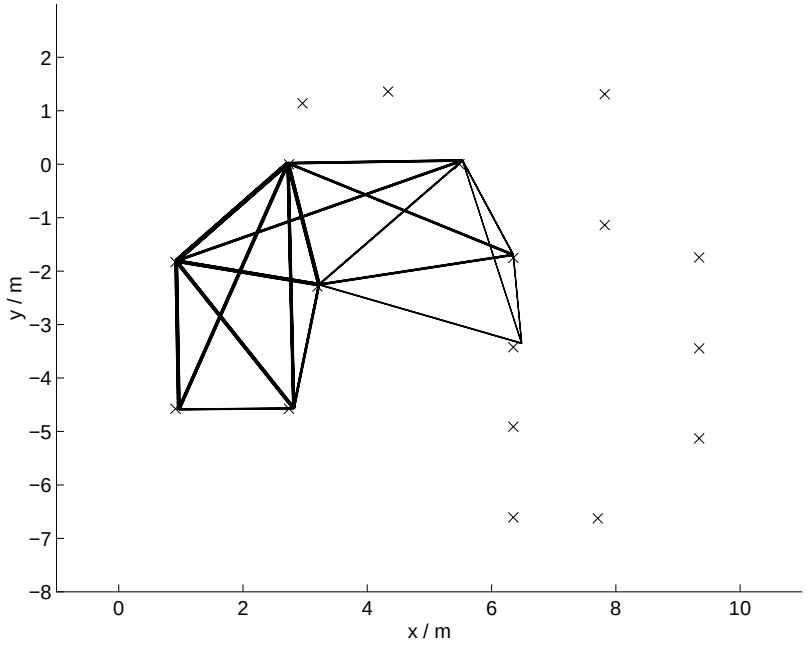

Figure 5.86: Timestep 410, a new feature has been added at (6.350m,-3.424m) and linked to the three closest neighbours. As the absolute position has to be propagated further down the mesh, it deteriorates as can be seen in the case of the newly added feature.

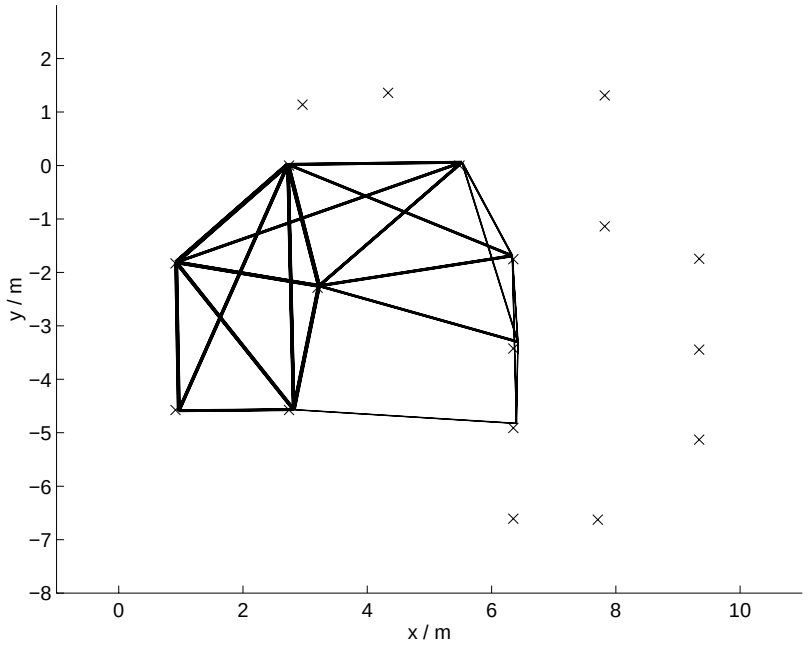

Figure 5.87: Timestep 577, the feature located at (6.350m,-4.911m) has been added to the map. Since it is joined to one of the first features initialised, it has access to a better quality absolute position estimate. Its absolute position estimate is not significantly worse than that of the previous feature.

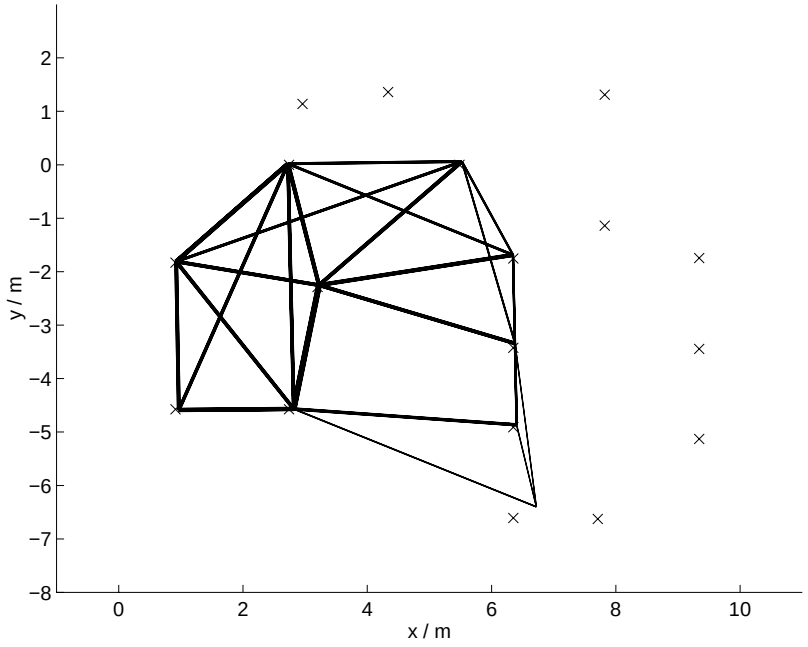

Figure 5.88: Timestep 987, a new feature has been added at (6.350m,-6.607m). The feature is the double pole of the railing, already identified in Section 5.6.2 as a feature that cannot be treated as a point feature. The large error in the absolute position estimate is therefore expected.

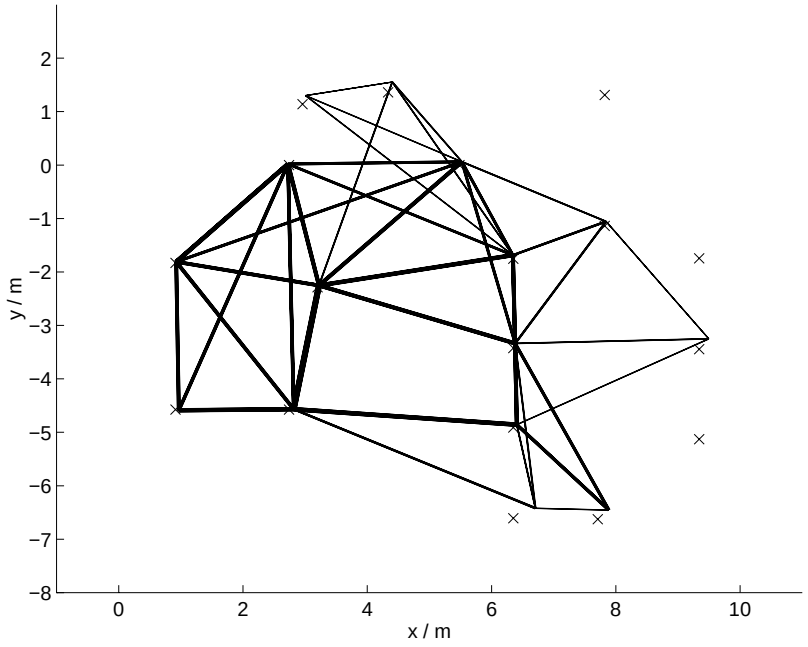

Figure 5.89: Timestep 1457, the feature added at (2.958m,1.140m) has not been linked to one of its closest neighbours at (2.742m,0.000m). The reason for this is that there was no observation in the buffer of the feature at (2.742m,0.000m) when the new feature at (2.958m,1.140m) was initialised. Comparing the thickness of the line segments indicates how the standard deviations change according to the region.

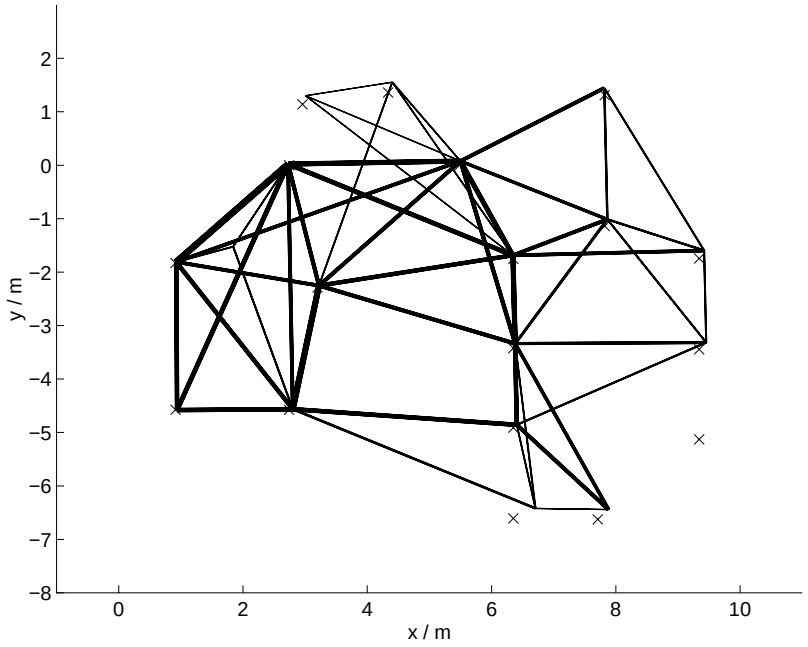

Figure 5.90: Timestep 2549, the 'ghost' feature identified in the MAK and the BOR filters also appears in the REL filter.

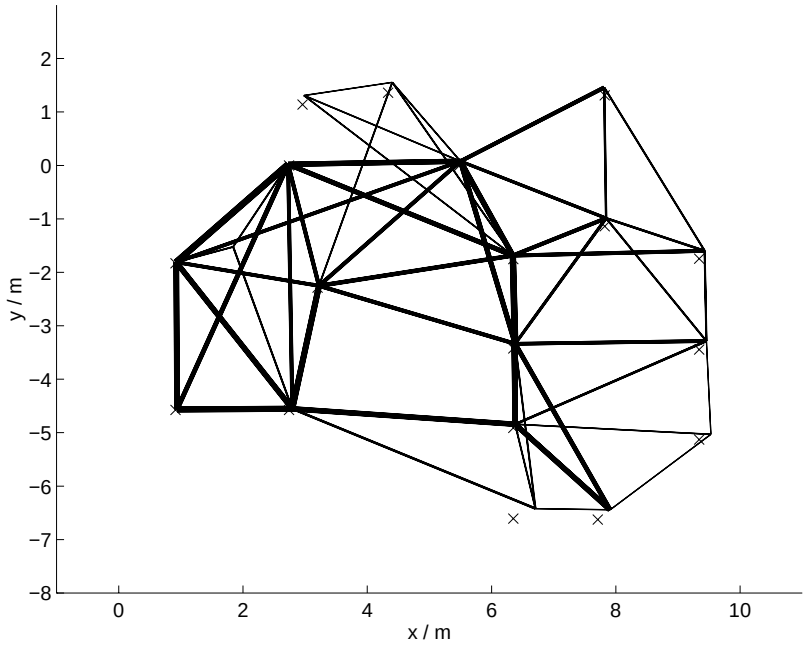

Figure 5.91: Timestep 3687, the Relative filter can successfully extend previously constructed parts of the map at a later stage, as seen by the new feature added at (-9.342m,-5.128m).

