4

Efficient Simultaneous Localization and Mapping Algorithms using Submap Networks

by

SungJoon Kim

B.S. in Naval Architecture and Ocean Engineering (1995)
Seoul National University, Republic of Korea
M.S. in Naval Architecture and Ocean Engineering (1997)
Seoul National University, Republic of Korea

Submitted to the Department of Ocean Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Ocean Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2004

Author .....
Department of Ocean Engineering
May 20, 2004

Certified by .....
John J. Leonard
Associate Professor of Ocean Engineering
Thesis Supervisor

Accepted by .....
Michael S. Triantafyllou
Professor of Ocean Engineering
Chairman, Department Committee on Graduate Students

Efficient Simultaneous Localization and Mapping Algorithms Using Submap Networks

by

SungJoon Kim

Submitted to the Department of Ocean Engineering
on May 20, 2004, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Ocean Engineering

Abstract

Autonomous mapping of large-scale environments has been a critical challenge confronting researchers in mobile robotics. This thesis investigates two aspects of the large-scale simultaneous localization and mapping (SLAM) problem: (1) the behavior of the covariance matrix in the Kalman filter solution to the linear Gaussian SLAM problem, and (2) the development of new algorithms for efficient large-scale mapping.

The key issue motivating study of the linear Gaussian SLAM problem is to understand the behavior of the uncertainty estimates with time. In this thesis, we provide an analysis of the asymptotic behavior of the full covariance SLAM solution. We present a novel generalized closed-form solution to the single degree-of-freedom SLAM problem (known as the MonoRob problem). We examine the cross correlation behavior for the case of observed and non-observed features, and show that a feature must be repeatedly reobserved for it to become fully correlated with other features. Additionally, we provide a new "tight" lower bound for the map uncertainty for a certain class of the MonoRob problem.

The second part of the thesis develops new techniques for attacking the scaling problem in SLAM. The work builds on the Constant Time SLAM (CTS) method developed by Newman and Leonard, which is the first SLAM algorithm to achieve global convergence while maintaining consistent error bounds with an $\mathcal{O}(1)$ growth of complexity for the linear Gaussian SLAM problem. Our work makes four contributions: (1) We describe a new algorithm, termed CTS 2.0, that achieves better performance than CTS while maintaining constant-time performance. (2) We present an alternative submap network SLAM algorithm, termed Network Optimized SLAM (NOS), that transfers information across submaps in $O(n)$ time to achieve faster convergence than CTS while maintaining its desirable consistency properties. (3) we provide a theoretical and experimental analysis of CTS, CTS 2.0, and NOS and compare all three algorithms with the full covariance solution. (4) We perform an analysis of the error metrics for measuring the global uncertainty of a SLAM solution, yielding new insights into the behavior of this type of algorithm.

Thesis Supervisor: John J. Leonard
Title: Associate Professor of Ocean Engineering

Acknowledgments

I have gotten to the point where I can write this thesis only because of the enormous good luck I have had over the years with my supportive family, friends and insightful teachers. I would like to thank those who have helped me along the way. I know this list is still horribly incomplete, but I hope that all those who have touched my life realize that this page does not begin to capture my feelings or memories.

First, I owe a great deal to my advisor, John Leonard, who has been an unfailingly patient sounding board and teacher. Throughout many starts and stops, twists and turns, John has always tried to guide me towards a better understanding of the problems we have investigated. He is not only a great teacher and advisor, but a wonderful person that I feel privileged to know. My other thesis committee memebrs, Paul Newman, Nick Patrikalakis, and Nick Roy, provided me with valuable feedback on the structure and contents of this thesis. They were instrumental in helping with the process of finalizing and presenting this research. I wish to show my sincere appreciation to them, for their time, effort, and thoughtful input.

In the Marine Robotics group, I have known many great people, all of whom have impacted my work and life in some way or another. I would like to thank, Rick Rikoski, Matt Walter, Mike Bosse, Tom Fulton, Chris Cassidy, Mike Benjamin, Jay Dryer, Mike Slavik, John Fenwick, Ryan Eustice, Alex Bahr, and Gleb Chuvpilo. It was a great pleasure working together with them. Special thanks goes to Rick and Matt for being good friends and for giving me selfless help. Larry Greenspun deserves special credit for helping and editing my English writing.

My gratitude extends to my OE Korean friends, Changho Lee, Wonjoon Cho, Sungeun Kim, Yonghwan Kim, Kwanghee Ko, Jay Auh, Youngwoong Lee, Hyunjo Kim, Sunwoong Lee, and Kwanghyun Lee. They have been loyal lunch companions and friends. I also thank my former advisor, Keypyo Rhee, for his encouraging and helpful advice.

I cannot thank my family, and especially my mother and father, enough. I thank them, most of all for always wanting more for me. They taught me well and allowed

me to follow my passions. My thanks also go to my other family members for their care and support.

Finally, I am forever grateful to my wife, Sein, for all her support. For all the times I was unsure of myself, stressed from the workload, or doubtful of my plans, she provided a listening ear, a loving smile, and just the right words to improve my mood. Without her, getting my Ph.D. would not have been possible. I thank my wonderful sons, Chiwan and Jaewan, for giving me so much joy and for enriching my life in ways I could never describe.

Contents

1 Introduction 15

1.1 The Robot Navigation Problem 15

1.2 Challenges in Achieving SLAM 17

1.3 Thesis Scope 19

2 Simultaneous Localization and Mapping 21

2.1 Formulation of the Basic Kalman Filter SLAM Problem 22

2.1.1 Uncertain spatial relationships 22

2.1.2 Simultaneous Localization and Mapping 30

2.1.3 Basic SLAM procedure 36

2.2 The Large-Scale SLAM Problem 36

2.2.1 Single map approaches 36

2.2.2 Submap approaches 38

2.3 Summary 40

3 Properties of the SLAM Covariance Matrix 43

3.1 Introduction 43

3.2 Properties of the Covariance Matrix of the Linear Gaussian SLAM Solution 46

3.2.1 Definition of the covariance matrix 48

3.2.2 General properties 51

3.2.3 Properties specific to the Kalman filter and SLAM 54

3.3 Analytical Solution of a More General Case for the MonoRob SLAM Problem . . . 71
3.3.1 Closed form solutions . . . 73
3.3.2 Analysis of the closed form solution . . . 84
3.3.3 Case study . . . 89
3.4 Conclusion . . . 95

4 Efficient SLAM Algorithms Using Submap Networks 97
4.1 Constant Time SLAM (CTS) Algorithm . . . 97
4.1.1 Manipulating local maps in CTS . . . 99
4.1.2 Algorithm description . . . 106
4.2 Review of CTS . . . 112
4.2.1 Nonlinearity in determinants . . . 112
4.2.2 Information dissemination depth . . . 114
4.3 Efficient SLAM algorithms . . . 116
4.3.1 Graphical description . . . 117
4.3.2 Improvement of the nonlinearity, Constant Time SLAM (CTS) 2.0 . . . 120
4.3.3 Improvement of the information dissemination depth, Network Optimized SLAM (NOS) . . . 121
4.4 Global Uncertainty of the Whole Map . . . 127
4.5 Experimental Results . . . 135
4.5.1 Experiment configuration . . . 135
4.5.2 Results and analysis . . . 142
4.6 Conclusion . . . 157

5 Conclusion 159
5.1 Summary of Contributions . . . 159
5.2 Future Work . . . 161
5.2.1 The nonlinear SLAM problem . . . 161
5.2.2 Implementation with undersea sonar data . . . 161

5.2.3 The multi-vehicle case 162
5.2.4 Integration with autonomous exploration 162

A Solution Procedure for the One DOF SLAM Problem **165**

A.1 One Observed Feature and Two Unobserved Features 165
A.1.1 $\dot{\mathbf{U}}$ and $\dot{\mathbf{V}}$ 166
A.1.2 Solutions of $\mathbf{U}$ and $\mathbf{V}$ 166
A.1.3 Summary in matrix form 173
A.1.4 Inverse of $\mathbf{V}(t)$ 175
A.1.5 Elements of the covariance matrix 176
A.2 Two Observed Features and One Unobserved Feature 191
A.2.1 $\dot{\mathbf{U}}$ and $\dot{\mathbf{V}}$ 192
A.2.2 Solution of $\mathbf{U}$ and $\mathbf{V}$ 192
A.2.3 Inverse of $\mathbf{V}(t)$ 207
A.2.4 Determinant of $\mathbf{V}(t)$ 207
A.2.5 Structure of $\mathbf{V}(t)$ inverse 209
A.2.6 Summary of the elements of the $\mathbf{V}(t)$ inverse 210
A.2.7 Elements of the covariance matrix 211

List of Figures

2-1 Illustration of the compounding process. 23

2-2 Illustration of the inversion process. 25

2-3 Illustration of two consecutive compounding operations. 26

2-4 Illustration of the composite process. 29

2-5 Feature initialization and size varying state. 32

2-6 Flow chart for a single cycle of Kalman filter SLAM. 35

2-7 Map configurations of various large-scale SLAM solutions. 41

3-1 Venn diagram of non-negative definite matrices. 52

3-2 Terminal determinants when only one feature is observed. 61

3-3 Terminal determinants when the feature observed is switched once. . 62

3-4 Terminal determinants when two features are simultaneously observed. 65

3-5 Cross correlations when two features are simultaneously observed. . . 66

3-6 Compounding process. 67

3-7 Root shifting. 68

3-8 Level-wise update and dependencies in the Kalman filter update. . . . 70

3-9 Terminal values of two observed feature columns. 87

3-10 Lower bound of the feature uncertainty. 89

3-11 Determinants of the feature blocks when only one feature is observed. 91

3-12 Cross correlation behavior when only one feature is observed. 91

3-13 Determinants of the feature blocks when the feature observed is switched once. 92

3-14 Cross correlation behavior when the feature observed is switched once. 92
3-15 Determinants of the feature blocks when the feature observed is switched every two seconds. 93
3-16 Cross correlation behavior when the feature observed is switched every two seconds. 93
3-17 Determinants of the feature blocks when two features are simultaneously observed. 94
3-18 Cross correlation behavior when two features are simultaneously observed. 94

4-1 Origin settings of the submaps. 99
4-2 Global uncertainty of a feature location through compounding. 104
4-3 Global uncertainty of feature locations through compounding. 106
4-4 Flowchart of the CTS algorithm. 107
4-5 No shared information between two submaps. 108
4-6 Root feature selection of the CTS algorithm. 111
4-7 Nonlinearity in compounding. 113
4-8 Nonlinearity in root feature selection. 114
4-9 Depth of the information dissemination of the CTS algorithm. 116
4-10 Submap networks and the map location estimation of the CTS algorithm. 118
4-11 Root feature selection in the CTS 2.0 algorithm. 121
4-12 Flowchart of the CTS 2.0 algorithm. 122
4-13 Submap networks and the map location estimation of the NOS algorithm. 124
4-14 Flowchart of the NOS algorithm. 125
4-15 Depth first search (DFS) in the NOS algorithm. 126
4-16 The uncertainties of the global map through the CTS based algorithms. 129
4-17 The effect of the best root feature. 134

4-18 Feature parameterization of a hurdle. 135
4-19 Feature locations on the tennis courts. 136
4-20 Dead reckoned robot trajectory. 136
4-21 Dynamic model of the Johnson Center experiment. 138
4-22 Measurement model of the Johnson Center experiment. 138
4-23 Johnson Center experiment. 140
4-24 Johnson Center experiment. 140
4-25 Johnson Center experiment. 141
4-26 Johnson Center experiment. 141
4-27 Global uncertainty profile of the root feature of the submap 3 with time. 144
4-28 Global uncertainty profile of the root feature of the submap 3 with time. 144
4-29 Global uncertainty profile of the root feature of the submap 5 with time. 145
4-30 Global uncertainty profile of the root feature of the submap 5 with time. 145
4-31 Global uncertainty profile of the root feature of the submap 6 with time. 146
4-32 Global uncertainty profile of the root feature of the submap 6 with time. 146
4-33 Global uncertainties of the root features at time step 1000. 147
4-34 Global uncertainties of the root features at time step 2000. 147
4-35 Global uncertainties of the root features at time step 3000. 148
4-36 Global uncertainties of the root features at time step 4000. 148
4-37 Global uncertainties of the root features at time step 6000. 149
4-38 Global uncertainties of the root features at time step 7000. 149
4-39 Global uncertainties of the root features at time step 10000. 150
4-40 Global uncertainties of the root features at time step 20000. 150
4-41 Global uncertainties of the root features at time step 30000. 151
4-42 Global uncertainties of the root features at time step 39000. 151
4-43 The structure of the map tree by CTS at timestep 1000. 152
4-44 The structure of the map tree by NOS at timestep 1000. 152
4-45 The structure of the map tree by CTS at timestep 2000. 152
4-46 The structure of the map tree by NOS at timestep 2000. 152
4-47 The structure of the map tree by CTS at timestep 3000. 152

4-48 The structure of the map tree by NOS at timestep 3000. 152
4-49 The structure of the map tree by CTS at timestep 4000. 153
4-50 The structure of the map tree by NOS at timestep 4000. 153
4-51 The structure of the map tree by CTS at timestep 5000. 153
4-52 The structure of the map tree by NOS at timestep 5000. 153
4-53 The structure of the map tree by CTS at timestep 6000. 153
4-54 The structure of the map tree by NOS at timestep 6000. 153
4-55 The structure of the map tree by CTS at timestep 10000. 154
4-56 The structure of the map tree by NOS at timestep 10000. 154
4-57 The structure of the map tree by CTS at timestep 20000. 154
4-58 The structure of the map tree by NOS at timestep 20000. 154
4-59 The structure of the map tree by CTS at timestep 39000. 154
4-60 The structure of the map tree by NOS at timestep 39000. 154
4-61 The global location estimate of submap 1. 155
4-62 The global location estimate of submap 2. 155
4-63 The global location estimate of submap 3. 155
4-64 The global location estimate of submap 4. 155
4-65 The global location estimate of submap 5. 155
4-66 The global location estimate of submap 6. 155
4-67 The global location estimate of submap 7. 156
4-68 The global location estimate of submap 8. 156
4-69 The global location estimate of submap 9. 156
4-70 The global location estimate of submap 10. 156
4-71 The global location estimate of submap 11. 156

Chapter 1

Introduction

1.1 The Robot Navigation Problem

People desire robots that will effectively and efficiently perform human, labor-intensive tasks. In many areas, this goal has been achieved. Robotic vacuum cleaners shine supermarket and train station floors. Automatic tooling systems assist in automobile production. Tour-guide robots lead visitors through museums, and autopilot systems direct aircraft around the world. However, the full potential of robots has not yet been reached. The ability of robots to assist individuals with physical differences remains underdeveloped. Similarly, the usefulness of robots in fields from surgery to formal education has not progressed beyond its nascent stage. Another potentially rich arena for the utilization of robots is in fully autonomous navigation in unknown environments.