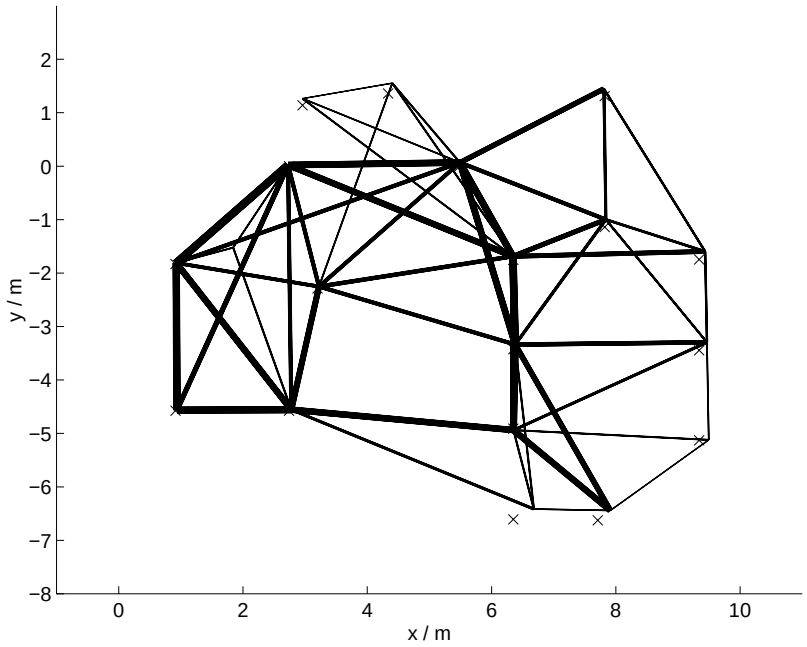

Figure 5.92: Timestep 6000, the Relative filter has reached the end of the logged data. The unusual appearance of the links associated with the feature at (2.958m,1.140m) has not changed. The link to the feature at (2.742m,0.000m) could have been added at a later stage, but new links between already mapped features are not constructed by the current version of the filter.

5.8.3 Conclusions

The Relative filter produced a relative map of the features, consisting of estimates of the relative distance and relative angle between features. The estimates produced were consistent with the true relative distances and angles.

The main advantage of the REL filter over the MAK filter is that it has a significantly lower computation and storage requirement than the MAK filter. In the REL filter, the storage requirement scales linearly with the number of features in the map, and the computation time is independent of the number of features. In contrast, the MAK filter requires storage and computation of at least order N^2 for N features.

An advantage of the REL filter over both the MAK and the BOR filters is that the quality of the map constructed does not have to deteriorate with the distance form the origin. This was associated in Section 5.8.2 with the fact that relative states are estimated. The low sensitivity of the REL filter to the tuning parameters, exhibited in a more robust gating performance, is probably also due to using relative estimates.

It should be pointed out, however, that low sensitivity to tuning parameters and the uniform map quality are achieved due to the relative framework. The data in the MAK filter can be transformed into a relative framework as well; this was done in Section 4.3.9 when relative distance and angle estimates were extracted from the MAK filter. The extracted estimates were found to have lower standard deviations than the estimates formed by the Relative filter. These could have been used for the purpose of gating, and would have probably eliminated the sensitivity advantage of the Relative filter.

It is also possible that feature estimates with large standard deviations in the MAK filter can lead to the extraction of relative estimates with small standard deviations, if the feature estimates are highly correlated. Therefore, the deterioration of the map quality in the MAK filter with the distance form the origin could still lead to a relative map, formed by extraction, of uniform quality. Hence a low gating sensitivity and a uniform map quality can also be achieved by the MAK filter if relative estimates are extracted.

The disadvantage of the Relative filter over the MAK filter is that the Relative filter does not produce absolute position estimates. However, if a relative framework can be adopted, the Relative filter is superior to the MAK filter due to its lower computation and memory requirement.

5.9 Summary

This chapter examined the performance of the MAK, the BOR and the Relative filter in a real map building scenario. The MAK and the Relative filter produced consistent maps (the true positions, relative distances and angles were all consistent with the mean and the variances of the estimates), whereas the BOR filter showed signs of inconsistency (the true feature positions often were beyond the bounds of the estimate).

The MAK filter produced a map of feature estimates that had low standard deviations near the starting point of the vehicle and high standard deviations in regions remote to the starting point. This distribution was due to the accumulation of process noise by the vehicle as it traveled further from its starting point to initialise new estimates. The distribution persisted during the entire experiment.

The standard deviations of the vehicle position estimate varied according to the region of the map. Areas of feature estimates with low standard deviations resulted in low standard deviations of the vehicle position estimate, and similarly areas of feature estimates with high standard deviations resulted in high standard deviations of the vehicle position estimate. During the operation of the MAK filter, the standard deviations of all feature estimates decreased, apparently approaching a common limit.

Although the MAK filter requires storage and computational resources at least of order N^2 for a map of N features, it does construct a converging and consistent map.

This was not the case for the BOR filter. It was suggested that the distribution of the noise may prohibit the successful operation of the BOR even if the noise is bounded. The inefficiency of the update in the BOR filter associated with certain noise distributions leads to the divergence of the filter if the true noise bounds used in the filter. If bounds lower than the true bounds are used the BOR filter produces inconsistent results. It was not possible to apply the BOR filter successfully.

The Relative filter does produce a consistent and convergent map. It produces a relative map, consisting of relative distance and angle estimates, which can be a disadvantage if absolute position estimates are desired. The vehicle position is localised relative to the features and not in terms of an absolute reference frame. However, the Relative filter requires storage of order N only and computational resources independent of N. This is a major advantage of the Relative filter.

Chapter 6

Summary and Conclusions

6.1 Introduction

This Chapter summarises the work presented in the thesis and draws some conclusions about the contributions made in this thesis. The summary is given chapter by chapter in Section 6.2. Section 6.3 reviews the contributions of the thesis. Section 6.4 identifies some of the limitations of the work and suggests future research areas.

6.2 Thesis Summary

In **Chapter 1** localisation was identified as a vital competency of Autonomous Guided Vehicles (AGVs). It is was argued that the AGV must be able to build a map of its environment and localise within it in order to achieve true "autonomous behaviour". This leads to the Simultaneous Localisation and Map Building (SLAM) problem. The solution to the SLAM problem was identified as a "Holy Grail" of AGV research.

Chapter 2 developed the necessary background to the solution of the SLAM Problem. Estimation using the Kalman filter was described. State vectors for the vehicle and the features in the environment were defined, and mathematical models of the vehicle and the sensor were presented. The observation of a feature relative to the vehicle was examined. The covariance of the innovation, the covariance between the innovation and the vehicle estimate, and the covariance between the innovation and the observed feature estimate were all shown to be dependent on the covariance between the vehicle estimate and the observed feature estimate. This result led to the identification of three special cases: localisation in a known environment, building a map from a vehicle of known position, and simultaneous localisation and map building. The application of the Kalman filter

in the first two cases was shown to be straightforward, but the application in the third case was identified as more complicated, establishing the SLAM problem in the estimation sense.

In **Chapter 3** the probabilistic estimation solution to the SLAM Problem was examined in detail. The vehicle state vector was augmented with the feature state vectors in Section 3.2, leading to the Map Augmented Kalman (MAK) filter. The augmentation allowed the existence of correlations between errors in different state estimates to be demonstrated in Section 3.3.

The error in the vehicle state estimate is correlated with the error in the feature state estimate for every feature, and the error in any feature estimate is correlated with the error in any other feature estimate.

The feature augmented vehicle state approach was shown to permit the Kalman filter to maintain a measure of these correlations via the covariance matrix. This was identified as necessary in order to produce estimation errors that are consistent with their variances. It was shown that the Kalman filter cannot be applied successfully if the feature augmented vehicle state approach is not used.

The Map Augmented Kalman (MAK) filter is the only consistent formulation of the problem of estimating the state of the vehicle and all the features

However, it was also identified that the feature augmented approach leads to a significant storage requirement and computational burden.

For an environment of N features, the storage and the computational demand of the MAK filter grows as N^2.

It was further shown in Section 3.3 that, if the error in an unobserved feature estimate is correlated with error in the observed feature estimate or is correlated with the error in the vehicle estimate, the unobserved feature estimate is updated. Therefore, correlations not only have to be considered to achieve consistency of the estimation, but also contribute directly to the efficiency of the estimation process.

Section 3.5 considered the evolution of the map. Section 3.5.1 showed that the determinant of the covariance of each feature estimate and any group of feature estimates, in particular the whole map, is a monotonically non-increasing function of the time step. Section 3.5.1 also showed that the variance of any dimension of the estimate of any feature is similarly a monotonically non-increasing function of the time step. These results were interpreted in Section 3.6 in terms of the map certainty.

The map constructed by the MAK filter will never become less certain.

It was further shown in Section 3.5.2 that the determinant of the covariance of the vector distance estimate between any two features is also a monotonically non-increasing function of the time step,

as is the variance of individual dimensions of the vector estimate. This was interpreted as a non-decreasing map rigidity.

> The map cannot become less rigid during the map building process.

Finally, in Section 3.7, a limit to the achievable map accuracy was demonstrated. It was argued in Section 3.7.2 that the initial uncertainty in the vehicle position estimate is passed onto the feature estimates during initialisation. Section 3.7.3 exploits the relative nature of the observations to show that the uncertainty common to the vehicle and the feature estimates cannot be filtered away with more observations, and therefore represents a lower limit to the achievable vehicle and feature estimate covariances. In Section 3.7.4 it was argued that only a non-relative observation, such as external vehicle position or terrain information would allow the limit to be surpassed.

> The variance of the feature position estimates has a lower limit given by the initial variance of the vehicle position. This is interpreted as a limit in the map accuracy that can be achieved with observations of the features made relative to the vehicle, and can only be surpassed by incorporating external terrain or vehicle position data.