When a person visits an unfamiliar environment such as an airport in a foreign country, s/he employs a variety of cues, skills, and mechanisms to successfully navigate from one place (point of disembarkation to baggage claim to taxi stand) to another. Though much of this process is understood, many questions remain about how people develop spatial memory and process orientation and directional information. One of the basic research and application goals in robotics is to endow robots with spatial memory as well as orientation and directional processing abilities that parallel those of human beings. We seek a positive answer to the question, "Can a robot build maps

of unknown areas and use these self-generated maps to execute desired paths?"

The difficulty in achieving fully autonomous navigation by robots is illustrated through a comparison to corresponding human capabilities at various ages. Infants do not navigate independently and manifest a limited knowledge of location and desired direction. In a similar way, many primitive robots, such as remote controlled vehicles and automatic tooling systems, move only along an established track, according to pre-programmed path plans, or through human control. Pre-teens evince a more highly developed internal navigation system. As long as an effective map is provided, adolescents can navigate independently. Similarly, if either an accurate map or an external localization system is available, an advanced mobile robot can execute the same class of navigational and exhibit task such as maze solving and museum tour guiding. Finally, adults have acquired spatial and directional processing and memory skills that allow them to travel on their own from place to place even within an unfamiliar environment. No current robot can imitate this level of autonomous navigation. Overcoming this hurdle is a primary goal of research in robotics and artificial intelligence.

The key technical challenge to autonomous robot navigation in an unknown environment is coping with uncertainty in all its forms, including noise, ambiguity, biases, and modeling errors. Brooks summarized the problem and suggested a probabilistic approach to this situation.

> *Mobile robots sense their environment and receive error laden readings. They try to move a certain distance and direction, only to do so approximately. Rather than try to engineer these problems away it may be possible, and may be necessary, to develop map making and navigation algorithms which explicitly represent these uncertainties, but still provide robust performance.* [12]

Successful robot navigation also requires a map of the environment. However, to build a map, the robot must move among multiple positions to localize itself. The navigation error incurred during the vehicle's motion combines with the uncertainty in

the robot's sensor to create a challenging interpretation problem. This is the problem of Simultaneous Localization and Mapping (SLAM) — to concurrently generate a map of the environment and localize the robot using the map being generated.

1.2 Challenges in Achieving SLAM

Many ideas for attacking the SLAM problem have been suggested and implemented over the past several decades, and significant progress has been achieved. However, a fully-realized SLAM solution remains elusive. One of the remaining challenges confronting the robotics community is the case of a large-scale environment.

As the size of the operating environment increases, the computational demands of SLAM grow significantly, making real-time computation problematic. The primary cause of the growth in computational burden is the need to represent correlations between the error estimates of the spatial poses of the vehicle and all the features in the map. The optimal algorithm for retaining the maximum number of correlations encounters an $\mathcal{O}(n^2)$ computational burden [62], where n represents the number of features.

These correlations are kept in the covariance matrix during the solving of a SLAM problem. Previous researchers have examined the behavior of the covariance matrix [34, 26], providing new insights into the solution behavior for a certain specialized class of SLAM problem. In this thesis, we add to this body of work by considering a more general formulation of the problem, providing new insights into the problem structure, and to applying these insights to evaluate new approaches for solving the large-scale SLAM dilemma.

Achieving large-scale SLAM is a complex process. An essential element of successful large-scale SLAM involves balancing the need for accurate estimation with the need for faster computation. If all the correlations between features and the robot are maintained, the resulting pose estimate provides a more accurate result but with the cost of prolonged computation. Conversely, if some correlations are ignored, the estimation can be executed faster, but with a less accurate or even an inconsistent

result. This thesis suggests a solution that satisfies the need for both accurate and consistent estimates and real-time computation by using a submap network.

In addition to balancing accuracy and consistency with efficiency, successful, real-time SLAM requires accurate data association, robust feature detection, and reliable loop closing technique. Even though these requirements are usually coupled, workable solutions for all of them have been developed by other researchers. This allows us, for the purpose of this thesis, to assume that we approach the matters of balancing consistency and efficiency armed with the best possible algorithms.

In order to accomplish large-scale SLAM, we divide the global region into multiple sub-regions, and conquer each local region with a conventional solution such as stochastic mapping [76] and combine the local solutions into a single global solution.

A graphical representation for the submap network will be adopted in this thesis because the local regions and spatial relations between regions have one-to-one correspondence to nodes and edges defining a graph. This correspondence enables us to exploit an abundance of previously developed algorithms in graph theory. For example, finding a best origin for a local region can be transformed into a process of finding a shortest path problem in an undirected graph with negative valued edges. In fact, solving best local origins edges plays a crucial role in evaluating the global estimates from local solutions and their relations. Previous work that has successfully exploited a graphical model for SLAM includes the work of Kuipers [49], Gutmann and Konolige [39], and Bosse *et al.* [10].

As a measure of the edge value, the determinant of the covariance matrix is a natural choice [10]. The determinant quantifies how strongly two local SLAM solutions are related to each other. Also, the behavior of the covariance matrix tests the estimation's consistency and demonstrates the minimum achievable limit of the uncertainty of the estimation.

1.3 Thesis Scope

This thesis investigates two aspects of the SLAM problem: 1) the examination of the properties of the covariance matrix for the Kalman filter solution to the linear Gaussian SLAM problem, and 2) the development of new and more efficient SLAM algorithms using a submap network representation. The new algorithms rely heavily on the insights into the solution structure provided in the first part of the work.

Of course, there are many important issues in SLAM research besides the scaling problem. These include data association, map representation, feature detection and modeling, information transfer management for multiple robot mapping, and autonomous exploration. These problems are clearly very difficult, and are often coupled to the scaling problem (e.g., data association for loop closing). For this thesis, however, we consider all of these issues as "out of scope". Essentially, we assume that good data association information is available, and that the environment are compatible with a feature-based map representation of "point" objects. We provide some thoughts on incorporating some of these issues with our new results in a discussion of future work at the conclusion of the thesis.

We now summarize the structure of this thesis document. In Chapter 2, we review the literature on large-scale SLAM. In Chapter 3, we examine the properties of the covariance matrix and solve two new cases of the "MonoRob" single degree-of-freedom SLAM problem analytically. The solutions are used to verify the properties of the covariance matrix. Extensive hand computations for the new closed-form solution are provided in Appendix A. In Chapter 4, we analyze the Constant Time SLAM (CTS) method in detail and describe two new SLAM algorithms that build on this approach to achieve improved performance. Finally, a conclusion and areas for further investigation are provided in Chapter 5.

Chapter 2

Simultaneous Localization and Mapping

As stated in the previous chapter, the goal of Simultaneous Localization and Mapping (SLAM) is to build a map of an unknown environment, while using that map to navigate concurrently. In order to achieve this goal, we employ the feature-based SLAM solution using Kalman filter estimation. This approach is introduced in the first part of this chapter. One of the key issues in feature-based SLAM is the map-scaling problem. The computational complexity of the standard full covariance SLAM solution is $\mathcal{O}(n^2)$, where n is the number of features in the environment being mapped. This complexity results from the cross correlations of the covariance matrix maintained during Kalman filter estimation. The size of the covariance matrix grows as the robot detects a new feature and integrates it into the estimated state vector. The latter part of this chapter provides brief reviews of the previous work to resolve this map-scaling issue in SLAM problems. As described in Chapter 1, we consider all of the other issues such as data association, map representation, and feature detection and modeling as "out of scope".

2.1 Formulation of the Basic Kalman Filter SLAM Problem

The Kalman filter approach to the SLAM problem was first published by Smith, Self, and Cheeseman [75] and Moutarlier and Chatila [62]. At the heart of the problem is the manipulation of uncertain spatial relationships. Following, Smith, Self, and Cheeseman, we first review the techniques of compounding and inversion of uncertain spatial relationships, which are essential for the material that follows.

2.1.1 Uncertain spatial relationships

The random process state vector in the SLAM problem is composed of the spatial locations of the robot that surveys the environment and the parameterized locations of features of the environment. The robot moves around the region through a given dynamic model, measures locations of features, and build a map of the features. The map is simultaneously used to update the robot's poses.

Every measurement is executed by the robot and, therefore, the sensor data are local poses with respect to the sensor usually attached on the robot. In order to keep and process those local data in one coordinate frame, we need operations that transform these local values to the global values. Since the sensor data includes noise, the transformation to the global coordinate frame should involve the uncertainty transformation. The followings are several operations used in SLAM solutions.

- compounding

  Compounding is the most important transformation. Whenever a feature location is measured with respect to the robot (say $\mathbf{x}_{jk}$), that measurement is transformed to the global coordinate via the robot location(say $\mathbf{x}_{ij}$). That is, the compounding process on two vectors $\mathbf{x}_{ij}$ and $\mathbf{x}_{jk}$ returns $\mathbf{x}_{ik}$ and the corresponding covariance $\mathbf{P}_{ik,ik}$. [1] $\mathbf{x}_{ij} \oplus \mathbf{x}_{jk}$ denotes the binary operation, *com-*

[1] The subscripts in this section differ from the way the covariance matrix in the other sections is expressed in order to explicitly denote the relative fiducial point. $\mathbf{P}_{ij,ij}$ represents that the variance associated with $\mathbf{x}_{ij}$, which is the location vector of point j relative to the point i.

pounding. For example, the two-dimensional compounding process for point vectors is defined as

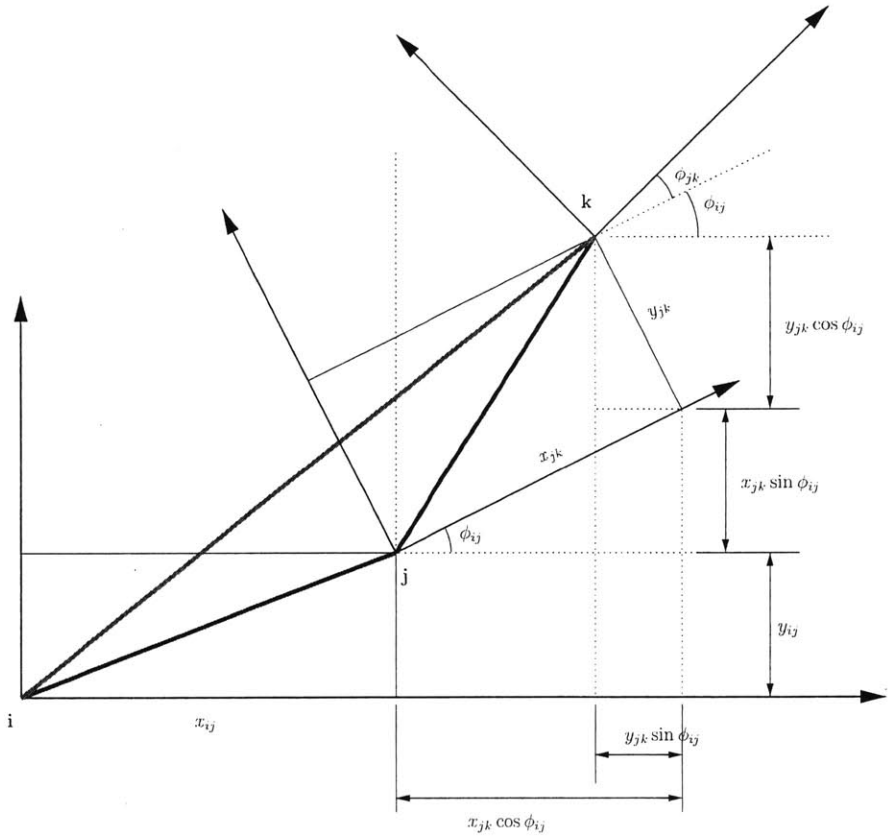

Figure 2-1: Illustration of the compounding process.

$$\mathbf{x}_{ik} \equiv \mathbf{x}_{ij} \oplus \mathbf{x}_{jk} = \begin{bmatrix} x_{jk}\cos\phi_{ij} - y_{jk}\sin\phi_{ij} + x_{ij} \\ x_{jk}\sin\phi_{ij} + y_{jk}\cos\phi_{ij} + y_{ij} \\ \phi_{ij} + \phi_{jk} \end{bmatrix} \tag{2.1}$$

Figure 2-1 shows these relationships. The resultant two moments can be calculated by using an appropriate linear transform relation with a linearization.

$$\hat{\mathbf{x}}_{ik} \approx \hat{\mathbf{x}}_{ij} \oplus \hat{\mathbf{x}}_{jk} \tag{2.2}$$

$$\mathbf{P}_{ik,ik} \approx \mathbf{J}_{\oplus} \begin{bmatrix} \mathbf{P}_{ij,ij} & \mathbf{P}_{ij,jk} \\ \mathbf{P}_{jk,ij} & \mathbf{P}_{jk,jk} \end{bmatrix} \mathbf{J}_{\oplus}^{T} \tag{2.3}$$

In the above, linearization approximates the nonlinear operations and the Jacobian of the compounding operation, $\mathbf{J}_{\oplus}$ is given by

$$\mathbf{J}_{\oplus} \equiv \frac{\partial(\mathbf{x}_{ij} \oplus \mathbf{x}_{jk})}{\partial(\mathbf{x}_{ij}, \mathbf{x}_{jk})} = \frac{\partial \mathbf{x}_{ik}}{\partial(\mathbf{x}_{ij}, \mathbf{x}_{jk})} = \begin{bmatrix} 1 & 0 & -(y_{ik} - y_{ij}) & \cos\phi_{ij} & -\sin\phi_{ij} & 0 \\ 0 & 1 & (x_{ik} - x_{ij}) & \sin\phi_{ij} & \cos\phi_{ij} & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

- inversion

The *inversion* operation is the simplest transformation that returns the inversion spatial pose and the corresponding uncertainty. It is a unitary operation on a vector and is denoted by $\ominus$. The following example shows an inversion operation on a two-dimensional point vector, $\mathbf{x}_{ij}$.

$$\mathbf{x}_{ji} \equiv \ominus \mathbf{x}_{ij} = \begin{bmatrix} -x_{ij}\cos\phi_{ij} - y_{ij}\sin\phi_{ij} \\ x_{ij}\sin\phi_{ij} - y_{ij}\cos\phi_{ij} \\ -\phi_{ij} \end{bmatrix} \tag{2.5}$$

The resulting two moments are

$$\hat{\mathbf{x}}_{ji} \approx \ominus \hat{\mathbf{x}}_{ij} \tag{2.6}$$

$$\mathbf{P}_{ji,ji} \approx \mathbf{J}_{\ominus} \mathbf{P}_{ij,ij} \mathbf{J}_{\ominus}^{T} \tag{2.7}$$

where the Jacobian for the inversion operation, $\mathbf{J}_{\ominus}$ is written in the form

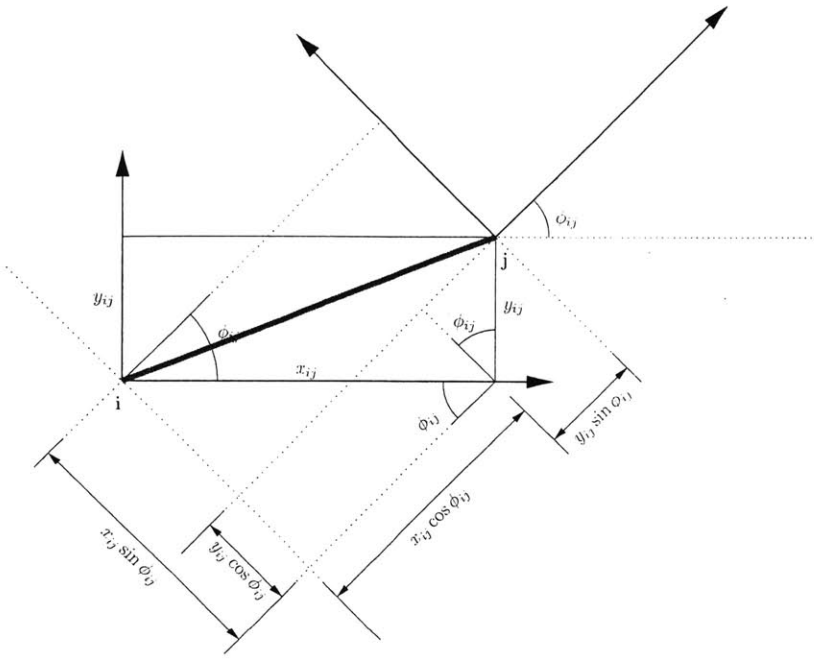

Figure 2-2: Illustration of the inversion process.

$$\mathbf{J}_{\ominus} \equiv \frac{\partial \mathbf{x}_{ji}}{\partial \mathbf{x}_{ij}} = \begin{bmatrix} -\cos\phi_{ij} & -\sin\phi_{ij} & y_{ji} \\ \sin\phi_{ij} & -\cos\phi_{ij} & -x_{ji} \\ 0 & 0 & -1 \end{bmatrix} \tag{2.8}$$

- composite relationship

The *composite* relationship is merely combinations of the above two transformations. First, two consecutive inversion operations return the identical vector and uncertainty. Secondly, two consecutive compounding operations return the following result. For the covariance, transform relations (2.10) are employed.

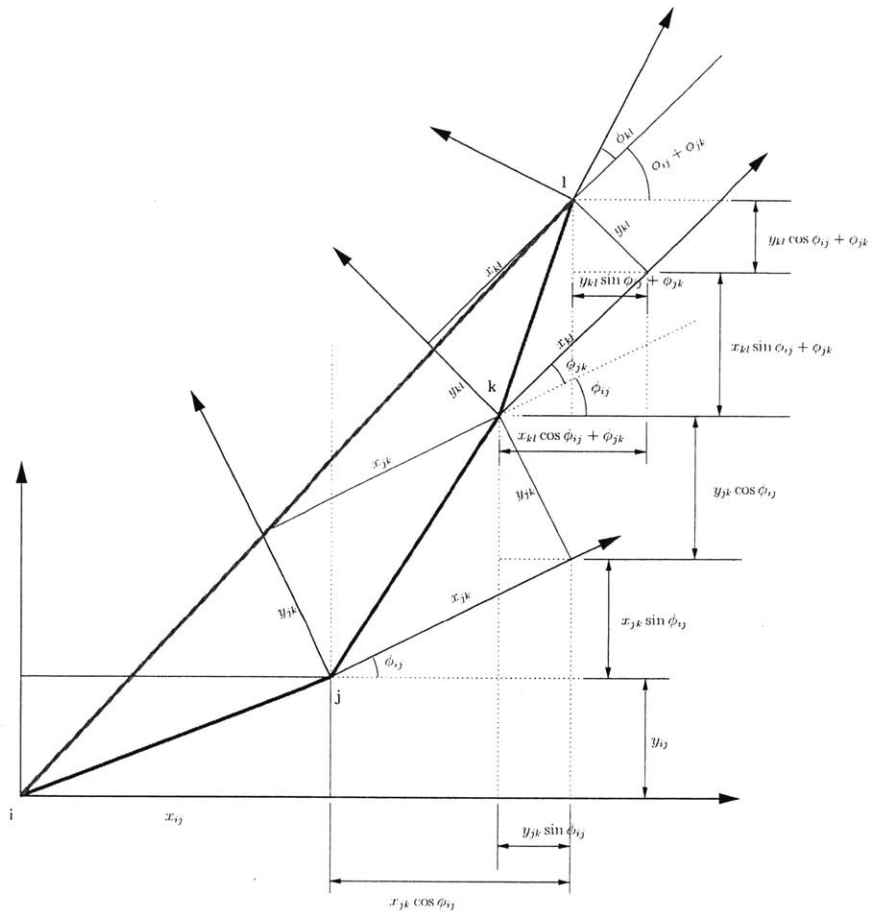

Figure 2-3: Illustration of two consecutive compounding operations.

$$
\begin{aligned}
\mathbf{x}_{il} &\equiv \mathbf{x}_{ik} \oplus \mathbf{x}_{jk} \\
&= (\mathbf{x}_{ij} \oplus \mathbf{x}_{jk}) \oplus \mathbf{x}_{jk} \\
&= \mathbf{x}_{ij} \oplus (\mathbf{x}_{jk} \oplus \mathbf{x}_{jk}) \\
&= \mathbf{x}_{ij} \oplus \mathbf{x}_{jl} \\
&= \begin{bmatrix} x_{ij} + \{x_{jk}\cos\phi_{ij} - y_{jk}\sin\phi_{ij}\} + \{x_{kl}\cos(\phi_{ij}+\phi_{jk}) - y_{kl}\sin(\phi_{ij}+\phi_{jk})\} \\ y_{ij} + \{x_{jk}\sin\phi_{ij} + y_{jk}\cos\phi_{ij}\} + \{x_{kl}\sin(\phi_{ij}+\phi_{jk}) + y_{kl}\cos(\phi_{ij}+\phi_{jk})\} \\ \phi_{ij} + \phi_{jk} + \phi_{kl} \end{bmatrix} \\
&\quad \text{(for 2D point vectors)}
\end{aligned}
\tag{2.9}
$$

$$\mathbf{P}_{il,il} \approx \mathbf{J}_{\oplus\oplus} \begin{bmatrix} \mathbf{P}_{ij,ij} & \mathbf{P}_{ij,jk} & \mathbf{P}_{ij,kl} \\ \mathbf{P}_{jk,ij} & \mathbf{P}_{jk,jk} & \mathbf{P}_{jk,kl} \\ \mathbf{P}_{kl,ij} & \mathbf{P}_{kl,jk} & \mathbf{P}_{kl,kl} \end{bmatrix} \mathbf{J}_{\oplus\oplus}^T \tag{2.10}$$

where,

$$\begin{aligned} \mathbf{J}_{\oplus\oplus} &\equiv \frac{\partial(\mathbf{x}_{ij} \oplus \mathbf{x}_{jk} \oplus \mathbf{x}_{kl})}{\partial(\mathbf{x}_{ij}, \mathbf{x}_{jk}, \mathbf{x}_{kl})} = \frac{\partial \mathbf{x}_{il}}{\partial(\mathbf{x}_{ij}, \mathbf{x}_{jk}, \mathbf{x}_{kl})} \\ &= \begin{bmatrix} \mathbf{J}_{1\oplus\oplus} & \mathbf{J}_{2\oplus\oplus} & \mathbf{J}_{3\oplus\oplus} \end{bmatrix} \end{aligned} \tag{2.11}$$

$$\mathbf{J}_{1\oplus\oplus} = \begin{bmatrix} 1 & 0 & \{-x_{jk}\sin\phi_{ij} - y_{jk}\cos\phi_{ij}\} + \{-x_{kl}\sin(\phi_{ij}+\phi_{jk}) - y_{kl}\cos(\phi_{ij}+\phi_{jk})\} \\ 0 & 1 & \{\ x_{jk}\cos\phi_{ij} - y_{jk}\sin\phi_{ij}\} + \{\ x_{kl}\cos(\phi_{ij}+\phi_{jk}) - y_{kl}\sin(\phi_{ij}+\phi_{jk})\} \\ 0 & 0 & 1 \end{bmatrix} \tag{2.12}$$

$$\mathbf{J}_{2\oplus\oplus} = \begin{bmatrix} \cos\phi_{ij} & -\sin\phi_{ij} & -x_{kl}\sin(\phi_{ij}+\phi_{jk}) - y_{kl}\cos(\phi_{ij}+\phi_{jk}) \\ \sin\phi_{ij} & \cos\phi_{ij} & x_{kl}\cos(\phi_{ij}+\phi_{jk}) - y_{kl}\sin(\phi_{ij}+\phi_{jk}) \\ 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

$$\mathbf{J}_{3\oplus\oplus} = \begin{bmatrix} \cos(\phi_{ij}+\phi_{jk}) & -\sin(\phi_{ij}+\phi_{jk}) & -x_{kl}\sin(\phi_{ij}+\phi_{jk}) - y_{kl}\cos(\phi_{ij}+\phi_{jk}) \\ \sin(\phi_{ij}+\phi_{jk}) & \cos(\phi_{ij}+\phi_{jk}) & x_{kl}\cos(\phi_{ij}+\phi_{jk}) - y_{kl}\sin(\phi_{ij}+\phi_{jk}) \\ 0 & 0 & 1 \end{bmatrix} \tag{2.14}$$

Finally, inversion followed by composition gives us the following spatial information.

$$
\begin{aligned}
\mathbf{x}_{jk} &= \mathbf{x}_{ji} \oplus \mathbf{x}_{ik} \\
&= (\ominus \mathbf{x}_{ij}) \oplus \mathbf{x}_{ik} \\
&= \begin{bmatrix} -x_{ij}\cos\phi_{ij} - y_{ij}\sin\phi_{ij} + x_{ik}\cos\phi_{ik} - y_{ik}\sin\phi_{ik} \\ x_{ij}\sin\phi_{ij} - y_{ij}\cos\phi_{ij} + x_{ik}\sin\phi_{ik} + y_{ik}\cos\phi_{ik} \\ -\phi_{ij} + \phi_{ik} \end{bmatrix}
\end{aligned} \tag{2.15}
$$

$$
\begin{aligned}
\mathbf{P}_{jk,jk} &\approx \begin{bmatrix} \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ji,ji} & \mathbf{P}_{ji,ik} \\ \mathbf{P}_{ji,ik}^T & \mathbf{P}_{ik,ik} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{1\oplus}^T \\ \mathbf{J}_{2\oplus}^T \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\ominus} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ij,ij} & \mathbf{P}_{ij,ik} \\ \mathbf{P}_{ij,ik}^T & \mathbf{P}_{ik,ik} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\ominus}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{1\oplus}^T \\ \mathbf{J}_{2\oplus}^T \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{J}_{1\oplus}\mathbf{J}_{\ominus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ij,ij} & \mathbf{P}_{ij,ik} \\ \mathbf{P}_{ij,ik}^T & \mathbf{P}_{ik,ik} \end{bmatrix} \begin{bmatrix} \mathbf{J}_{\ominus}^T\mathbf{J}_{1\oplus}^T \\ \mathbf{J}_{2\oplus}^T \end{bmatrix}
\end{aligned} \tag{2.16}
$$

In the above, $\mathbf{J}_{1\oplus}$, $\mathbf{J}_{2\oplus}$, and $\mathbf{J}_{\ominus}$ are Jacobians that were used for the compounding and inversion operations

Two consecutive compoundings are frequently used to evaluate the uncertainties of submaps with respect to the global origin. Additionally, the inversion and compounding combinations are necessary to execute the *root shifting* operation (changing the base reference of a local map.) Both processes will be explained in chapter 4.