Chapter 4 addressed the storage and computational burden of the Kalman filter approach identified in Section 3.3. Two alternative solutions to the SLAM Problem were introduced in Section 4.1. The first solution, described in Section 4.2, is the Bounded Region (BOR) filter. The BOR filter makes the assumption of bounded process and observation noise, as described in Section 4.2.1. This allows the BOR filter to formulate bounded estimates of the vehicle state and the feature states, where a bounded estimate of a state is a bounded volume of the state space within which the true state is guaranteed to lie. It is argued in Section 4.2.1 that the BOR filter update is the intersection of two bounded estimates, and therefore the BOR filter need not consider correlations between estimate errors. This leads to a very simple solution to the SLAM Problem, with very low storage and computational requirements.

> The storage requirement of the BOR filter was shown to increase linearly with the number of features and the computational burden was shown not to increase with the number of features.

The aim for simplicity also motivated the choice of the particular parametrisation of the bounded estimate, described in Section 4.2.3. The bounded estimate was defined as a minimum and a maximum bound on each dimension of the vehicle coordinates, and similarly a minimum and a maximum bound on each dimension of the feature coordinates. The bounded estimate for the orientation of the vehicle was defined as two bearings within which the true orientation of the vehicle must lie.

The details of forming predictions and updates for the BOR filter were described in Sections 4.2.5–4.2.10, with an overview given in Section 4.2.11.

In the discussion of Section 4.2.12 it was pointed out that the BOR filter does not deliver a mean state estimate, which may be an inconvenience. However, it was argued that if a small enough bound on the state estimate can be obtained, the SLAM Problem is solved.

The second alternative solution to the SLAM Problem, described in Section 4.3, is the Relative (REL) filter. The REL filter makes the same assumptions about the process and observation noise as the MAK filter, but estimates relative distances and angles between features rather than absolute locations. It was argued in Section 4.3.1 that observation of relative states, such as the relative distances and angles between features, does not depend on the vehicle state. Consequently, such observations are referred to as direct observations of a state. It was further shown in Section 4.3.1 that a direct observation of a state allows the state to be updated individually, because it is not necessary to consider correlations between the errors in different state estimates. This leads to a very simple solution to the SLAM Problem, with very low storage and computational requirements.

> The storage requirement of the REL filter was shown to increase linearly with the number of features and the computational burden was shown not to increase with the number of features.

Sections 4.3.4 – 4.3.8 described how to construct direct observations when the features needed to construct a relative distance or angle measurement are not observed simultaneously. It was shown in Sections 4.3.4 – 4.3.8 that a REL filter can also be formulated if the vehicle is moving while it is observing a relative state.

In **Chapter 5** experimental evaluations of the MAK, the BOR and the REL filters were presented. Section 5.2 described the environment and Section 5.3 described the sensor. In Section 5.3.1 it was described how range and bearing was extracted from the raw sensor data. Section 5.4 described the mobile robot used as the test vehicle.

To verify the models, preliminary investigations were performed, described in SectionPreliminary Investigations. These investigations addressed the predictions made using the vehicle model, Section 5.5.1, and also addressed the overall operation of the vehicle model, the sensor model, and the feature extraction algorithm, Section 5.5.2.

Section 5.6 examined the MAK filter performance. The results presented in Section 5.6.1 indicate that the accuracy of the vehicle localisation is not uniform throughout the environment.

> The MAK filter can localise the vehicle best, equivalent to a low vehicle estimate variance, when the vehicle is near its starting point. The variance of the vehicle estimate increases the further the vehicle travels from the starting point.

This was linked to the observation that the variance of the feature estimates are also not uniform throughout the environment.

> The variance of the feature estimates produced by the MAK filter are lowest for features near the starting point of the vehicle.

This was explained in Section 5.6.2 as the result of the vehicle accumulating process noise on its way to initialise estimates of remote features.

In the conclusions presented in Section 5.6.3, it was identified that the experiment supported the theory developed in Chapter 3. In particular, the variance of feature estimates were non-increasing and exhibited a lower limit.

Section 5.7 examined the performance of the BOR filter. Section 5.7.1 identified that the BOR filter exhibited a similar behaviour to the MAK filter.

> The BOR filter can localise the vehicle best, equivalent to a small estimate bound, when the vehicle is near its starting point. The bounds on the vehicle estimate increase the further the vehicle travels from the starting point.

As in the MAK filter, this was linked to the observation that the bounds on the feature estimates are not uniform throughout the environment.

> The bounds on the feature estimates produced by the BOR filter are smallest for features near the starting point of the vehicle.

In Section 5.7.2 this was explained as the result of the accumulation of process noise by the vehicle on its way to initialise estimates of remote features. Section 5.7.1 also examined the efficiency of the BOR filter.

> The distribution of the noise has an effect on the efficiency of the BOR filter updates. In particular, the truncated Gaussian distribution, though meeting the assumption of a bounded noise distribution, leads to very inefficient updates.

This was identified in the conclusions presented in Section 5.7.3 as the principle reason for the failure of the BOR filter.

The experimental evaluation of the REL filter was presented in Section 5.8. In Section 5.8.1 it was observed that the variance of the relative states produced by the REL filter never increase, in agreement with the derivations presented in Section 4.3.3.

The variance of the estimates of relative distances and angles were larger in the REL filter than the variance of the same estimates extracted from the MAK filter. The largest difference was observed in the estimate of the relative angles. This is probably due to the fact that the estimation of relative angles requires the fusion of the more information than the estimation of relative distances, and consequently suffers more from the restricted information available to the REL filter compared to the MAK filter.

The evolution of the relative map examined in Section 5.8.2 exhibited no particular difference between the variance of estimates in certain regions of the map and the variance of estimates in other regions of the map.

The quality of the relative map is uniform over the map and does not deteriorate with the distance from the starting point of the vehicle.

This was linked to the fact that the observation of the states in the REL are not dependent on the vehicle state, and therefore estimation of the states does not depend on how much process noise is accumulated prior to the observation of a state.

In the conclusions in Section 5.8.3, the low sensitivity of the filter to the tuning parameters was also linked to the fact that the REL filter operates in terms of relative states. However, it was also pointed out that the low sensitivity to tuning parameters could also have been achieved by the MAK filter if relative states were extracted and used to gate the observations.

6.3 Contributions

6.3.1 Theoretical Investigation of the SLAM Problem

The theoretical investigations presented are a major contribution of the thesis. Part of these investigations developed the Simultaneous Localisation and Map Building (SLAM) problem in estimation theoretic sense, allowing the problem to be mathematically analysed.

The investigations identified the presence of correlations between the errors in all feature estimates and the vehicle estimate, and identified that these correlations are essential to the SLAM problem. They must be considered when estimating the state of the vehicle and the features in order for the estimate errors to be consistent with the estimate covariances. It further identified that the correlations allow the estimates of unobserved features to be updated.

In particular, it was shown that the Map Augmented Kalman (MAK) filter is the only consistent formulation of the probabilistic estimation of the vehicle and feature states. However, the MAK filter was also identified as computationally impractical.

6.3.2 The Evolution of the MAK filter

The nature of the MAK filter was elucidated by demonstrating that the determinant of the covariance matrix of the feature estimates cannot increase. In particular, various estimates involving the features were considered. It was deduced that the determinant of the covariance matrix of individual feature estimates, of groups of feature estimates, and all feature estimates do not increase.

The determinant of the covariance matrix was interpreted as representing a confidence volume of the state estimate, thus the certainty in each feature in the map, in all possible regions of the map, and in the map as a whole was shown not to decrease. The variance of individual dimensions of the feature estimates were also shown to be non-increasing, again showing the non-decreasing certainty in the map.

The rigidity of the map, defined as the confidence in the relative distance estimates between different features, was also examined. The determinant of the covariance matrix of the relative distance vector was shown to be non-increasing for any two features in the map. Therefore the uncertainty in the relative distance estimate between any two features cannot increase, and thus the map rigidity cannot decrease. The variance of individual dimensions of the relative distance vector estimate was also shown to be non-increasing for any two features, again showing the rigidity of the map.

6.3.3 The Limit of the Map Accuracy

The map accuracy, defined as the variance of the feature estimates, was shown to have a theoretical limit. The theoretical limit was derived as the initial variance in the vehicle state estimate with the help of two insights into the SLAM Problem. The first insight identified that the initialisations of the feature estimates lead to each initialised feature estimate inheriting the initial variance of the vehicle estimate. The second insight relates to the Jacobians of the observation model associated with the SLAM Problem. Two insights were combined and the theoretical limit in the map accuracy was deduced. This cannot be surpassed using observations of the features made relative to the vehicle. It was argued that only a non-relative observation, such as the inclusion of external terrain or vehicle position data would allow the limit to improved.

6.3.4 The BOR Filter

A contribution of the thesis was also made in terms of formulating computationally practical solutions to the SLAM Problem. The first of these contributions is the BOR filter. The theory behind the BOR filter was developed. The assumptions of bounded noise made by the BOR filter, which are different to the assumptions made by the Kalman filter, were presented and justified. The use of these assumptions to formulate a filter is not an original contribution. However, it was shown how the assumptions lead to computationally feasible solutions to the SLAM Problem. Exploiting assumptions of bounded noise in order to arrive at a computationally feasible solution to the SLAM Problem is a contribution of the thesis.

Algorithmic details of the BOR filter were devised and presented in the thesis, and a complete

implementation of a BOR filter was also developed.

6.3.5 The REL filter

The second contribution of a computationally practical solution to the SLAM Problem is the REL filter.

The theory behind the REL filter was developed. The computational advantage of the REL filter was shown to result from the relative representation of the environment the filter employs. The development of the relative approach to the solution of the SLAM Problem is a contribution of the thesis.

Algorithmic details of the REL filter were devised and presented in the thesis, and a complete implementation of a REL filter was also developed.

6.3.6 Experimental Investigations

The final contributions of the thesis are in terms of the experimental investigations of the MAK, the BOR and the REL filters.

The contributions associated with the MAK filter include the verification of the theoretical developments, in particular the stability of the map and the lower limit to the map accuracy.

Further, it was identified that the map quality is best near the starting point of the vehicle and deteriorates with the distance from starting point. Similarly, it was identified that the quality of the vehicle state estimate follows the quality of the observable feature state estimates. This findings were explained using estimation theoretic arguments.

The contributions associated with the BOR filter include the observation of a similar map quality distribution in the BOR filter as in the MAK filter. The feature state estimates have a better quality near the starting point of the vehicle and deteriorate with distance. Again, the quality of the vehicle estimate follows the quality of the observable feature state estimates. These observations were explained using arguments relating to the operation of the BOR filter.