By the described operations, we can calculate transformed information of spatial quantities. Those transforms cover not only the first moment, mean value, but also the second central moment, error covariances. In SLAM problems, the error covariances are used to measure the certainties of the estimates. The smaller a covariance value is, the closer the estimated mean is to the true value. To visualize the uncertainties of spatial locations, ellipses (for 2 dimensional locations) and ellipsoids (for

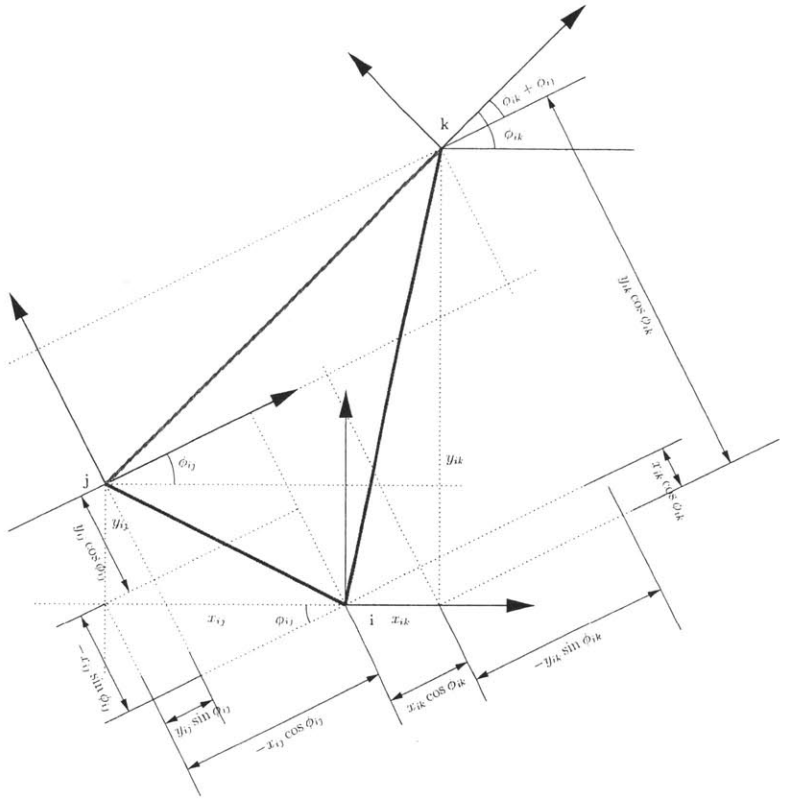

Figure 2-4: Illustration of the composite process (inversion and compounding).

3 dimensional locations) are often used. If the mean and the error covariance matrix of a random vector are $\mathbf{m}$ and $\mathbf{P}$, then the error ellipse or ellipsoid is defined as

$$(\eta - \mathbf{m})\,\mathbf{P}^{-1}\,(\eta - \mathbf{m})^T = C \tag{2.17}$$

and C determines the confidence level of the estimates. The center of the ellipse is located at the mean values. The variances, which are the diagonal elements of the covariance matrix, determine the magnitudes of the major and minor axes. If the cross correlation is zero, the major axis and the minor axis are parallel to the Cartesian axes. If the cross correlation is not zero, the angular orientation of the ellipse is tilted. The ellipse becomes a circle if the magnitudes of the variances are the same. If the covariance matrix becomes singular, which implies that the rank of the covariance matrix becomes 1 for a 2 dimensional case, one axis shrinks down to zero and the ellipse become a line segment. This means that the specific value is known without

uncertainty. The properties for ellipses are generalized to n-dimensional ellipsoids.

2.1.2 Simultaneous Localization and Mapping

The goal of the SLAM problem is to build a map of an unknown environment, while using that map to navigate. The robot moves around the surveying region according to the dynamic model and observes its surroundings by one or more sensors according to the provided observation model. The observation is used to create a map of the environment, which, in turn, is used to localize the robot itself.

Basically, SLAM is a special kind of state estimation problem for a dynamic system. Thus, many estimation methods can be applied to solve the main part of a SLAM problem. This thesis uses the feature based extended Kalman filter as the estimation tool. In feature based SLAM, a feature is parameterized to be added to the map and is estimated by the robot. For example, a point can be initialized by two position values in a 2 dimensional environment and a line segment can be expressed by four parameters such as x and y positions of two end points. In a feature based SLAM, the objective of estimation is a set of location values of the surveying robot and parameterized feature locations. The state vector consists of these location values and, as a matter of convenience, the locations of the robot occupy the first elements of the state vector. The equation (2.18) is a typical form of the state vector of a feature based SLAM.

$$\mathbf{x}[k] = \begin{bmatrix} \mathbf{x}_v[k] \\ \mathbf{x}_{f_1}[k] \\ \mathbf{x}_{f_2}[k] \\ \vdots \\ \mathbf{x}_{f_n}[k] \end{bmatrix} \tag{2.18}$$

The linear least-square estimate of the state vector is denoted as $\hat{\mathbf{x}}[k]$ and its difference from the true states becomes the error of the estimates, $\tilde{\mathbf{x}}[k]$.

$$\hat{\mathbf{x}}[k] = \mathbf{x}[k] - \tilde{\mathbf{x}}[k] \tag{2.19}$$

The error covariance of the estimates is defined as $\mathbf{P}[k] = \mathbf{E}\left[\tilde{\mathbf{x}}[k]\tilde{\mathbf{x}}^T[k]\right]$ and is written in the form

$$\mathbf{P}[k] = \begin{bmatrix} \mathbf{P}_{vv}[k] & \mathbf{P}_{vf_1}[k] & \mathbf{P}_{vf_2}[k] & \cdots & \mathbf{P}_{vf_n}[k] \\ \mathbf{P}_{f_1v}[k] & \mathbf{P}_{f_1f_1}[k] & \mathbf{P}_{f_1f_2}[k] & \cdots & \mathbf{P}_{f_1f_n}[k] \\ \mathbf{P}_{f_2v}[k] & \mathbf{P}_{f_2f_1}[k] & \mathbf{P}_{f_2f_2}[k] & \cdots & \mathbf{P}_{f_2f_n}[k] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{f_nv}[k] & \mathbf{P}_{f_nf_1}[k] & \mathbf{P}_{f_nf_2}[k] & \cdots & \mathbf{P}_{f_nf_n}[k] \end{bmatrix} \tag{2.20}$$

In the equations above, k is a time index. When the extended Kalman filter is applied for state estimation, the notation $[k|k]$ and $[k|k-1]$ is frequently used. By convention, $\hat{\mathbf{x}}[k|k]$ represents the state estimate at time k with the information acquired up to k. As described, the information up to time k includes all the control inputs and measurements. $\hat{\mathbf{x}}[k|k-1]$ expresses the state estimate at time k with the information given Up to $k-1$. One should note that the latter case does not include the current observation. Consequently, $\hat{\mathbf{x}}[k|k-1]$ is the estimate by the prediction step and $\hat{\mathbf{x}}[k|k]$ by the update step of the Kalman filter.

When an extended Kalman filter is employed for a SLAM problem, the prediction step is usually caused by the robot movement. As is usual in SLAM, we assume that all features in the surveying region are static. If we deal with an environment containing moving features, their dynamic models and prediction steps should be considered. The update step occurs whenever the robot observes features that are already in the state vector. The measured information improves the state estimates through the measurement model of the Kalman filter.

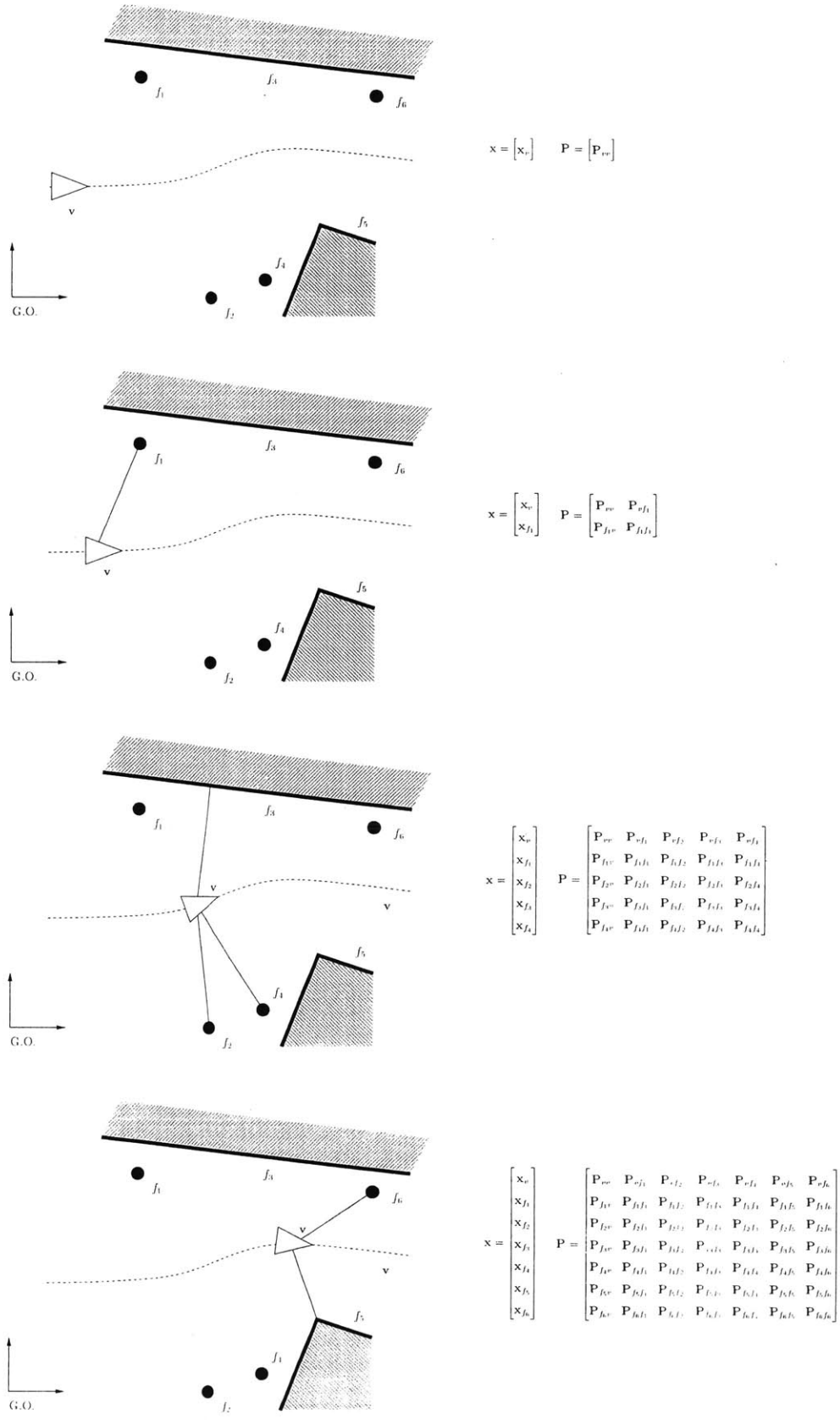

Figure 2-5: Feature initialization and size varying state.

At the beginning of the survey, the state vector includes only the states of the robot. New state elements are added to the state vector when new features are detected. The measured location parameters of the new feature are transformed to be globally referenced through the basic compounding process and are added to the state vector. This process occurs as follows [75]:

$$\begin{aligned} \hat{\mathbf{x}}[k] &\leftarrow \begin{bmatrix} \hat{\mathbf{x}}[k] \\ \hat{\mathbf{x}}_{f_{new}} \end{bmatrix} \\ \mathbf{P}[k] &\leftarrow \begin{bmatrix} \mathbf{P}[k] & \mathbf{P}[k]\mathbf{G}_v^T \\ \mathbf{G}_v\mathbf{P}[k] & \mathbf{G}_v\mathbf{P}[k]\mathbf{G}_v^T + \mathbf{G}_z\mathbf{R}\mathbf{G}_z^T \end{bmatrix} \end{aligned} \tag{2.21}$$

where, $\mathbf{G}_v$ and $\mathbf{G}_z$ are the Jacobians of the feature initialization model with respect to the robot state $\mathbf{x}_v[k]$ and to the new measurement $\mathbf{z}_{new}$ associated with the new feature $\mathbf{x}_{f_{new}}$. The covariance of the state of the new feature is given *a priori* by $\mathbf{R}$. For example, if a new point feature is detected by the distance and direction with respect to the robot in a 2 dimensional environment, the feature initialization model and its corresponding Jacobians become

$$\begin{aligned} \mathbf{z}_{new} &= \begin{bmatrix} r \\ \theta \end{bmatrix} \\ \mathbf{x}_{f_{new}} &= \mathbf{l}(\mathbf{x}[k], \mathbf{z}_{new}) = \begin{bmatrix} x_v + r\cos(\theta + \phi_v) \\ y_v + r\sin(\theta + \phi_v) \end{bmatrix} \\ \mathbf{G}_v &= \frac{\partial \mathbf{l}}{\partial \mathbf{x}_v} = \begin{bmatrix} 1 & 0 & -r\sin(\theta + \phi_v) \\ 0 & 1 & r\cos(\theta + \phi_v) \end{bmatrix} \\ \mathbf{G}_z &= \frac{\partial \mathbf{l}}{\partial \mathbf{z}_{new}} = \begin{bmatrix} \cos(\theta + \phi_v) & -r\sin(\theta + \phi_v) \\ \sin(\theta + \phi_v) & r\cos(\theta + \phi_v) \end{bmatrix} \end{aligned} \tag{2.22}$$

The size varying property of the state vector is the key difference in SLAM in comparison to a conventional estimation problem for a dynamic system. This size

varying characteristic clearly raises the issue of computational complexity, as the size of the state vector increases.