A further contribution associated with the BOR filter is the explanation of its inefficiency, depending on the distribution of the noise. The explanation was given in terms of the operation of the BOR filter and demonstrated in a controlled experiment. Further, it was concluded that the BOR filter may fail even if the assumptions made by the filter are not violated. Moreover, it was argued that this was probably the situation encountered in the experiment.

The contributions associated with the REL filter include the observation of the stability of its estimates, supporting the theory developed for the REL filter. Further, it was observed that the map quality does not exhibit the distribution encountered in the MAK and the BOR filter. The quality

of the relative distance and angle estimates do not decrease with the distance travelled form the starting point of the vehicle. This was explained by differentiating between the type of observations used in the REL filter and the type of observations used in the MAK and the BOR filter.

The relative approach to the SLAM Problem was also identified as more robust to tuning parameters.

6.4 Future Work

Future work in the solution to the SLAM Problem can be envisaged in several areas. Some of these areas are identified below.

6.4.1 Improvement of the REL Filter

The current implementation of the REL filter was designed only as a proof-of-concept, and is therefore lacking in several ways. For example, it does not initialise relative estimates between features that are already in the map. In certain situations, such as shown in Figure 5.89, this capability would be desirable.

Moreover, there is no concept of estimate management. However, the pruning of features from the map that fail to be observed over long periods of time is needed if the environment exhibits some dynamic behaviour. Similarly, linking new features to a fixed number of nearest neighbours in the map is not necessarily the best strategy. The estimate management process could decide on how many links to set up and to which features the new feature will be linked, depending on the quality of the available relative observations or the density of estimates already maintained in the region.

A REL filter operating on observations from which relative states can be extracted is more desirable than the current implementation. Using predictions of the relative motion of the vehicle between the observations introduces additional errors, which can be avoided by extracting the measurements from single observations. However, this requires a fast sensor.

The REL filter could also be improved by using more sophisticated environment models than point features. Two point features are more easily confused than two geometric primitives, such as corners or planes. This is due to the additional information contained in geometric primitives, such as orientation, that can help to differentiate between such features. A map consisting of geometric primitives is therefore potentially more robust.

Improvements to the BOR filter could also be investigated, however they are not considered here since the low efficiency of the bounded region approach has been very discouraging.

6.4.2 A Complete AGV Implementation

As was pointed out in Section 4.3.10, the operation of the REL filter restricts the navigational system to a relative framework. It remains to be investigated if such a relative framework can be used in actual applications. Of particular interest would be an investigation into a path planner in a relative framework.

Therefore, the proof-of-concept of the REL filter needs to be taken further.

Appendix A

Matrix Theory

This appendix provides essential matrix theory results for the derivations in this thesis. The matrix theory developed in this appendix is not a contribution. Some results were obtained in the course of the research (for example Theorem 13), but were subsequently found in the literature. The results and their proofs have been included for completeness.

A.1 Definitions

A matrix $\mathbf{A} = [a_{ij}] \in \mathbf{C}^{n\times n}$ is said to be Hermitian if $\mathbf{A} = \mathbf{A}^*$, where $\mathbf{A}^* \equiv \overline{\mathbf{A}}^T$ and $\overline{\mathbf{A}}$ is the complex conjugate of $\mathbf{A}$.

A Hermitian matrix $\mathbf{A} \in \mathbf{C}^{n\times n}$ is said to be positive definite (pd) if $\mathbf{x}^*\mathbf{A}\mathbf{x} > 0$ for all nonzero $\mathbf{x} \in \mathbf{C}^n$. $\mathbf{A}$ is said to be positive semidefinite (psd) if $\mathbf{x}^*\mathbf{A}\mathbf{x} \geq 0$.

A matrix $\mathbf{A} \in \mathbf{C}^{n\times n}$ is said to be normal if $\mathbf{A}^*\mathbf{A} = \mathbf{A}\mathbf{A}^*$.

A matrix $\mathbf{U} \in \mathbf{C}^n$ is said to be unitary if $\mathbf{U}^*\mathbf{U} = \mathbf{I}$. If, in addition , $\mathbf{U} \in \mathbf{R}^n$, $\mathbf{U}$ is said to be real orthogonal.

A.2 Basic Properties Of Matrices

The determinant of a matrix $\mathbf{A} = [a_{ij}] \in \mathbf{C}^{n\times n}$ can be defined as

$$\det(\mathbf{A}) = \sum_{\sigma} sgn(\sigma) \prod_{i=1}^{n} a_{i\sigma(i)}$$

Where the sum is taken over the $n!$ permutations σ of n items $\{1, \ldots, n\}$ and $sgn(\sigma)$ is +1 or -1 depending whether the minimum number of pairwise interchanges necessary to achieve the permutation starting from $\{1, \ldots, n\}$ is even or odd.
From the definition the following results are easily established

$$\begin{aligned}
\det[\beta_1\mathbf{a}_1 \ \ldots \ \beta_n\mathbf{a}_n] &= \beta_1 \ldots \beta_n \det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_n] \\
\det[\mathbf{a}_{\sigma(1)} \ \ldots \ \mathbf{a}_{\sigma(n)}] &= sgn(\sigma) \det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_n] \\
\det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_k + \mathbf{a}'_k \ \ldots \ \mathbf{a}_n] &= \det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_k \ \ldots \ \mathbf{a}_n] + \det[\mathbf{a}_1 \ \ldots \ \mathbf{a}'_k \ \ldots \ \mathbf{a}_n] \\
\det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_n] &= 0 \ \text{(if dependent columns are present)}
\end{aligned}$$

Using these results, the following important expression is obtained [88]

$$\begin{aligned}
\det(\mathbf{AB}) &= \det[\beta_{11}\mathbf{a}_1 + \cdots + \beta_{n1}\mathbf{a}_n \ \ldots \ \beta_{1n}\mathbf{a}_1 + \cdots + \beta_{nn}\mathbf{a}_n] \\
&= \sum_{\sigma} \beta_{\sigma(1),1} \ldots \beta_{\sigma(n),n} \det[\mathbf{a}_{\sigma(1)} \ \ldots \ \mathbf{a}_{\sigma(n)}] \\
&= \sum_{\sigma} sgn(\sigma) \beta_{\sigma(1),1} \ldots \beta_{\sigma(n),n} \det[\mathbf{a}_1 \ \ldots \ \mathbf{a}_n] \\
\det(\mathbf{AB}) &= \det(\mathbf{A}) \det(\mathbf{B}) \qquad \text{(A.1)}
\end{aligned}$$

In the Laplace expansion [41] the determinant is defined inductively for $\mathbf{A} = [a_{ij}] \in \mathbf{C}^{n\times n}$. Let $\mathbf{A}_{ij} \in \mathbf{C}^{(n-1)\times(n-1)}$ denote the submatrix of $\mathbf{A}$ obtained by deleting row i and column j. Then

$$\begin{aligned}
\det(\mathbf{A}) &= \sum_{j=1}^{n} (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij}) \\
&= \sum_{i=1}^{n} (-1)^{i+j} a_{ij} \det(\mathbf{A}_{ij})
\end{aligned}$$

From this it follows that if $\mathbf{A} \in \mathbf{C}^{n\times n}$ and columns $i_1, \ldots, i_k$ are unit vectors $\mathbf{e}_{i_1}, \ldots, \mathbf{e}_{i_k}$, then $\det(\mathbf{A})$ is equal to the principal minor of $\mathbf{A}$ obtained by striking rows and columns $i_1, \ldots, i_k$. This result will now be used to derive another result in matrix theory.

Consider the characteristic equation of $\mathbf{A}$ with columns $\mathbf{a}_1, \ldots, \mathbf{a}_n$ and let $\mathbf{e}_1, \ldots, \mathbf{e}_n$ be unit vectors, then

$$\begin{aligned}
c(\mu) &\equiv \det(\mu\mathbf{I} - \mathbf{A}) \\
&= (-1)^n \det(\mathbf{A} - \mu\mathbf{I}) \\
&= \det([\mathbf{a}_1 - \mu\mathbf{e}_1 \ \mathbf{a}_2 - \mu\mathbf{e}_2 \ \ldots \ \mathbf{a}_n - \mu\mathbf{e}_n]) \qquad \text{(A.2)}
\end{aligned}$$

The determinant of a matrix is a homogeneous linear function of its columns, therefore Equation A.2 can be expressed as the sum of 2^n determinants having columns of the form $\mathbf{a}_i$ or $-\mu\mathbf{e}_i$. There are $\binom{n}{r}$ ways of choosing determinants involving r columns of the form $-\mu\mathbf{e}_i$. These determinants can be simplified. The μ factors combine resulting in $(-\mu)^r$ and the remaining unit vector columns (and their corresponding rows) are deleted resulting in a principal minor of order $n-r$ and leaving the determinant unchanged. All the 2^n determinants identified above are included if r is taken from n to 0. Thus if c_r is the sum of all principle minors of $\mathbf{A}$ of order r, then [54]

$$\begin{aligned}
c(\mu) &= \mu^n - c_1\mu^{n-1} + c_2\mu^{n-2} - \cdots + (-1)^n c_n \\
&= (\mu - \lambda_1)(\mu - \lambda_2) \ldots (\mu - \lambda_n) \\
c_n &= \det(\mathbf{A}) \\
&\Rightarrow \det(\mathbf{A}) = \prod_{r=1}^{n} \lambda_r \qquad \text{(A.3)}
\end{aligned}$$

Following a similar argument another useful result is obtained

$$\begin{aligned}\det(\mathbf{A}+\mathbf{I}) &= \det([\mathbf{a}_1+\mathbf{e}_1\ \mathbf{a}_2+\mathbf{e}_2\ldots\mathbf{a}_n+\mathbf{e}_n]) \\ &= 1+c_1+c_2+\cdots+c_n \\ &= (1+\lambda_1)(1+\lambda_2)\ldots(1+\lambda_n) \\ \Rightarrow\ \det(\mathbf{A}+\mathbf{I}) &= \prod_{r=1}^{n}(\lambda_r+1)\end{aligned}$$

Theorem 1 (Schur) *Given $\mathbf{A}\in\mathbf{C}^{n\times n}$ with eigenvalues $\lambda_1,\ldots,\lambda_n$ in any prescribed order, there is a unitary matrix $\mathbf{U}\in\mathbf{C}^{n\times n}$ such that*

$$\mathbf{U}^*\mathbf{A}\mathbf{U}=\mathbf{T}=[t_{ij}]$$

is upper triangular, with diagonal entries $t_{ij}=\lambda_i,\ i=1,\ldots,n$. If $\mathbf{A}\in\mathbf{R}^{n\times n}$ and if all the eigenvalues of $\mathbf{A}$ are real, then $\mathbf{U}$ may be chosen to be real and orthogonal.

proof: [41]
Let $\mathbf{x}^{(1)}$ be a normalized eigenvector of $\mathbf{A}$ associated with eigenvalue λ_1. The nonzero vector $\mathbf{x}^{(1)}$ may be extended to a orthonormal basis $\mathbf{x}^{(1)},\mathbf{z}^{(2)},\ldots,\mathbf{z}^{(n)}$ of $\mathbf{C}^n$. A unitary matrix $\mathbf{U}_1$ can be formed with these orthonormal vectors arrayed left to right. The first column of $\mathbf{A}\mathbf{U}_1$ is $\lambda_1\mathbf{x}^{(1)}$ and hence

$$\mathbf{U}_1^*\mathbf{A}\mathbf{U}_1=\begin{bmatrix}\lambda_1 & * \\ \mathbf{0} & \mathbf{A}_1\end{bmatrix}$$

where $*$ represents arbitrary terms. The matrix $\mathbf{A}_1\in\mathbf{C}^{(n-1)\times(n-1)}$ has eigenvalues $\lambda_2,\ldots,\lambda_n$. Let $\mathbf{x}^{(2)}\in\mathbf{C}^{n-1}$ be a normalised eigenvector of $\mathbf{A}_1$ corresponding to λ_2, and repeat the procedure. Determine a unitary $\mathbf{U}_2\in\mathbf{C}^{(n-1)\times(n-1)}$ such that

$$\mathbf{U}_2^*\mathbf{A}_1\mathbf{U}_2=\begin{bmatrix}\lambda_1 & * \\ \mathbf{0} & \mathbf{A}_2\end{bmatrix}$$

and let

$$\mathbf{V}_2=\begin{bmatrix}1 & 0 \\ \mathbf{0} & \mathbf{U}_2\end{bmatrix}$$

The matrices $\mathbf{V}_2$ and $\mathbf{U}_1\mathbf{V}_2$ are then unitary, and

$$\mathbf{V}_2^*\mathbf{U}_1^*\mathbf{A}\mathbf{U}_1\mathbf{V}_2=\begin{bmatrix}\lambda_1 & * & * \\ 0 & \lambda_2 & * \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2\end{bmatrix}$$

The matrix $\mathbf{U}=\mathbf{U}_1\mathbf{V}_2\mathbf{V}_3\ldots\mathbf{V}_{n-1}$ is unitary and $\mathbf{U}^*\mathbf{A}\mathbf{U}$ yields the desired form. If all eigenvalues of $\mathbf{A}\in\mathbf{R}^{n\times n}$ happen to be real, then the corresponding eigenvectors can be chosen to be real and all the above steps may be carried out in real arithmetic, verifying the final assertion.

Theorem 2 *If $\mathbf{A}\in\mathbf{C}^{n\times n}$ is normal it is unitarily diagonalisable.*

$$\mathbf{A} = \mathbf{U}^*\Lambda\mathbf{U}$$

If $\mathbf{A} \in \mathbf{R}^{n\times n}$ *and if all the eigenvalues of* $\mathbf{A}$ *are real, then* $\mathbf{U}$ *may be chosen to be real and orthogonal.*

proof: [41]
Let $\mathbf{T} = [\mathbf{t}_{ij}] \in \mathbf{C}^{n\times n}$ be an upper triangular matrix which is unitarily equivalent to $\mathbf{A}$ (Theorem 1), that is $\mathbf{T} = \mathbf{U}^*\mathbf{A}\mathbf{U}$ for some unitary $\mathbf{U} \in \mathbf{C}^n$. $\mathbf{T}$ is then normal since

$$\begin{aligned}\mathbf{T}^*\mathbf{T} &= \mathbf{U}^*\mathbf{A}^*\mathbf{U}\mathbf{U}^*\mathbf{A}\mathbf{U}\\ &= \mathbf{U}^*\mathbf{A}^*\mathbf{A}\mathbf{U}\\ &= \mathbf{U}^*\mathbf{A}\mathbf{A}^*\mathbf{U}\\ &= \mathbf{U}^*\mathbf{A}\mathbf{U}\mathbf{U}^*\mathbf{A}^*\mathbf{U}\\ &= \mathbf{T}\mathbf{T}^*\end{aligned}$$

Since the (1,1) entry of $\mathbf{T}^*\mathbf{T}$ is the same as that of $\mathbf{T}\mathbf{T}^*$, it follows that

$$\overline{\mathbf{t}}_{11}\mathbf{t}_{11} = \mathbf{t}_{11}\overline{\mathbf{t}}_{11} + \sum_{j=2}^{n}\mathbf{t}_{1j}\overline{\mathbf{t}}_{1j} = |\mathbf{t}_{11}|^2 + \sum_{j=2}^{n}|\mathbf{t}_{1j}|^2$$

Hence $\mathbf{t}_{1j} = 0,\ j = 2,\ldots,n$. Since the (2,2) entry of $\mathbf{T}^*\mathbf{T}$ is the same as that of $\mathbf{T}\mathbf{T}^*$ it similarly follows that

$$\overline{\mathbf{t}}_{22}\mathbf{t}_{22} = \mathbf{t}_{22}\overline{\mathbf{t}}_{22} + \sum_{j=3}^{n}\mathbf{t}_{2j}\overline{\mathbf{t}}_{2j} = |\mathbf{t}_{22}|^2 + \sum_{j=3}^{n}|\mathbf{t}_{2j}|^2$$

Therefore $\mathbf{t}_{2j} = 0,\ j = 3,\ldots,n$. Extending this argument to all diagonal entries and using the fact that $\mathbf{T}$ is upper triangular leads to

$$\begin{aligned}\mathbf{t}_{ij} &= 0,\ j > i\ for\ i = 1,\ldots,n\\ \mathbf{t}_{ij} &= 0,\ j < i\ for\ i = 1,\ldots,n\end{aligned}$$

and hence $\mathbf{T}$ is diagonal ($\mathbf{T} = \Lambda$). If $\mathbf{A} \in \mathbf{R}^{n\times n}$ and all eigenvalues of $\mathbf{A}$ are real, then $\mathbf{U}$ may be chosen to be real and orthogonal (Theorem 1).

A.3 Basic Properties Of Hermitian Matrices

Theorem 3 *For a Hermitian matrix* $\mathbf{A} \in \mathbf{C}^{n\times n}$ *and a vector* $\mathbf{x} \in \mathbf{C}^n$ *the expression* $\mathbf{x}^*\mathbf{A}\mathbf{x}$ *is always a real number.*

proof:

$$\begin{aligned}\mathbf{x}^*\mathbf{A}\mathbf{x} &= \mathbf{x}^*\mathbf{A}^*\mathbf{x}\\ &= (\mathbf{x}^*\mathbf{A}\mathbf{x})^*\end{aligned}$$

Since $\mathbf{x}^*\mathbf{A}\mathbf{x}$ is a scalar and it is equal to its conjugate the result follows.

Theorem 4 (Rayleigh-Ritz) *Let* $\mathbf{A} \in \mathbf{C}^{n\times n}$ *be Hermitian, with eigenvalues ordered as* $\lambda_{min} = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n = \lambda_{max}$. *Then*

$$\lambda_1 \mathbf{x}^*\mathbf{x} \leq \mathbf{x}^*\mathbf{A}\mathbf{x} \leq \lambda_n \mathbf{x}^*\mathbf{x} \; for \; all \; \mathbf{x} \in \mathbf{C}^n$$

$$\lambda_{max} = \lambda_n = \max_{\mathbf{x}\neq 0} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \max_{\mathbf{x}^*\mathbf{x}=1} \mathbf{x}^*\mathbf{A}\mathbf{x}$$

$$\lambda_{min} = \lambda_1 = \min_{\mathbf{x}\neq 0} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \min_{\mathbf{x}^*\mathbf{x}=1} \mathbf{x}^*\mathbf{A}\mathbf{x}$$

proof: [41]
Since $\mathbf{A}$ is Hermitian it is normal and can be expressed as $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^*$, where $\mathbf{U} \in \mathbf{C}^{n\times n}$ is a unitary matrix and $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ as shown in Theorem 2. For any $\mathbf{x} \in \mathbf{C}^n$

$$\begin{aligned} \mathbf{x}^*\mathbf{A}\mathbf{x} &= \mathbf{x}^*\mathbf{U}\Lambda\mathbf{U}^*\mathbf{x} \\ &= (\mathbf{U}^*\mathbf{x})^* \Lambda (\mathbf{U}^*\mathbf{x}) \\ &= \sum_{i=1}^{n} \lambda_i \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 \end{aligned}$$

Hence

$$\lambda_{min} \sum_{i=1}^{n} \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 \leq \mathbf{x}^*\mathbf{A}\mathbf{x} = \sum_{i=1}^{n} \lambda_i \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 \leq \lambda_{max} \sum_{i=1}^{n} \mid (\mathbf{U}^*\mathbf{x})_i \mid^2$$

and since $\mathbf{U}$ is unitary

$$\sum_{i=1}^{n} \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 = (\mathbf{U}^*\mathbf{x})^* (\mathbf{U}^*\mathbf{x}) = \mathbf{x}^*\mathbf{U}\mathbf{U}^*\mathbf{x} = \mathbf{x}^*\mathbf{x}$$

Therefore

$$\lambda_1 \mathbf{x}^*\mathbf{x} = \lambda_{min}\mathbf{x}^*\mathbf{x} \leq \mathbf{x}^*\mathbf{A}\mathbf{x} \leq \lambda_{max}\mathbf{x}^*\mathbf{x} = \lambda_n \mathbf{x}^*\mathbf{x}$$

If $\mathbf{x}$ is an eigenvector of $\mathbf{A}$ corresponding to λ_1, then $\mathbf{x}^*\mathbf{A}\mathbf{x} = \lambda_1\mathbf{x}^*\mathbf{x}$, and similarly for λ_n. Hence

$$\max_{\mathbf{x}\neq 0} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \lambda_n$$

If $\mathbf{x} \neq 0$

$$\frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \left(\frac{\mathbf{x}}{\sqrt{\mathbf{x}^*\mathbf{x}}}\right)^* \mathbf{A} \left(\frac{\mathbf{x}}{\sqrt{\mathbf{x}^*\mathbf{x}}}\right)$$

and since

$$\left(\frac{\mathbf{x}}{\sqrt{\mathbf{x}^*\mathbf{x}}}\right)^* \left(\frac{\mathbf{x}}{\sqrt{\mathbf{x}^*\mathbf{x}}}\right) = 1$$

it follows that

$$\max_{\mathbf{x}^*\mathbf{x}=1} \mathbf{x}^*\mathbf{A}\mathbf{x} = \lambda_n$$

The arguments for λ_1 are similar.