The other critical issue in a SLAM problem is *data association*, which involves assigning measurements to the features in the environment from which they originate, while rejecting spurious measurements. When a feature is initialized and added to the state vector, the robot needs to extract the location parameters of a feature from the noisy sensor data. This is called *feature extraction* and is one part that demands a accurate and robust data association method. Sometimes, a sequence of consistent measurements is required to initialize a feature. When the robot re-observes a feature to update the state vector, each sensor measurement should be matched with the features in the map. That is, we should figure out which measurement comes from which feature. The Nearest Neighbor (NN) gating [5] and the Joint Compatibility Branch and Bound (JCBB) method developed by Neira and Tardos [63] are frequently utilized techniques for data association.

As already introduced, how to define and parameterize a feature is another critical issue if a feature-based approach is utilized. Once a feature is defined by an appropriate set of parameters, it is relatively easy to manipulate. Only very few real-world objects, however, can be defined with a small number of parameters.

Size variance (computational complexity), noisy observations (data association), and feature parameterization (feature extraction) are the three key issues in the feature-based SLAM problem and have driven most of the research in the robotics field. These issues make the problem different from the conventional state estimation problems for dynamic systems. As mentioned in Chapter 1, this thesis address only the first of these three problems, computational complexity.

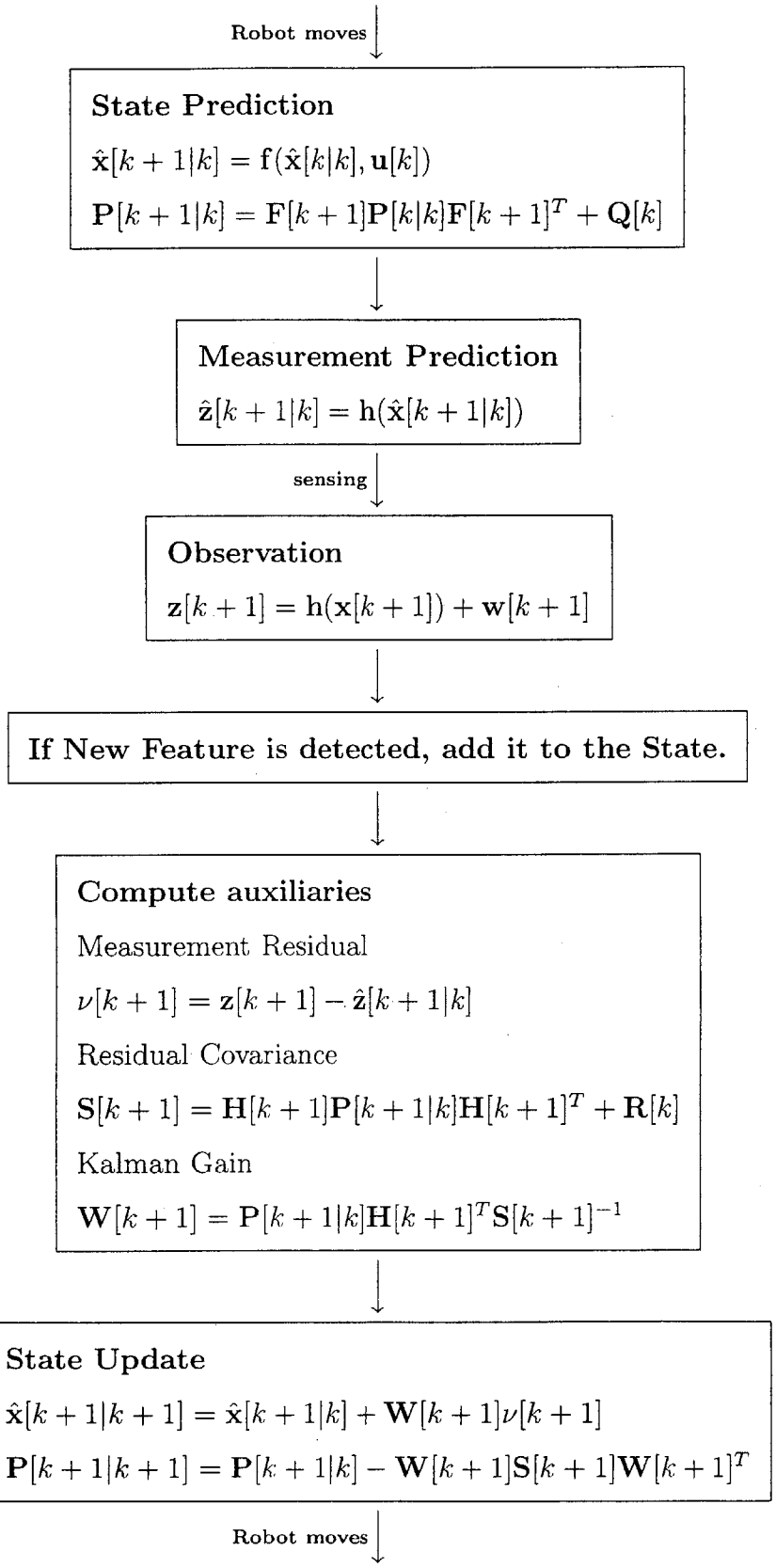

Figure 2-6: Flow chart for a single cycle of Kalman filter SLAM.

2.1.3 Basic SLAM procedure

The followings are the steps performed in executing one cycle of the basic SLAM procedure.

1. Robot moves
2. Predict the state estimate, the state covariance, and the new measurements
3. Obtain new sensor measurements
4. If a new feature is detected, initialize the feature and integrate it into the state
5. Calculate measurement residual and filter gain
6. Update the state estimate and the state covariance

2.2 The Large-Scale SLAM Problem

Ever since the $\mathcal{O}(n^2)$ computational complexity of the Kalman filter approach to the SLAM problem was first identified in the late 1980s, the map-scaling problem has been an important topic for study. This scaling problem is one of the main stumbling blocks for a practical implementation of SLAM solutions in real-world situations. In this section, we review the various methods that have been employed in the attempt to reduce this high computational complexity.

2.2.1 Single map approaches

a. Postponement and the Compressed Filter

The compressed filter by Guivant *et al.* [37] uses the Kalman filter to update dependencies between features. The analysis of the dependencies is similar to the *level-wise update* in the Chapter 3 of this thesis. Since the previous information of the unobserved features is not required to update the robot and observed feature blocks of the covariance matrix, the compressed filter postpones the update of some features'

covariances until the robot approaches and observes them again. This discriminative update can reduce the constant factor of the computation while the robot stays in a local region and also provably consistent. However, all the postponed updates should be completed before the robot re-observes those castoff features. This update requires a significant set of updates.

b. FastSLAM — A Factored Solution to SLAM

Since it uses the Bayesian formulation, the FastSLAM by Montemerlo *et al.* [61] does not require the Gaussian assumptions. FastSLAM decouples the SLAM problem into two separate problems using the conditional independences between feature locations and the whole set of robot poses s^t and of the feature-measurement correspondence variables n^t. Once the SLAM problem is decomposed, the feature location estimation can be executed in a very efficient manner. For the safe separation, however, the whole set of the robot's poses and the correspondence variables (perfect data association) should be perfectly known. FastSLAM utilizes the particle filter to provide the perfect knowledge on the robot's poses, but theoretically an infinite number of particles are necessary for the perfect estimation. Even when a finite number of particles is used for practical purposes, the performance depends upon the number of particles, which may cause a severe computational burden.

c. Sparse Extended Information Filters

The Sparse extended information filter (SEIF) [85, 84] solves the SLAM problem with the information filter form of the Kalman filter, instead of the more typical approach that maintains the covariance matrix. When a standard Kalman filter form is used for the SLAM with static features, the prediction step is executed very efficiently, but the update step suffers $\mathcal{O}(n^2)$ computation. From the duality between the information filter and standard Kalman filter, the information filter can complete the update step very efficiently, but has to execute an $\mathcal{O}(n^3)$ operation for the prediction step. Hence, it would seem that the information filter form does not have any advantage over the standard Kalman filter form, when applied to SLAM.

However, Thrun and his co-authors make a powerful insight that many of the elements of the inverse covariance matrix in SLAM are very nearly zero. By approximating these terms as zero, the SEIF approach transforms the prediction step into an optimization problem using the sparseness of the information matrix. Once the optimization problem is constructed, it can be calculated in a very efficient manner. The method has been used to obtain impressive results with challenging real-world data sets, such as underground mine data sets and the Victoria Park dataset made publicly available by the University of Sydney.

However, the information matrix is not really sparse, but almost sparse. Thus, the SEIF makes an approximation to enforce sparseness. Empirically, it seems that assuming that elements of the inverse covariance are near zero is a much better approximation than assuming that elements of the covariance matrix are zero [18]. Hence, the SEIF method requires empirical tests to verify the consistency of the results.

2.2.2 Submap approaches

In any large-scale engineering or computational problem, a divide-and-conquer strategy seems reasonable to pursue. Hence the idea of breaking the environment into local submaps has intuitive appeal . Examples of submap approaches to SLAM include the sequential map joining method of Tardos *et al.*, the constrained local submap filter proposed by Williams *et al.*, the Atlas Framework developed by Bosse *et al.*, and the Constant Time SLAM (CTS) algorithm developed by Leonard and Newman.

a. Sequential map joining

The sequential map joining [80] and constrained local submap filter methods [86] are essentially the same method developed concurrently by researchers working independently on opposite sides of the globe. They provide a consistent way to join two independent local maps, to find matchings features between two local maps, and to fuse the matched features for updating the merged map. The computational com-

plexity of the sequential map joining is still $\mathcal{O}(n^2)$ even though the constant factor is much less than that of the full covariance SLAM solution. Also, once two maps are merged, they cannot be separated again in a consistent manner, so the matching and fusion should be executed prudently during a mission.

b. Decoupled Stochastic Mapping (DSM)

Decoupled stochastic mapping (DSM) by Feder *et al.* [52, 53, 31] divides the global environment into multiple globally referenced submaps. Each submap has its own SLAM solution (state vector and covariance matrix), which is activated when the robot enters and is deactivated once the robot leaves the submap. During transitions between maps, DSM applies two techniques for activating the state vector and the covariance matrix: (1) cross-map relocation and (2) cross-map update. If cross-map relocation is used, then the transition satisfies consistency (under certain assumptions), but the solution loses the global convergence. If the cross-map update transition technique is utilized, the transition can achieve the global convergence, but the consistency is not satisfied theoretically. Thus, like SEIF's, if the DSM method is used, then empirical validation is necessary to verify consistency.

c. Atlas

The *Atlas* framework [9, 10, 11] for a large-scale SLAM employs multiple interconnected local maps. The local maps of Atlas form a graph structure, and the edges represent the transformation of the coordinate frame. Atlas does not maintain the global origin and one or more local maps compete to explain the current robot's pose and the measurement. Four possible states of local maps capture the traversal of the Atlas graph – Dominant, Mature, Juvenile, and Retired – and maintain the Atlas framework to focus on the local area where the robot navigates. This strategy enables large-scale SLAM to be performed very efficiently. In order to construct one global map from the local maps, however, a nonlinear optimization technique should be utilized, whose computational complexity grows with the number of maps. The Atlas framework has been used to obtain many successful experimental results, such

as mapping the main buildings of the MIT campus. In particular, Atlas excels in dealing with the loop closing problem in SLAM.

Another methods that is closely related to Atlas is the Constant Time SLAM (CTS) algorithm [50]. Like Atlas, CTS uses a network of locally reference submaps. However, for all but the first map, each local map is referenced to one of the features in the environment. This allows the global convergence to be achieved, while maintaining $\mathcal{O}(1)$ complexity (assuming that the data association problem for relocation into previous maps (loop closing) is solved.

2.3 Summary

This chapter has reviewed previous work on large-scale SLAM and has summarized the basic formulation of the Kalman filter solution to SLAM. After consideration of the properties of the SLAM covariance matrix and presentation of a new more general closed form solution to the single degree-of-freedom SLAM problem in Chapter 3, we present an in-depth description and theoretical and experimental analysis of CTS in Chapter 4. We then proceed to describe two new algorithms, CTS 2.0 and Network Optimized SLAM (NOS), that build on the CTS approach to achieve improved performance.