If $\mathbf{x}$ is restricted to be orthogonal to $\mathbf{u}_{n-k+1}, \ldots, \mathbf{u}_n$ where $\mathbf{U} = [\mathbf{u}_1 \ldots \mathbf{u}_n]$ and k is an integer such that $1 \leq k < n$, then

$$\mathbf{x}^*\mathbf{A}\mathbf{x} = \sum_{i=1}^{n} \lambda_i \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 = \sum_{i=1}^{n} \lambda_i \mid \mathbf{u}_i^*\mathbf{x} \mid^2 = \sum_{i=1}^{n-k} \lambda_i \mid \mathbf{u}_i^*\mathbf{x} \mid^2$$

$$\sum_{i=1}^{n-k} \lambda_i \mid \mathbf{u}_i^*\mathbf{x} \mid^2 \leq \lambda_{n-k} \sum_{i=1}^{n-k} \mid \mathbf{u}_i^*\mathbf{x} \mid^2 = \lambda_{n-k} \sum_{i=1}^{n} \mid (\mathbf{U}^*\mathbf{x})_i \mid^2 = \lambda_{n-k}\mathbf{x}^*\mathbf{x}$$

Equality is reached when $\mathbf{x} = \mathbf{u}_{n-k}$. Hence

$$\max_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{u}_{n-k+1},\ldots,\mathbf{u}_n}} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x} * \mathbf{x}} = \max_{\substack{\mathbf{x}=1 \\ \mathbf{x}\perp \mathbf{u}_{n-k+1},\ldots,\mathbf{u}_n}} \mathbf{x}^*\mathbf{A}\mathbf{x} = \lambda_{n-k} \tag{A.4}$$

Theorem 5 (Courant-Fischer) *Let $\mathbf{A} \in \mathbf{C}^{n\times n}$ be a Hermitian matrix with eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$, and let k be a given integer with $1 \leq k \leq n$. Then*

$$\min_{\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_{n-k}\in\mathbf{C}^n} \max_{\substack{\mathbf{x}\neq 0,\,\mathbf{x}\in\mathbf{C}^n \\ \mathbf{x}\perp \mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_{n-k}}} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \lambda_k$$

and

$$\max_{\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_{n-k}\in\mathbf{C}^n} \min_{\substack{\mathbf{x}\neq 0,\,\mathbf{x}\in\mathbf{C}^n \\ \mathbf{x}\perp \mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_{n-k}}} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \lambda_k$$

proof: [41]
Since $\mathbf{A}$ is Hermitian, express it as $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^*$ where $\mathbf{U}$ is unitary and $\Lambda = \mathrm{diag}\,(\lambda_1, \lambda_2, \ldots, \lambda_n)$. For $1 < k \leq n$ and $\mathbf{x} \neq 0$

$$\frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \frac{(\mathbf{U}^*\mathbf{x})^*\,\Lambda\,(\mathbf{U}^*\mathbf{x})}{\mathbf{x}^*\mathbf{x}} = \frac{(\mathbf{U}^*\mathbf{x})^*\,\Lambda\,(\mathbf{U}^*\mathbf{x})}{(\mathbf{U}^*\mathbf{x})^*\,(\mathbf{U}^*\mathbf{x})}$$

A change of variables $\{\mathbf{U}^*\mathbf{x} : x \in \mathbf{C}^n \text{ and } \mathbf{x} \neq 0\} = \{\mathbf{y} \in \mathbf{C}^n : \mathbf{y} \neq 0\}$ leads to

$$\begin{aligned}
\sup_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{w}_1,\ldots,\mathbf{w}_{n-k}}} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} &= \sup_{\substack{\mathbf{y}\neq 0 \\ \mathbf{y}\perp \mathbf{U}^*\mathbf{w}_1,\ldots,\mathbf{U}^*\mathbf{w}_{n-k}}} \frac{\mathbf{y}^*\Lambda\mathbf{y}}{\mathbf{y}^*\mathbf{y}} \\
&= \sup_{\substack{\mathbf{y}^*\mathbf{y}=1 \\ \mathbf{y}\perp \mathbf{U}^*\mathbf{w}_1,\ldots,\mathbf{U}^*\mathbf{w}_{n-k}}} \sum_{i=1}^{n} \lambda_i \mid y_i \mid^2 \\
&\geq \sup_{\substack{\mathbf{y}^*\mathbf{y}=1 \\ \mathbf{y}\perp \mathbf{U}^*\mathbf{w}_1,\ldots,\mathbf{U}^*\mathbf{w}_{n-k} \\ y_1=y_2=\cdots=y_{k-1}=0}} \sum_{i=1}^{n} \lambda_i \mid y_i \mid^2 \\
&= \sup_{\substack{|y_k|^2+|y_{k+1}|^2+\cdots+|y_n|^2=1 \\ \mathbf{y}\perp \mathbf{U}^*\mathbf{w}_1,\ldots,\mathbf{U}^*\mathbf{w}_{n-k}}} \sum_{i=1}^{n} \lambda_i \mid y_i \mid^2 \\
&\geq \lambda_k
\end{aligned}$$

The result holds for any $n-k$ vectors $\mathbf{w}_1,\ldots,\mathbf{w}_{n-k}$. As derived above, equality holds if the vectors $\mathbf{w}_i$ are chosen such that they equal to $\mathbf{u}_{n-k+1},\ldots,\mathbf{u}_n$. Therefore

$$\inf_{\mathbf{w}_1,\ldots,w_{n-k}} \sup_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{w}_1,\ldots,\mathbf{w}_{n-k}}} \frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} = \lambda_k$$

Since the extremum is reached, "inf" and "sup" are replaced with "min" and "max."

Theorem 6 (Weyl) *Let $\mathbf{A},\mathbf{B}\in\mathbf{C}^{n\times n}$ be Hermitian with the eigenvalues $\lambda_i(\mathbf{A})$, $\lambda_i(\mathbf{B})$, and $\lambda_i(\mathbf{A}+\mathbf{B})$ arranged in increasing order. For each $k=1,2,\ldots,n$ the following expression holds*

$$\lambda_k(\mathbf{A})+\lambda_1(\mathbf{B}) \leq \lambda_k(\mathbf{A}+\mathbf{B}) \leq \lambda_k(\mathbf{A})+\lambda_n(\mathbf{B})$$

proof: [41]
For any $\mathbf{x}\in\mathbf{C}^n$ Theorem 4 gives the bound

$$\lambda_1(\mathbf{B}) \leq \frac{\mathbf{x}^*\mathbf{B}\mathbf{x}}{\mathbf{x}^*\mathbf{x}} \leq \lambda_n(\mathbf{B})$$

For any $k=1,2,\ldots,n$ the following derivation can be made using Theorem 5

$$\begin{aligned}
\lambda_k(\mathbf{A}+\mathbf{B}) &= \min_{\mathbf{w},\ldots,\mathbf{w}_{n-k}\in\mathbf{C}^n} \max_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{w}_1,\ldots,\mathbf{w}_{n-k}}} \frac{\mathbf{x}^*(\mathbf{A}+\mathbf{B})\mathbf{x}}{\mathbf{x}^*\mathbf{x}} \\
&= \min_{\mathbf{w},\ldots,\mathbf{w}_{n-k}\in\mathbf{C}^n} \max_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{w}_1,\ldots,\mathbf{w}_{n-k}}} \left[\frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}}+\frac{\mathbf{x}^*\mathbf{B}\mathbf{x}}{\mathbf{x}^*\mathbf{x}}\right] \\
&\geq \min_{\mathbf{w},\ldots,\mathbf{w}_{n-k}\in\mathbf{C}^n} \max_{\substack{\mathbf{x}\neq 0 \\ \mathbf{x}\perp \mathbf{w}_1,\ldots,\mathbf{w}_{n-k}}} \left[\frac{\mathbf{x}^*\mathbf{A}\mathbf{x}}{\mathbf{x}^*\mathbf{x}}+\lambda_1(\mathbf{B})\right] \\
\lambda_k(\mathbf{A}+\mathbf{B}) &\geq \lambda_k(\mathbf{A})+\lambda_1(\mathbf{B})
\end{aligned}$$

A similar argument establishes the upper bound as well. Equation A.4 cannot be used since the eigenvectors of $\mathbf{A}+\mathbf{B}$, $\mathbf{A}$, and $\mathbf{B}$ are not necessarily the same. The min max operator over

$\mathbf{w}, \ldots, \mathbf{w}_{n-k} \in \mathbf{C}^n$ overcomes this problem. The inner max operator makes $\mathbf{x}$ the eigenvector corresponding to the largest eigenvalue of the expression that has not been excluded by the condition $\mathbf{x} \perp \mathbf{w}_1, \ldots, \mathbf{w}_{n-k}$. The outer min operator achieves the minimum when it sets $\mathbf{w}_1, \ldots, \mathbf{w}_{n-k}$ to the $n-k$ eigenvectors of the expression corresponding to the $n-k$ largest eigenvalues of the expression, thereby excluding these eigenvalues from the total set of n eigenvalues in the inner optimisation. The inner optimisation then leads to the $n-(n-k)=k$ eigenvalue.

Corollary. *Let $\mathbf{A}, \mathbf{B} \in \mathbf{C}^n$ be Hermitian. Assume that $\mathbf{B}$ is positive semidefinite and that the eigenvalues of $\mathbf{A}$ and $\mathbf{A}+\mathbf{B}$ are arranged in increasing order. Then*

$$\lambda_k(\mathbf{A}) \leq \lambda_k(\mathbf{A}+\mathbf{B}) \text{ for all } k = 1, 2, \ldots, n$$

proof: [41]
The result follows from the lower bound in Theorem 6 and the fact that $\lambda_1(\mathbf{B}) \geq 0$ (Theorem 8).

A.4 Basic Properties Of Positive Semidefinite Matrices

Theorem 7 *The sum of two positive psd matrices is a psd matrix.*

proof: [41]
Consider a matrix $\mathbf{M} \in \mathbf{C}^{n \times n}$ and two positive psd matrices $\mathbf{A}, \mathbf{B} \in \mathbf{C}^{n \times n}$:

$$\begin{aligned}
\mathbf{a}^*\mathbf{A}\mathbf{a} &\geq 0 \text{ ; for all } \mathbf{a} \in \mathbf{C}^n \\
\mathbf{b}^*\mathbf{B}\mathbf{b} &\geq 0 \text{ ; for all } \mathbf{b} \in \mathbf{C}^n \\
\mathbf{M} &= \mathbf{A}+\mathbf{B} \\
\mathbf{m}^*\mathbf{M}\mathbf{m} &= \mathbf{m}^*(\mathbf{A}+\mathbf{B})\mathbf{m} \\
&= \mathbf{m}^*\mathbf{A}\mathbf{m} + \mathbf{m}^*\mathbf{B}\mathbf{m} \\
&\geq 0 \text{ ; for all } \mathbf{m} \in \mathbf{C}^n \\
&\Rightarrow \mathbf{M} \text{ is psd}
\end{aligned}$$

Theorem 8 *All eigenvalues of a psd matrix are greater than or equal to zero, and all eigenvalues of a pd matrix are greater than zero.*

proof:
Consider a psd matrix $\mathbf{M} \in \mathbf{C}^{n \times n}$, and let $\mathbf{x} \in \mathbf{C}^n$ be an eigenvector $\mathbf{u}_i$ of $\mathbf{M}$, with associated eigenvalue λ_i:

$$\begin{aligned}
\mathbf{x}^*\mathbf{M}\mathbf{x} &\geq 0 \text{ ; for all } \mathbf{x} \in \mathbf{C}^n \\
\mathbf{u}_i^*\mathbf{M}\mathbf{u}_i &\geq 0 \\
\lambda_i \mathbf{x}^*\mathbf{x} &\geq 0 \\
\Rightarrow \lambda_i &\geq 0 \text{ ; for all } i = 1, \ldots, n
\end{aligned}$$

The result for a pd follows similarly by using $\mathbf{x}^*\mathbf{M}\mathbf{x} > 0$.