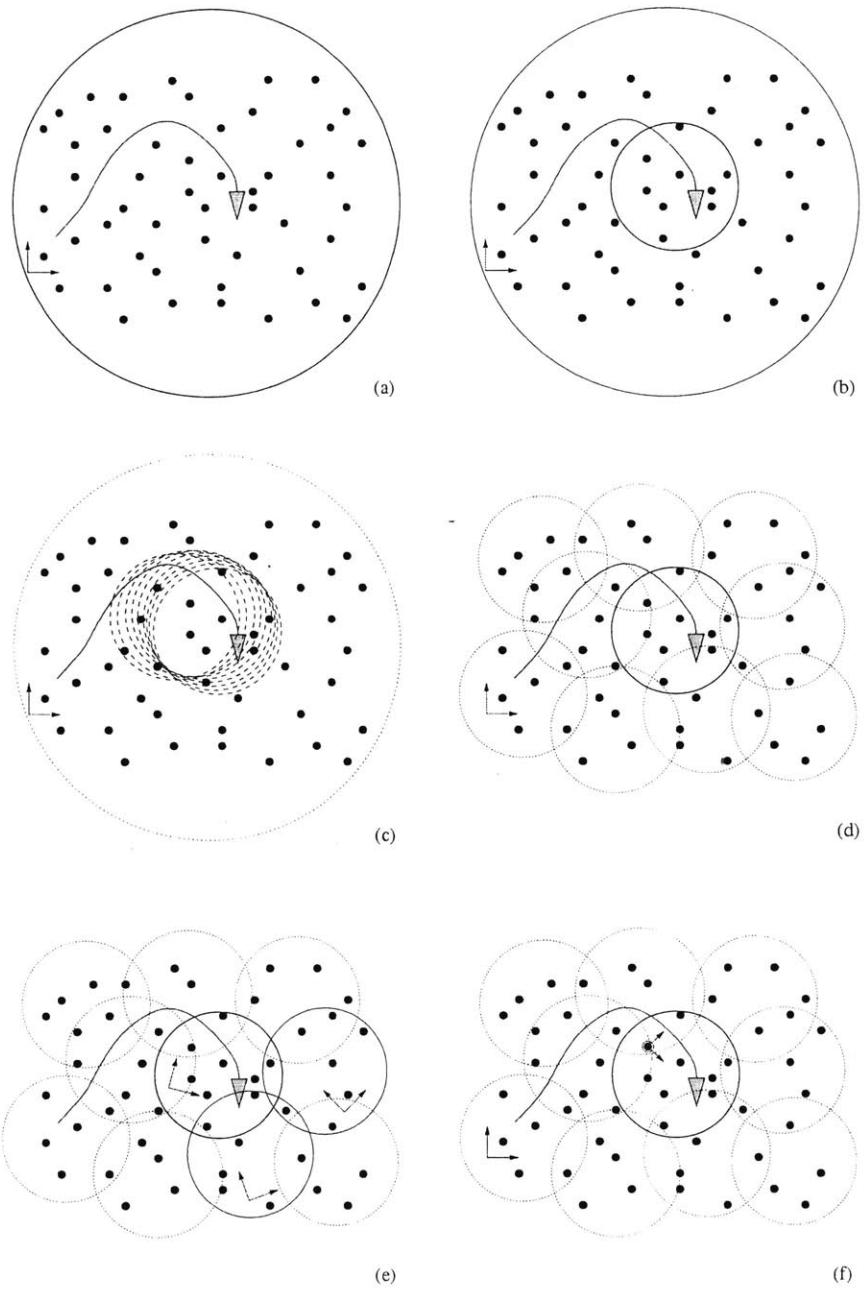

Figure 2-7: (a) The full covariance SLAM solution [76] and FastSLAM [61] employ a single globally-referenced map. (b) Postponement [24, 47] and the compressed filter [36] postpone Kalman updates of features if they are far away from the robot. The active region automatically follows along with the robot position. (c) The SEIF method for SLAM thrun02 focuses on the features in the vicinity of the robot. (d) DSM [55] utilizes a globally referenced multiple submap system. (e) ATLAS [10] employs a network of multiple locally-referenced submaps. At any one time, several submaps compete to explain the current robot pose and sensor observations. (f) Like Atlas, CTS [50] uses multiple overlapping locally-referenced submaps. In CTS, the origin of each local submap is set either at the global origin or on top of a feature that is shared between adjacent submaps.

Chapter 3

Properties of the SLAM Covariance Matrix

This chapter will examine the properties of the covariance matrix for the Kalman filter solution to the linear Gaussian single degree-of-freedom SLAM problem (also known as the MonoRob problem). While previous research has considered the situation in which all features are observed all the time, this chapter considers a more general case that includes both observed and non-observed features. We begin the chapter with a review of several known properties of the covariance matrix, followed by a presentation of several new insights into the covariance behavior. Subsequently, we present the new closed form solution and utilize this solution to analyze several representative cases in which features are observed unequally. This allows us to examine and verify several key aspects of the covariance matrix behavior. These insights are important in considering the behavior of computationally efficient solutions to the large-scale SLAM problem that will be considered next in Chapter 4.

3.1 Introduction

The Kalman filter computes the optimal solution to the SLAM problem when four criteria are satisfied: (1) linear motion and measurement models, (2) zero-mean Gaussian error distributions for the motion and measurement noise, (3) known data asso-

ciation, and (4) representation of the environment in terms of features that are points in an appropriate parameter space [76, 62, 51]. Of course, we know that in many, if not most, real-world SLAM problems, these criteria are *not* satisfied [82]. This is perhaps no more true in the undersea environment, where AUV motion models are quite nonlinear, sonar measurements are error-prone, the data association problem is complex, and the mapping of natural terrain poses difficult issues in environment representation.

Despite all these concerns, we nevertheless believe that it is important to understand the nature of the Kalman filter solution for the Linear Gaussian problem, for two reasons. Firstly, in many real-world situations, existing feature detection and data association techniques can perform well, and angular errors such as the heading of the surveying robot can be kept small, so that the linear Gaussian assumptions can be satisfied. Secondly, from a theoretical perspective, the simplified linear Gaussian SLAM problem has many attributes that generalize to more complex nonlinear SLAM situations, and hence insights into the simpler problem can be generalized.

In the recent literature that has studied the linear Gaussian SLAM problem, some of the most important contributions have been provided by researchers at the University of Sydney, including Csorba [23], Newman [67], Dissanayake *et al.* [27], and Gibbens *et al.* [34]. Following these authors, we designate the single degree-of-freedom linear Gaussian SLAM problem as the "MonoRob" (or mono-dimensional robot) problem.

Newman and his co-authors were the first researchers to investigate the asymptotic behavior of the SLAM covariance matrix for the SLAM problem [67, 27]. This work influenced the SLAM community substantially, helping to build confidence that the SLAM problem could indeed be solved. This work made a number of important claims applicable to the linear Gaussian SLAM problem, for the special case that all features are observed by the robot at every time step. The three key results were:

1. "The determinant of any submatrix of the map covariance matrix decreases monotonically as observations are successively made.

2. In the limit as the number of observations increases, the landmark estimates become fully correlated.

3. In the limit, the covariance associated with any single landmark location estimate is determined only by the initial covariance in the vehicle location estimate [27]."

Based on these three results, a number of claims were made, including:

1. "In the limit, the errors in the estimates of any pair of landmarks becomes fully correlated. This means that given the exact location of any one landmark, the location of any other landmark in the map can also be determined with absolute certainty.

2. As the vehicle moves through the environment taking observations of individual landmarks, the error in the estimates of the relative location between different landmarks reduces monotonically to the point where the map of relative locations is known with absolute precision.

3. As the map converges in the above manner, the error in the absolute location of every landmark (and thus the whole map) reaches a lower bound determined only by the error that existed when the first observation was made. [27]"

In Section 3.2, we consider the same issues for the more general case when only a subset of the features in the environment are observed.

Another second key contribution coming from the University of Sydney, based on research conducted during approximately the same time period, was the development of a closed form solution to the MonoRob problem. The solution strategy developed was to formulate MonoRob as a continuous, rather than a discrete, state estimation problem, and then to solve the resulting Riccati equation. The problem is difficult because, even with simplifying assumptions for the vehicle dynamic model, the algebra becomes extremely tedious and tiresome. Gibbens *et al.* solved this problem by

first considering the case when there were only a small number of features in the environment, and then using a symbolic mathematics package to generalize their result to an arbitrary number of landmarks.

The Gibbens *et al.* solution yielded some valuable insights into rate of convergence and also agreed with the claims of the Newman *et al.* analysis that is summarized above. However, the Gibbens *et al.* solution only applies to the special case when the vehicle makes observations of all features at each time step. Below in Section 3.3 (and Appendix A), we present a more general solution to this problem, obtained by solving the Riccati equation by hand.

Before investigating these two issues further, we begin with some definitions and a review of well-known properties for the linear Gaussian SLAM covariance matrix.

3.2 Properties of the Covariance Matrix of the Linear Gaussian SLAM Solution

Estimation is the process of finding a best value based on observed information [46]. For example, we can estimate the position and velocity of an aircraft using range measurements obtained by one or more ground radar sensors. Usually, an estimate for a stochastic process forms a probability distribution because no deterministic estimation to find a true value is possible. The shape of the distribution shows not only the expected value, but also the confidence level of the expectation.

Mathematically, the shape of the distribution can be described by either a finite or an infinite linear summation of moments of the random variable. The n-th moment of a random variable x is defined as $\mathrm{E}[\mathrm{x}^n] = \int x^n p_{\mathrm{x}}(x)\,\mathrm{d}x$. We call the first two moments as the mean value and the mean squared value respectively. In order to fully characterize the distribution of a random variable, an infinite number of moments must be provided. Fortunately, in many situations, partial characterizations in the form of expectations are enough to approximate distributions to a sufficient accuracy. Also, a few distributions require only several moments to be fully characterized.

For example, if the distribution is known to be Gaussian, the first two moments are enough to fully characterize the distribution. As described in the previous chapter, we assume a Gaussian process for the SLAM solution, so our main goal in approaching a SLAM problem is to determine the best values for the first two moments that characterize the estimated distribution.

The variance (the second order central moment) is often used to characterize the estimated distribution because it is defined as $\mathbf{E}[(x - \mathbf{E}[\mathrm{x}])^2]$ and measures how close the actual value is to the expected mean. The higher the variance, the more uncertain the estimation becomes. On the other hand, a smaller variance implies a higher confidence in the estimated value. Zero variance means the expected mean value is indeed the true value.

When estimation is performed for a multivariate problem, the second moment captures the covariances as well as the variances of the state estimates. Since the estimates now form a vector, the second moment becomes $\mathbf{E}[(\mathrm{X} - \mathbf{E}[\mathbf{x}])(\mathrm{X} - \mathbf{E}[\mathbf{x}])^T]$ rather than $\mathbf{E}[(\mathrm{X} - \mathbf{E}[\mathbf{x}])^2]$. That is, a set of moments forms a matrix whose diagonal elements are variances and off-diagonal terms represent moments of two different estimates, $\mathbf{E}[(x_i - \mathbf{E}[\mathrm{x}_i])(x_j - \mathbf{E}[\mathrm{x}_j])]$. We call these off-diagonal elements the covariances, which measure how strongly two estimates are related to each other. Mathematically, the covariance represents the dependency of the corresponding pair of estimates. The correlation coefficient is the covariance normalized by the product of the corresponding variances. If the correlation coefficient of two corresponding random variables is ± 1, then perfect knowledge of one variable can be obtained from perfect knowledge of the other. On the other hand, if the value of the correlation coefficient is zero, then two estimates are said to be uncorrelated and one estimate does not affect the other estimated result.

As the covariance matrix is updated during Kalman filter estimation for the linear SLAM problem, during certain circumstances it becomes fully correlated; that is, every correlation coefficient reaches ± 1, which implies that one correlated estimate allows us to assume that all other estimates are also perfectly known [27]. Furthermore, the covariance matrix tests the consistency of a SLAM solution and demonstrates the

lowest allowable threshold of the uncertainty that can be achieved.

This section examines the properties of the covariance matrix. It includes a review of general behavior of the covariance matrix as well as the properties specific to the Kalman filter based SLAM problem. Although many of these properties are well known, a review of these properties is useful for clarifying the definitions and terminology that we employ. Further, a clear understanding of these properties is essential for analysis of the new algorithms that will be presented in Chapter 4.

3.2.1 Definition of the covariance matrix

For two random variables, x_1 and x_2, their joint moments (about the origin) are defined by

$$\mathbf{E}[\mathsf{x}_1^n \mathsf{x}_2^m] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1^n x_2^m p_{\mathsf{x}_1,\mathsf{x}_2}(x_1, x_2)\, \mathrm{d}x_1 \mathrm{d}x_2 \quad . \tag{3.1}$$

where $p_{\mathsf{x}_1,\mathsf{x}_2}(x_1, x_2)$ is the joint probability density function. The second order moment $\mathbf{E}[\mathsf{x}_1\mathsf{x}_2] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 x_2 p_{\mathsf{x}_1,\mathsf{x}_2}(x_1, x_2)\, \mathrm{d}x_1 \mathrm{d}x_2$ is called the correlation of x_1 and x_2. Similarly, the $(n_1 + n_2 + \cdots + n_k)$ order joint moment can be written in the form

$$\begin{aligned} &\mathbf{E}[\mathsf{x}_1^{n_1} \mathsf{x}_2^{n_2} \cdots \mathsf{x}_k^{n_k}] \\ &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} x_1^{n_1} x_2^{n_2} \cdots x_k^{n_k} p_{\mathsf{x}_1,\mathsf{x}_2,\cdots,\mathsf{x}_k}(x_1, x_2, \cdots, x_k)\, \mathrm{d}x_1 \mathrm{d}x_2 \cdots \mathrm{d}x_k \quad . \end{aligned} \tag{3.2}$$

As special forms of the expectations, the second order joint central moments are defined as $\mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])^2]$ for a random variable x_1 or $\mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])(x_2 - \mathbf{E}[\mathsf{x}_2])]$ for two random variables, x_1 and x_2. The former measures how closely a random variable is distributed to the mean and is called variance. Since it is squared, the variance is always non-negative. The latter measures how closely two random variables are correlated with each other and is called the cross correlation. The equation

$$\begin{aligned}
&\mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])(x_2 - \mathbf{E}[\mathsf{x}_2])] \\
&\qquad = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x_1 - \mathbf{E}[\mathsf{x}_1])(x_2 - \mathbf{E}[\mathsf{x}_2]) p_{\mathsf{x}_1,\mathsf{x}_2}(x_1, x_2)\, \mathrm{d}x_1 \mathrm{d}x_2 \\
&\qquad\qquad = \mathbf{E}[\mathsf{x}_1\mathsf{x}_2] - \mathbf{E}[\mathsf{x}_1]\mathbf{E}[\mathsf{x}_2] \quad (3.3)
\end{aligned}$$

illustrates that if two random variables x_1 and x_2 are uncorrelated ($\mathbf{E}[\mathsf{x}_1\mathsf{x}_2] = \mathbf{E}[\mathsf{x}_1]\mathbf{E}[\mathsf{x}_2]$) or independent ($p_{\mathsf{x}_1,\mathsf{x}_2}(x_1, x_2) = p_{\mathsf{x}_1}(x_1)p_{\mathsf{x}_2}(x_2)$), their cross correlation becomes zero.

Often, the normalized second order moment of two corresponding variances is utilized to measure cross correlations between random variables. The cross correlation coefficient is defined as

$$\rho = \mathbf{E}\left[\frac{(x_1 - \mathbf{E}(\mathsf{x}_1))(x_2 - \mathbf{E}(\mathsf{x}_2))}{\sqrt{\mathbf{E}[(x_1 - \mathbf{E}(\mathsf{x}_1))^2]}\sqrt{\mathbf{E}[(x_2 - \mathbf{E}(\mathsf{x}_2))^2]}}\right] \tag{3.4}$$

and always has a value $-1 \leq \rho \leq 1$. This will be proved in the next subsection. As described in the introductory section, a ± 1 value of cross correlation enables us to obtain perfect knowledge of one variable from knowledge of the other.

For a jointly Gaussian random vector $\mathsf{X} = [\mathsf{x}_1, \mathsf{x}_2, \cdots, \mathsf{x}_n]^T$, the error mean, the variances, and covariances of elements of $\mathbf{x}$ can be written in the form:

Error mean

$$\mathbf{E}[(\mathsf{X} - \mathbf{E}[\mathbf{x}])] = \begin{bmatrix} \mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])] \\ \mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_2])] \\ \vdots \\ \mathbf{E}[(x_n - \mathbf{E}[\mathsf{x}_n])] \end{bmatrix}, \tag{3.5}$$

Variance

$$
\begin{aligned}
P_{11} &= \mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])^2] \\
P_{22} &= \mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_2])^2] \\
&\vdots \qquad\qquad \vdots \\
P_{nn} &= \mathbf{E}[(x_n - \mathbf{E}[\mathsf{x}_n])^2] \quad ,
\end{aligned}
\tag{3.6}
$$

Cross correlation

$$
\begin{aligned}
P_{12} &= \mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])(x_1 - \mathbf{E}[\mathsf{x}_2])] \\
P_{13} &= \mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_1])(x_1 - \mathbf{E}[\mathsf{x}_3])] \\
&\vdots \qquad\qquad \vdots \\
P_{n-1n} &= \mathbf{E}[(x_{n-1} - \mathbf{E}[\mathsf{x}_{n-1}])(x_n - \mathbf{E}[\mathsf{x}_n])] \quad .
\end{aligned}
\tag{3.7}
$$

In the matrix form, these variances and cross correlations define the covariance matrix for the random vector X; that is, the diagonal elements of the covariance matrix consist of the variances of the states and the off-diagonal elements are composed of the cross correlations. Therefore,

$$
\begin{aligned}
&\mathbf{E}[(\mathsf{X} - \mathbf{E}[\mathsf{x}])(\mathsf{X} - \mathbf{E}[\mathsf{x}])^T] \\
&= \begin{bmatrix}
\mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])^2] & \mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])(x_2 - \mathbf{E}[\mathsf{x}_2])] & \cdots & \mathbf{E}[(x_1 - \mathbf{E}[\mathsf{x}_1])(x_n - \mathbf{E}[\mathsf{x}_n])] \\
\mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_2])(x_1 - \mathbf{E}[\mathsf{x}_1])] & \mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_2])^2] & \cdots & \mathbf{E}[(x_2 - \mathbf{E}[\mathsf{x}_2])(x_n - \mathbf{E}[\mathsf{x}_n])] \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{E}[(x_n - \mathbf{E}[\mathsf{x}_n])(x_1 - \mathbf{E}[\mathsf{x}_1])] & \mathbf{E}[(x_n - \mathbf{E}[\mathsf{x}_n])(x_2 - \mathbf{E}[\mathsf{x}_2])] & \cdots & \mathbf{E}[(x_n - \mathbf{E}[\mathsf{x}_n])^2]
\end{bmatrix} \\
&= \begin{bmatrix}
P_{11} & P_{12} & \cdots & P_{1n} \\
P_{21} & P_{22} & \cdots & P_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
P_{n1} & P_{n2} & \cdots & P_{nn}
\end{bmatrix}
\end{aligned}
\tag{3.8}
$$

represents the matrix form of the covariance matrix. This covariance matrix contains

the second order joint central moments of the random vector X, and is utilized to measure the performance of a stochastic estimation. Especially, for a jointly Gaussian random vector, the covariance matrix fully characterizes the error distribution with the error mean. In the remainder of this section, we will examine general properties of the covariance matrix as well as several properties specific to the SLAM problem using the Kalman filter estimation.

3.2.2 General properties

a. Symmetry

A covariance matrix is always *symmetric* because $P_{ij}(= \mathbf{E}[(x_i - \mathbf{E}[\mathsf{x}_i])(x_j - \mathbf{E}[\mathsf{x}_j])])$ for any indices i and j equals to $P_{ji}(= \mathbf{E}[(x_j - \mathbf{E}[\mathsf{x}_j])(x_i - \mathbf{E}[\mathsf{x}_i])])$. This equality is easily verified from the integral form of the expectation.

b. Cross correlation coefficient

The cross correlation coefficient, defined as the cross correlation of two random variables normalized by the product of their corresponding variances, is bounded between -1 and 1, because its form $\frac{\mathbf{E}[(x_i - \mathbf{E}[\mathsf{x}_i])(x_j - \mathbf{E}[\mathsf{x}_j])]}{\sqrt{\mathbf{E}[(x_i - \mathbf{E}[\mathsf{x}_i])^2]}\sqrt{\mathbf{E}[(x_j - \mathbf{E}[\mathsf{x}_j])^2]}}$ is the same as the *cosine* angle of two vectors $(x_i - \mathbf{E}[\mathsf{x}_i])$ and $(x_j - \mathbf{E}[\mathsf{x}_j])$ in an abstract space. Since a cosine value is always bounded between $+1$ and -1, so is the cross correlation coefficient.

c. Non-negative definiteness

A matrix $\mathbf{A}$ is non-negative definite if the matrix satisfies $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \mathbf{0}$ for any $\mathbf{x}$. In various research fields, the term positive (semi) definite is defined as the same meaning as non-negative definite without clear distinction. This thesis, however, distinguishes sharply between the two terms. A positive definite matrix $\mathbf{A}$ is defined as a non-negative definite matrix that satisfies the equality of $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \mathbf{0}$ only if $\mathbf{x} = \mathbf{0}$. If a non-zero vector $\mathbf{x}$ that satisfies this equality exists, the matrix is defined as a positive semi-definite matrix. According to these definitions, the set of the non-negative definite matrices is the union of the two disjoint sets, positive definite

matrices and positive semi-definite matrices. It should be noted that a positive semi-definite matrix is always singular; that is, its determinant is always zero. In the next subsection, this singular property is employed to verify a non-zero convergent determinant of the covariance matrix of a SLAM solution. Figure 3-1 represents the relationship between the terms, non-negative definite, positive definite, and positive semi-definite.

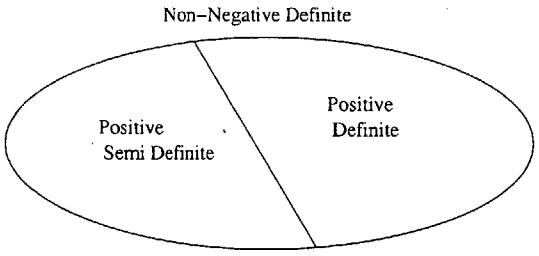

Figure 3-1: Subset structure of the non-negative definite matrices.

From the structure defined, the covariance matrix is always non-negative definite. This property is often used to check the consistency of the estimated outputs. The following is a brief proof of the non-negative definiteness of the covariance matrix.

Proof. In order to show a covariance matrix $\mathbf{P}$ is non-negative definite, it suffices to show $\mathbf{a}^T\mathbf{P}\mathbf{a} \geq 0$ for any vector $\mathbf{a} = [a_1 \ a_2 \ \cdots \ a_n]^T$.

Let $\mathsf{X} = [\mathsf{x}_1 \ \mathsf{x}_2 \ \cdots \ \mathsf{x}_n]^T$ be defined as a random vector that has $\mathbf{P}$ as its covariance matrix and a new random variable y be defined as a weighted sum of X; that is, $\mathsf{y} = \mathbf{a}^T\mathsf{X} = \sum_{i=1}^{n} a_i\mathsf{x}_i$. Then, the expectation $\mathbf{E}[\mathsf{y}]$ becomes $\sum_{i=1}^{n} a_i\mathbf{E}[\mathsf{x}_i]$ and $y - \mathbf{E}[\mathsf{y}]$ can be written in the form $\sum_{i=1}^{n} a_i\,(x_i - \mathbf{E}[\mathsf{x}_i])$.

From this, it is easily derived that $\mathbf{a}^T\mathbf{P}\mathbf{a}$ is the variance of y and, therefore, is always non-negative. Equation 3.9 represents this reasoning.

$$
\begin{aligned}
\mathbf{a}^T\mathbf{P}\mathbf{a} &= \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} & \cdots & P_{1n} \\ P_{21} & P_{22} & \cdots & P_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n1} & P_{n2} & \cdots & P_{nn} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \cdots \\ a_n \end{bmatrix} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j P_{ij} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i a_j \mathbf{E}[(x_i - \mathbf{E}[\mathsf{x}_i])(x_j - \mathbf{E}[\mathsf{x}_j])] \\
&= \mathbf{E}\left[\sum_{i=1}^{n} a_i (x_i - \mathbf{E}[\mathsf{x}_i]) \sum_{j=1}^{n} a_j (x_j - \mathbf{E}[\mathsf{x}_j])\right] \\
&= \mathbf{E}\left[(y - \mathbf{E}[\mathsf{y}])\right)^2] \\
&\geq 0 \quad .
\end{aligned}
\tag{3.9}
$$

d. The determinant as a measure of uncertainty

In stochastic estimation problems, the determinant of the covariance matrix is often used to measure the uncertainty of the estimation. As a scalar value, the determinant measures the uncertainty effectively in various situations. Physically, the determinant of the covariance matrix is interpreted as the *volume* of the probability density. Every covariance matrix $\mathbf{P}$ can be expressed as [41]

$$
\mathbf{P} = \mathbf{R} \begin{bmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_n^2 \end{bmatrix} \mathbf{R}^T \quad . \tag{3.10}
$$

where σ_is are the eigenvalues and $\mathbf{R}$ is the rotation matrix to the principal axes. $\mathbf{R}$ is orthonormal and its determinant is always 1. Each eigenvalue represents the variance along one of the Principal axis. It should be noted that the determinant of the covariance is merely the product of the eigenvalues, because the determinant of the orthonormal rotation matrix is one.

If the distribution of the stochastic estimates follows a Gaussian distribution, the determinant is closed related to the volume of the distribution; that is,

$$\text{Volume} \quad \propto \quad \prod_{i=1}^{n} \sigma_i \quad = \quad |\mathbf{P}| \quad . \tag{3.11}$$

As the summation of the eigenvalues, the trace of the covariance matrix is another tool for measuring the uncertainty. The trace, however, does not include the cross correlation information between two random variables in the covariance matrix, so their dependencies on each other are not captured by the trace.

The determinant of a non-negative definite matrix is always greater than or equal to zero. Also, the inequality, $|\mathbf{A}| > |\mathbf{B}|$ holds if $\mathbf{A} - \mathbf{B}$ is non-negative definite.

Though the determinant possesses many desirable properties to measure and visualize the uncertainty of the estimation, it also has several undesirable properties resulting from its nonlinearity. For example, summation is nonlinear: that is, $|\mathbf{A}| + |\mathbf{B}| \neq |\mathbf{A} + \mathbf{B}|$. Also, when $|\mathbf{B}| > |\mathbf{C}|$, $|\mathbf{A} + \mathbf{B}| > |\mathbf{A} + \mathbf{C}|$ is not generally true. These unfavorable properties of the determinant complicate the map management of the SLAM solution using submap networks (See Chapter 4).

Before moving on the next subsection, let us introduce a important inequality which will be employed to prove the acyclic property of the submap networks in Chapter 4 [58].

$$\text{For non-negative definite} \quad \mathbf{A} \text{ and } \mathbf{B}, \quad |\mathbf{A} + \mathbf{B}| \geq |\mathbf{A}| + |\mathbf{B}| \quad . \tag{3.12}$$

It should be noted that this inequality has the opposite sign of inequality to that of the *triangle inequality*.

3.2.3 Properties specific to the Kalman filter and SLAM

The previous subsection reviewed several general properties of the covariance matrix. In this subsection, several properties of the covariance matrix will be investigated especially when the covariance matrix is computed via a Kalman filter in performing SLAM. The properties analyzed in this section will be examined using the closed form

solution to the MonoRob problem that will be investigated in the next section.

a. Block behavior

Let us begin with the block behavior of the covariance matrix in a SLAM problem. As previously described in this thesis, feature-based SLAM using the Kalman filter is a stochastic estimation problem in which the size of the estimated state varies during the estimation. In a SLAM solution, there exists one or multiple robot which is the subject of the map building and features which are the objects constituting the map. In many situations, these features being mapped are assumed to be static and, therefore, the dynamic (plant) model of the Kalman filter has an effect only on the robot state. Also, these features are added to or deleted from the state vector during a SLAM mission. Because of these characteristics of the SLAM solution, the state vector of a SLAM is usually designed in the following form:

$$\mathsf{X} = \left[\mathbf{x}_v^T \mid \mathbf{x}_{f1}^T,\ \mathbf{x}_{f2}^T,\ \cdots,\ \mathbf{x}_{fn}^T\right]^T \quad , \tag{3.13}$$

where v stands for the vehicle (robot) and fi's for the feature indices. Also, the corresponding structure of the covariance matrix is

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{VV} & \mathbf{P}_{VF} \\ \mathbf{P}_{FV} & \mathbf{P}_{FF} \end{bmatrix} \quad , \tag{3.14}$$

where $\mathbf{P}_{VV}$ is the covariance matrix block of the robot, $\mathbf{P}_{FF}$ is the block of the features being mapped, and $\mathbf{P}_{VF}$ is the cross correlation block between the robot and the features. As a SLAM mission progresses, the size of $\mathbf{P}_{FF}$ varies whenever the robot initializes or eliminates a feature.