Theorem 9 *All psd matrices can be written as the product of a matrix and its Hermitian adjoint.*

proof:
Consider a psd matrix $\mathbf{M} \in \mathbf{C}^{n \times n}$. Since a psd matrix is Hermitian, it is also normal and can therefore be unitarily diagonalised (Theorem 2).

$$\begin{aligned}
\mathbf{M} &= \mathbf{U}\Lambda\mathbf{U}^* \\
&= \mathbf{U}\begin{bmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{bmatrix}\begin{bmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{bmatrix}\mathbf{U}^* \\
&= (\mathbf{U}\sqrt{\Lambda})(\mathbf{U}\sqrt{\Lambda})^*
\end{aligned}$$

If $\mathbf{M} \in \mathbf{R}^{n\times n}$, then by Theorem 2 $\mathbf{U}$ can be chosen to be real. Since all the eigenvalues of $\mathbf{M}$ are nonzero by Theorem 8, it follows that $(\mathbf{U}\sqrt{\Lambda}), (\mathbf{U}\sqrt{\Lambda})^* \in \mathbf{R}^{n\times n}$.

Theorem 10 *The product of a matrix and its Hermitian adjoint is a psd matrix.*

proof:[88]
Consider a matrix $\mathbf{A} \in \mathbf{C}^{n\times m}$ and any $\mathbf{x} \in \mathbf{C}^n$

$$\begin{aligned}
\mathbf{x}^*\mathbf{A}^*\mathbf{A}\mathbf{x} &= \mathbf{y}^*\mathbf{y} \text{ ; for a } \mathbf{y} \in \mathbf{C}^m \\
&= \sum_{i=1}^{m} |y_i|^2 \\
&\geq 0 \\
&\Rightarrow \mathbf{A}^*\mathbf{A} \text{ is psd}
\end{aligned}$$

Theorem 11 *If $\mathbf{A} \in \mathbf{C}^{n\times n}$ is psd then for any matrix $\mathbf{B} \in \mathbf{C}^{n\times m}$ $\mathbf{B}\mathbf{A}\mathbf{B}^*$ is psd.*

proof: Using Theorem 9 to write $\mathbf{A}$ as product of a matrix $\mathbf{C}$ and its Hermitian adjoint

$$\begin{aligned}
\mathbf{B}\mathbf{A}\mathbf{B}^* &= \mathbf{B}\mathbf{C}\mathbf{C}^*\mathbf{B}^* \\
&= (\mathbf{B}\mathbf{C})(\mathbf{B}\mathbf{C})^*
\end{aligned}$$

and $(\mathbf{B}\mathbf{C})(\mathbf{B}\mathbf{C})^*$ is psd by Theorem 10.

A.5 The Determinant Of Positive Semidefinite Matrices

Theorem 12 *If $\mathbf{A}, \mathbf{B} \in \mathbf{R}^{n\times n}$ are psd, then*

$$\det(\mathbf{A}+\mathbf{B}) \geq \det(\mathbf{A})$$

proof:

$$\begin{aligned}
\det(\mathbf{A}+\mathbf{B}) &= \prod_{i=1}^{n} \lambda_r(\mathbf{A}+\mathbf{B}) \text{ ; from Equation A.3} \\
&\geq \prod_{i=1}^{n} \lambda_r(\mathbf{A}) \text{ ; from Theorem 6} \\
\det(\mathbf{A}+\mathbf{B}) &\geq \det(\mathbf{A})
\end{aligned}$$

A.6 Principle Submatrices Of Positive Semidefinite Matrices

Theorem 13 *All principle submatrices of a psd matrix are psd matrices, and the diagonal entries of a psd matrix are non-negative.*

proof: [88]
Consider a psd matrix $\mathbf{M} \in \mathbf{C}^{n \times n}$.

$$\mathbf{x}^* \mathbf{M} \mathbf{x} \geq 0 \ ; \text{ for all } \mathbf{x} \in \mathbf{C}^n$$

$$\begin{bmatrix} \mathbf{x}_1^* & \mathbf{x}_2^* & \mathbf{x}_3^* & \dots & \mathbf{x}_n^* \end{bmatrix} \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} & \mathbf{M}_{13} & \dots & \mathbf{M}_{1n} \\ \mathbf{M}_{21} & \mathbf{M}_{22} & \dots & & \\ \mathbf{M}_{31} & \vdots & \ddots & & \\ \vdots & & & & \\ \mathbf{M}_{n1} & & & & \mathbf{M}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \geq 0$$

$$\begin{bmatrix} \mathbf{0} & \mathbf{x}_i^* & \mathbf{0} & \mathbf{x}_j^* & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} & \mathbf{M}_{13} & \dots & \mathbf{M}_{1n} \\ \mathbf{M}_{21} & \mathbf{M}_{22} & \dots & & \\ \mathbf{M}_{31} & \vdots & \ddots & & \\ \vdots & & & & \\ \mathbf{M}_{n1} & & & & \mathbf{M}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{x}_i \\ \mathbf{0} \\ \mathbf{x}_j \\ \mathbf{0} \end{bmatrix} \geq 0$$

$$\begin{bmatrix} \mathbf{x}_i^* & \mathbf{x}_j^* \end{bmatrix} \begin{bmatrix} \mathbf{M}_{ii} & \mathbf{M}_{ij} \\ \mathbf{M}_{ji} & \mathbf{M}_{jj} \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_j \end{bmatrix} \geq 0 \ ; \text{ for all } \mathbf{x}_i \text{ and } \mathbf{x}_j$$
$$\Rightarrow \begin{bmatrix} \mathbf{M}_{ii} & \mathbf{M}_{ij} \\ \mathbf{M}_{ji} & \mathbf{M}_{jj} \end{bmatrix} \text{ is psd}$$

The number of elements of $\mathbf{x}$ that are zeroed is arbitrary. Thus, given a psd matrix, forming a principal submatrix (by striking an arbitrary number of columns along with the corresponding rows) results in a psd matrix.

Choosing $\mathbf{x}$ to be nonzero only in dimension i it follows from $\mathbf{x}^* \mathbf{M} \mathbf{x} \geq 0$ that the i^{th} diagonal entry of $\mathbf{M}$ must be non-negative.

Bibliography

[1] B. Abraham and J. Ledolter. *Statistical Methods in Forecasting.* John Wiley & Sons, 1983.

[2] N. Ayache and O.D. Faugeras. Maintaining representations of the environment of a mobile robot. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 205–220. Springer-Verlag, 1990.

[3] M. Baeg, H. Hashimoto, F. Harashima, and J.B. Moore. Pose estimation of quadratic surface using surface fitting technique. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 204–209, August 1995.

[4] Y. Bar-Shalom and T.E. Fortman. *Tracking and Data Association.* Academic Press, 1988.

[5] J.O. Berger. *Statistical Decision Theory and Bayesian Analysis.* Springer-Verlag, 1985.

[6] R.S. Bertin and T.W. Pendleton. Using qaulitative maps to direct reactive robots. In *Proceedings of SPIE - The International Society for Optical Engineering*, pages 262–273, 1993.

[7] D.P. Bertsekas and I.B. Rhodes. Recursive state estimation for a set-membership description of uncertainty. *IEEE Transactions on Automatic Control*, 16:171–128, 1971.

[8] P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

[9] S. Betge-Brezet, P. Hebert, R. Chatila, and M. Devy. Uncertain map making in natural environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1048–1053, Minneapolis, Minnesota, April 1996.

[10] M. Betke and L. Guruits. Mobile robot localization using landmarks. *IEEE Transactions on Robotics and Automation*, 13(2):251–263, April 1997.

[11] S. Blackman. *Multiple-Target Tracking with Radar Applications.* Artech House, 1986.

[12] J. Borenstein and L. Feng. Gyrodometry: A new method for combining data from gyros and odometry in mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 423–428, Mineapolis, Minnesota, April 1996.

[13] K. Braden, C. Browning, H. Gelderloos, F. Smith, C. Marttila, and L. Vallot. Integrated INS/GPS for a manned return vehicle autoland application. In *Position Location and Navigation Symposium*, pages 74–82, 1990.

[14] M. Brady, S. Cameron, H. Durrant-Whyte, M. Fleck, D. Forsyth, A. Noble, and I. Page. Progress toward a system that can acquire pallets and clean warehouses. In *Proceedings of the International Symposium on Robotics Research*, pages 359–374, Santa Cruz, CA, August 1987.

[15] M. Brendt. *Neural Networks: An Introduction.* Springer-Verlag, 1990.

[16] V. Broman and M.J. Shensa. A compact algorithm for the intersection and approximation of n-dimensional polytopes. *Mathematics and Computers in Simulation*, 32:469–480, 1990.

[17] H.H. Buelthoff, J.J. Little, and H.A. Mallot. Neural mapping and parellel optical flow computation for autonomous navigation. *Neural Networks*, 1(1):482, 1988.

[18] D.E. Catlin. *Estimation, Control, and the Discrete Kalman Filter*, volume 71 of *Applied Mathematical Sciences*. Springer-Verlag, 1989.

[19] L. Chisci, A. Garulli, and G. Zappa. Recursive state bounding by parallelotopes. *Automatica*, 32(7):1049–1055, July 1996.

[20] I.J. Cox. Blanche: Position estimation for an autonomous robot vehicle. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 221–228. Springer-Verlag, 1990.

[21] I.J. Cox and G.T. Wilfong, editors. *Autonomous Robot Vehicles*. Springer-Verlag, USA, 1990.

[22] S.C. Crow and F.L. Manning. Differential GPS control of starcar 2. *Navigation: Journal of the Institute of Navigation*, 39(4):383–405, Winter 1992-93.

[23] Personal correspondence of the author with the Australian defense industry in August 1997.

[24] M. Csorba and H.F. Durrant-Whyte. New approach to map building using relative position estimates. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 3087, pages 115–125, Orlando, FL, April 1997.

[25] M. Csorba, J.K. Uhlmann, and H.F. Durrant-Whyte. A new approach to simultaneous localization and dynamic map building. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 2738, pages 26–36, Orlando, FL, April 1996.

[26] A. de la Escalera, L. Moreno, M.A. Salichs, and J.M. Armigol. Continuous mobile robot localization using structured light and a geometric map. *International Journal of Systems Science*, 27(8):771–782, August 1996.

[27] R. Deveza, D. Thiel, R.A. Russel, and A. Mackay-Sim. Odour sensing for robot guidance. *International Journal of Robotics Research*, 13(3):232–239, June 1994.

[28] H.F. Durrant-Whyte. An autonomous guided vehicle for cargo handling applications. *International Journal of Robotics Research*, 15(5):407–440, October 1996.

[29] A. Elfes. Sonar-based real-world mapping and navigation. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 233–249. Springer-Verlag, 1990.

[30] H. Evers and G. Kasties. Differential GPS in a real-time land vehicle environment - satellite based van carrier location system. *IEEE Aerospace and Electrical Systems Magazine*, 9(8):26–32, August 1994.

[31] E. Fogel and F. Huang. On the value of information in system identification – bounded noise case. *Automatica*, 18:229–238, 1982.

[32] C.W. Gardiner. *The Handbook of Stochastic Methods*. Springer-Verlag, 1983.

[33] J. Gonzalez, A. Stentz, and A. Ollero. A mobile robot iconic position estimator using a radial laser scanner. *Journal of Intelligent and Robotic Systems*, pages 161–179, 1995.

[34] G. Hager. Interval-based techniques for sensor data fusion. Technical report, GRASP Lab, University of Pennsylvania, 3401 Walnut Street, Philadelphia, PA 19104/6228, September 1990.

[35] G.D. Hager, S.P. Engelson, and S. Atiya. On comparing statistical and set-based methods in sensor data fusion. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 352–358, Atlanta, GA, 1993.

[36] U.D. Hanebeck, J. Horn, and G. Schmidt. On combining set theoretical and bayesian estimation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3081–3086, Minneapolis, MN, April 1996.

[37] U.D. Hanebeck and G. Schmidt. Set theoretical localization of fast mobile robots using an angle measurement technique. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1387–1394, Minneapolis, MN, April 1996.

[38] S.S. Haykin. *Neural Networks: A Comprehensive Foundation*. Maxwell Macmillan International, 1994.

[39] P. Hebert, S. Betge-Brezet, and R. Chatila. Decoupling odometry and exteroceptive perception in building a global map of a mobile robot: The use of local maps. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 757–764, Minneapolis, Minnesota, April 1996.

[40] S.K. Honey and M.S. White. Cartographic databases. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 250–258. Springer-Verlag, 1990.

[41] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 1992.

[42] L.D. Hostetler and R.D. Andreas. Nonlinear Kalman filtering techniques for terrain-aided navigation. *IEEE Transactions on Automatic Control*, 28(3):315–323, March 1983.

[43] G. Hyslop, D. Gerth, and J. Kraemer. GPS/INS integration on the standoff land attack missile (SLAM). In *Position Location and Navigation Symposium*, pages 407–412, 1990.

[44] A.H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, 1970.

[45] J.L. Jones and A.M. Flynn. *Mobile Robots*. A.K. Peters, Wellesley, MA, 1993.

[46] S.J. Julier. *Process Models for the Navigation of High Speed Land Vehicles*. PhD thesis, University of Oxford, 1997.

[47] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME Journal of Basic Engineering*, pages 35–45, March 1960.

[48] R.E. Kalman and R.S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME Journal of Basic Engineering*, pages 95–1–08, March 1961.

[49] K.Ebihara, T. Otani, and E. Kume. Position localisation for mobile robots using a colour image of equipment at nuclear plants. *Robotica*, 14(6):677–685, 1996.

[50] L. Kleeman. Optimal estimation of position and heading for mobile robots using ultrasonic beacons and dead reckoning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2582–2587, Nice, France, May 1992.

[51] D.J. Kriegman, E. Triendl, and T.O. Binford. A mobile robot: Sensing, planning and locomotion. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 450–458. Springer-Verlag, 1990.

[52] B.J. Kuipers and Y.T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*, 8:47–63, 1991.

[53] B.J. Kuipers and Y.T. Byun. A robust, qualitative approach to a spatial learning mobile robot. In S.S. Iyengar and A. Elfes, editors, *Autonomous Mobile Robots: Perception, Mapping, Navigation*, pages 353–362. IEEE Computer Science Press, 1991.

[54] P. Lancaster. *Theory of Matrices*. Academic Press, 1969.

[55] A. Lawrence. *Modern Inertial Technology*. Springer-Verlag, 93.

[56] J.J. Leonard. *Directed Sonar Sensing for Mobile Robot Navigation*. PhD thesis, University of Oxford, 1990.

[57] J.J. Leonard and H.F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, 1992.

[58] J.J Leonard, B.A. Moran, I.J. Cox, and M.L. Miller. Underwater sonar data fusion using an efficient multiple hypothesis algorithm. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2995–3002, 1995.

[59] S. Li, S. Nagata, and S. Tsuji. A navigational system based upon panoramic representation. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 142–147, Pittsburgh, PA, August 1995.

[60] J.H. Lim and D.W. Cho. Multipath Bayesian map construction model from sonar data. *Robotica*, 14(5):527–540, 1996.

[61] E.N. Macleod and M. Chiarella. Navigation and control breakthrough for automation technology. In *Proceedings of SPIE - The International Society for Optical Engineering*, pages 57–68, Boston, 1993.

[62] D. Maksarov and J.P. Norton. State bounding with ellipsoidal set description of the uncertainty. Technical report, School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, 1995.

[63] D. Maksarov and J.P. Norton. State bounding with minimal volume ellipsoids. Technical report, School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, 1996.

[64] L. Matthies, E. Gat, R. Harrison, B. Wilcox, R. Volpe, and T. Litwin. Mars microrover navigation performance evaluation and enhancement. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 433–440, 1995.

[65] L. Matthies and T. Kanade. The cycle of uncertainty and constraint in robot perception. In *Proceedings of the International Symposium on Robotics Research*, pages 327–335, Santa Cruz, CA, August 1987.

[66] P.S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, 1979.

[67] M. Milanese and G. Belforte. Estimation theory and uncertainty intervals evaluation in the presence of unknown but bounded errors: Linear families of models and estimates. *IEEE Transactions on Automatic Control*, 27:408–414, 1982.

[68] S.H. Mo and J.P. Norton. Fast and robust algorithm to compute exact polytope parameter bounds. *Mathematics and Computers in Simulation*, 32:481–493, 1990.

[69] H. Moravec. High resolution maps from wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 116–121, 1985.

[70] H.P. Moravec. The Stanford cart and the CMU rover. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 407–419. Springer-Verlag, 1990.

[71] V. Morellas, J. Minners, and M. Donath. Implementation of real time spatial mapping in robot systems through self-organizing neural networks. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 277–284, Pittsburgh, PA, August 1995.

[72] P. Moutarlier and R. Chatila. Stochastic multisensor data fusion for mobile robot localization and environment modelling. In *Proceedings of the International Symposium on Robotics Research*, pages 85–94, 1989.

[73] G.P.R. Netzler. Ndc news. Technical Report 6, Netzler & Dahlgren Company AB, SE-430 40 SARO, Sweden, 1994.

[74] A. Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, second edition, 1991.

[75] D. Pomerlau. Ralph: Rapidly adapting lateral position handler. In *Proceedings of the International Symposium on Robotics Research*, Detroit, 1995.

[76] R. Poovendran, S. Speigle, S. Srinivasan, S. Raghavan, and R. Chellappa. Qualitative landmark recognition using visual cues. In *Proceedings of SPIE - The International Society for Optical Engineering*, pages 74–83, Orlando, FL, April 1997.

[77] R.F. Poppen. Using partial or nonstandard GPS solutions in automotive navigation. In *Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation*, pages 59–62, 1993.

[78] W.D. Rencken. Concurrent localisation and map building for mobile robots using ultrasonic sensors. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 2192–2197, Yokohama, Japan, 1993.

[79] R.A. Russel. Mobile robot guidance using a short lived heat trail. *Robotica*, 11(5):427–431, 1993.

[80] R.A. Russel, D. Thiel, and A. Mackay-Sim. Sensing odour trails for mobile robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2672–2677, San Diega, CA, May 1994.

[81] A. Sabater and F. Thomas. Set membership approach to the propagation of uncertain geometric information. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2718–2723, 1991.

[82] F.C. Schweppe. Recursive state estimation: Unknown but bounded errors and system inputs. *IEEE Transactions on Automatic Control*, 13:22–28, 1968.

[83] J.S. Shamma and K.Y. Tu. Approximate set-valued observers for nonlinear systems. *IEEE Transactions on Automatic Control*, 42(5):648–658, May 1997.

[84] M.I. Skolnik. *Introduction to Radar Systems.* McGraw-Hill, New York, 1980.

[85] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I.J. Cox and G.T. Wilfon, editors, *Autonomous Robot Vehicles*, pages 167–193. Springer-Verlag, 1990.

[86] H.W. Sorenson, editor. *Kalman Filtering: Theory and Application.* IEEE Press, 1985.

[87] M.D. Squires, M. Whalen, G. Moody, C. Jacobus, and M. Taylor. Real-time landmark based optical vehicle self-location. In *Proceedings of SPIE - The International Society for Optical Engineering*, pages 187–197, Orlando, FL, April 1996.

[88] G.W. Stewart. *Introduction to Matrix Computation*. Academic Press, 1973.

[89] K.T. Sutherland and W.B. Thompson. Localizing in unstructured environments: Dealing with the errors. *IEEE Transactions on Robotics and Automation*, 10(6):740–754, December 1994.

[90] T. Tsumura. Survey of autonomous guided vehicles in japanese factory. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1329–1334, San Fransisco, CA, April 1986.

[91] J. Vandorpe, H. Xu, H. van Brussel, and E. Aertbelien. Positioning of the mobile robot lias using natural landmarks and a 2d rangefinder. In *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 257–264, Washington, DC, 1996.

[92] E. Yeh and D.J. Kriegman. Toward selecting and recognizing natural landmarks. In *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, pages 47–53, Pittsburgh, PA, August 1995.

[93] Y. Zhang and R. Webber. Dynamic world modeling for a mobile robot among moving objects. *Robotica*, 14(5):553–560, 1996.

[94] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, pages 119–152, 1994.