Since the covariance matrix forms a partitioned matrix, its determinant can be easily calculated through the Schur complement [41, 58] form defined as

$$\begin{aligned} |\mathbf{P}| &= |\mathbf{P}_{VV}|\left|\mathbf{P}_{FF} - \mathbf{P}_{VF}^T\mathbf{P}_{VV}^{-1}\mathbf{P}_{VF}\right| \\ &= |\mathbf{P}_{FF}|\left|\mathbf{P}_{VV} - \mathbf{P}_{VF}\mathbf{P}_{FF}^{-1}\mathbf{P}_{VF}^T\right| \quad . \end{aligned} \tag{3.15}$$

This special form of the determinant is frequently employed to compare the determinants of two covariance matrix forms in Chapter 4.

b. Non-negative definiteness

Though the non-negative definiteness of the covariance matrix has already been investigated in the previous subsection, the property is re-examined to check if the Kalman filter maintains this property during its two main steps, the prediction and the update written in the following forms:

$$\begin{aligned}
\text{(Prediction)}: \quad & \mathbf{P}[k+1|k] = \mathbf{F}[k]\mathbf{P}[k|k]\mathbf{F}^T[k] + \mathbf{Q}[k] \\
\text{(Update)}: \quad & \mathbf{P}[k+1|k+1] = (\mathbf{I} - \mathbf{W}[k+1]\mathbf{H}[k+1])\mathbf{P}[k+1|k](\mathbf{I} - \mathbf{W}[k+1]\mathbf{H}[k+1])^T \\
& \qquad + \mathbf{W}[k+1]\mathbf{R}[k+1]\mathbf{W}[k+1]^T \quad . \qquad (3.16)
\end{aligned}$$

First, the prediction step in the above equation is composed of the sum of the two square matrices $\mathbf{F}[k]\mathbf{P}[k|k]\mathbf{F}[k]^T$ and $\mathbf{Q}[k]$. Through the Gaussian assumption, $\mathbf{Q}[k]$ is non-negative definite for all time steps k and so is the initial covariance matrix, $\mathbf{P}[0|0]$. From the following theorems [41, 58], the non-negative definiteness of the covariance matrix is maintained after the prediction step.

- If a $n \times n$ square matrix A is non-negative definite, BAB^T is also non-negative definite for all $m \times n$ matrix B.
- If $n \times n$ square matrix A and B are both non-negative definite, their sum $A + B$ is a non-negative definite matrix, too.

Similarly, the update step maintains the property of non-negative definiteness because the process is composed of the sum of two quadratic forms of non-negative definite matrices.

Finally, let us examine whether this non-negative definiteness is maintained when a feature is removed from or added to the state vector. The form of the covariance matrix after a feature is removed from the state is merely a diagonal sub-block of

the original covariance matrix before the removal occurs. Since any diagonal sub-block of a non-negative definite matrix is also non-negative definite, the non-negative definite property is preserved during the feature removal process. When a feature is initialized and added to the state vector, the covariance matrix is transformed through the following procedure:

$$\mathbf{P} \Leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}^T\mathbf{G_x}^T \\ \mathbf{G_x}\mathbf{P} & \mathbf{G_x}\mathbf{P}\mathbf{G_x}^T + \mathbf{G_z}\mathbf{P_z}\mathbf{G_z}^T \end{bmatrix} . \tag{3.17}$$

where $\mathbf{P_z}$ is the feature noise covariance due to the noisy measurement, and $\mathbf{G_x}$ and $\mathbf{G_z}$ are appropriate Jacobians. In order to verify that this resultant matrix is non-negative definite, it suffices to show the following inequality holds for all $\mathbf{A}$ and $\mathbf{B}$ of appropriate dimensions.

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{P} & \mathbf{P}^T\mathbf{G_x}^T \\ \mathbf{G_x}\mathbf{P} & \mathbf{G_x}\mathbf{P}\mathbf{G_x}^T + \mathbf{G_z}\mathbf{P_z}\mathbf{G_z}^T \end{bmatrix} \begin{bmatrix} \mathbf{A}^T \\ \mathbf{B}^T \end{bmatrix}$$
$$= (\mathbf{A} + \mathbf{B}\mathbf{G_x})\,\mathbf{P}\,(\mathbf{A} + \mathbf{B}\mathbf{G_x})^T + \mathbf{B}\mathbf{G_z}\mathbf{P_z}\mathbf{G_z}^T\mathbf{B}^T \geq \mathbf{0} \quad . \tag{3.18}$$

It is straightforward to check if the above inequality holds because it is a summation of two non-negative definite matrices.

c. Monotonicity

As stated previously, the determinant of the covariance matrix measures the uncertainties of the estimates. In a SLAM solution, the uncertainty originates from the spatial pose estimates of the features in the map built by the robot. The most important property of the determinant as the measuring tool of the uncertainties is that the determinant of the feature block ($\mathbf{P}_{FF}$ in Equation 3.14) monotonically decreases as SLAM goes on; that is, as the robot repeats observing features in the environment [26].

This monotonicity may not hold during the feature initialization and the removal

processes. Since those processes, however, occur only once per feature, the changes to the determinant through those processes disappear and their effects are negligible.

During the prediction step of the Kalman filter, monotonicity holds because the system dynamic model governs only the robot block if the environment is static. Therefore, the prediction step does not change the determinant of the feature block of the covariance matrix.

In order to examine the monotonicity during the update step of the Kalman filter, Equation 3.12 is re-written as

$$\text{For non-negative definite } \mathbf{A} \text{ and } \mathbf{B}, \quad |\mathbf{A}+\mathbf{B}| \geq |\mathbf{A}| + |\mathbf{B}| \quad .$$

Now, the determinants of the feature blocks of the covariance matrix before and after the update step of the Kalman filter are compared through the above inequality as

$$\begin{aligned} \mathbf{P}^{+}_{FF} &= \mathbf{P}^{-}_{FF} - \mathbf{WSW}^{T} \\ \mathbf{P}^{+}_{FF} + \mathbf{WSW}^{T} &= \mathbf{P}^{-}_{FF} \\ \left|\mathbf{P}^{+}_{FF} + \mathbf{WSW}^{T}\right| &= \left|\mathbf{P}^{-}_{FF}\right| \\ \left|\mathbf{P}^{+}_{FF}\right| + \left|\mathbf{WSW}^{T}\right| &\leq \left|\mathbf{P}^{-}_{FF}\right| \\ \left|\mathbf{P}^{+}_{FF}\right| &\leq \left|\mathbf{P}^{-}_{FF}\right| \end{aligned} \tag{3.19}$$

where $\mathbf{P}^{-}_{FF}$ and $\mathbf{P}^{+}_{FF}$ represent the feature covariance block before and after the update step respectively, and $\mathbf{W}$ and $\mathbf{S}$ are the Kalman gain and the innovation covariance. Since $\mathbf{S}$ and its quadratic form $\mathbf{WSW}^{T}$ are non-negative definite, the above inequality holds during the update step, therefore, the monotonicity is maintained.

d. Terminal uncertainties

In the previous two subsections, the non-negative definiteness of the covariance matrix and the monotonicity of its determinant are investigated. Also, since the determinant of a non-negative definite matrix is always non-negative, we can conclude that the

determinants of the covariance matrix in SLAM solutions are (1) non-negative and (2) decrease monotonically as the SLAM estimations progress. One question this conclusion raises is " What value does the determinant converge to?".

Many researchers in the field of the SLAM problem have regarded the convergent value as zero. A proof of this zero terminal value of the determinant is provided in [26]. In this thesis, however, the terminal uncertainty is re-investigated with a more rigorous statement of the conditions.

The monotonicity and the non-negative definiteness properties guarantee that the determinant converges to a non-negative value; that is,

$$\lim_{t\to\infty} \left|\mathbf{P}^{+}_{FF}\right| = \left|\mathbf{P}^{-}_{FF}\right| \quad . \tag{3.20}$$

From Equation 3.19, the above limiting form implies that

$$\left|\mathbf{W}\mathbf{S}\mathbf{W}^{T}\right| = \left|\left[\mathbf{P}^{T}_{VF}\ \mathbf{P}_{FF}\right]\mathbf{H}^{T}\mathbf{S}^{-1}\mathbf{H}\begin{bmatrix}\mathbf{P}_{VF}\\ \mathbf{P}_{FF}\end{bmatrix}\right| = \mathbf{0} \quad . \tag{3.21}$$

Since $\mathbf{S}$ is always strictly positive definite, the determinant of the quadratic form $\mathbf{H}\begin{bmatrix}\mathbf{P}_{VF}\\ \mathbf{P}_{FF}\end{bmatrix}$ should be zero to satisfy the given condition. Let us examine what this last condition requires the feature covariance matrix to be. Suppose, for example, that from a certain time step onward, the robot keeps observing only one feature, i, and never re-observes the other features in the state vector until the determinant becomes steady state. The feature block of the covariance matrix will then be:

$$\mathbf{H}\begin{bmatrix}\mathbf{P}_{VF}\\ \mathbf{P}_{FF}\end{bmatrix} = \left[\mathbf{H}_{vi} \;\middle|\; 0 \;\cdots\; \mathbf{H}_I \;\cdots\; 0\right] \left[\begin{array}{c|ccccc} \mathbf{P}_{v1} & \cdots & \mathbf{P}_{vi} & \cdots & \mathbf{P}_{vn} \\ \hline \mathbf{P}_{11} & \cdots & \mathbf{P}_{1i} & \cdots & \mathbf{P}_{1n} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \mathbf{P}_{i1} & \cdots & \mathbf{P}_{ii} & \cdots & \mathbf{P}_{in} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ \mathbf{P}_{n1} & \cdots & \mathbf{P}_{ni} & \cdots & \mathbf{P}_{nn} \end{array}\right] = 0 \quad . \quad (3.22)$$

$$\text{which implies that} \begin{cases} \mathbf{H}_{vi}\mathbf{P}_{v1} + \mathbf{H}_I\mathbf{P}_{i1} = 0 \\ \vdots \\ \mathbf{H}_{vi}\mathbf{P}_{vi} + \mathbf{H}_I\mathbf{P}_{ii} = 0 \\ \vdots \\ \mathbf{H}_{vi}\mathbf{P}_{vn} + \mathbf{H}_I\mathbf{P}_{in} = 0 \end{cases}$$

where $\mathbf{H}_{vi}$ and $\mathbf{H}_I$ are the first derivatives of the measurement model of the Kalman filter with the associated states. Since the measurement in a SLAM problem is executed directly through sensors attached to the robot, the measurement model associated with a certain feature is a function of only the states of the robot and the observed feature. This characteristic of the measurement model causes the other feature block of $\mathbf{H}$ to be the zero matrix as shown in Equation 3.22.

The series of equations given in the above are the conditions that the two properties (monotonicity and the non-negative definiteness) of the covariance matrix must satisfy. In other words, as long as the above conditions are satisfied, the determinant of the feature block becomes steady state; that is, $|\mathbf{P}^-_{FF}|$ becomes the same as $|\mathbf{P}^+_{FF}|$. An analytical solution of a one-dimensional MonoRob SLAM problem is provided in the next section and the following figure captures the situation that the determinant does not converges to zero.

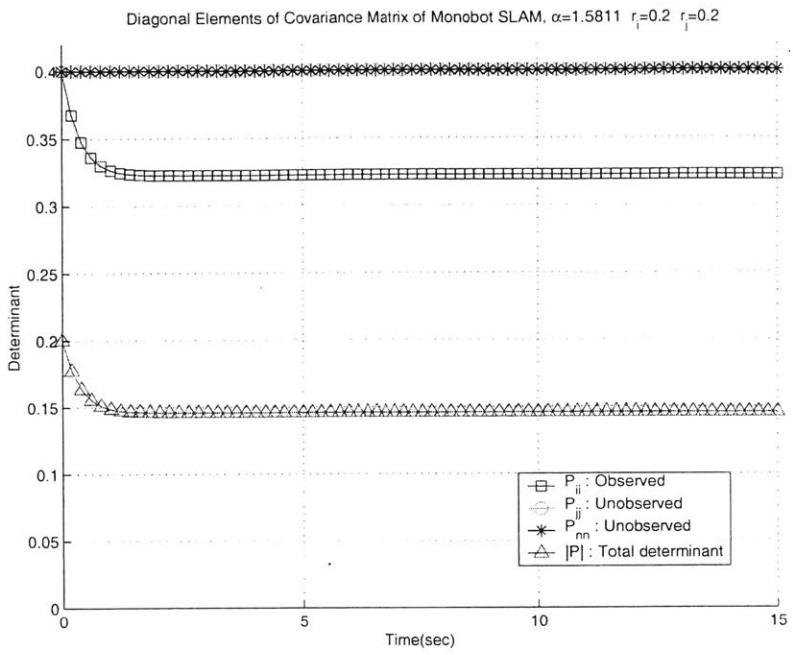

Figure 3-2: The terminal determinant of the covariance matrix does not converge to zero if the robot keeps observing only one feature i.

For a more general situation, suppose that the robot starts observing the feature j after the steady state achieved by observing the feature i. This breaks the steady state conditions and the determinant of the feature covariance matrix decreases again until another steady state condition is reached. Of course, the next steady state determinant must not necessarily to be zero, either. From this, we can conclude that

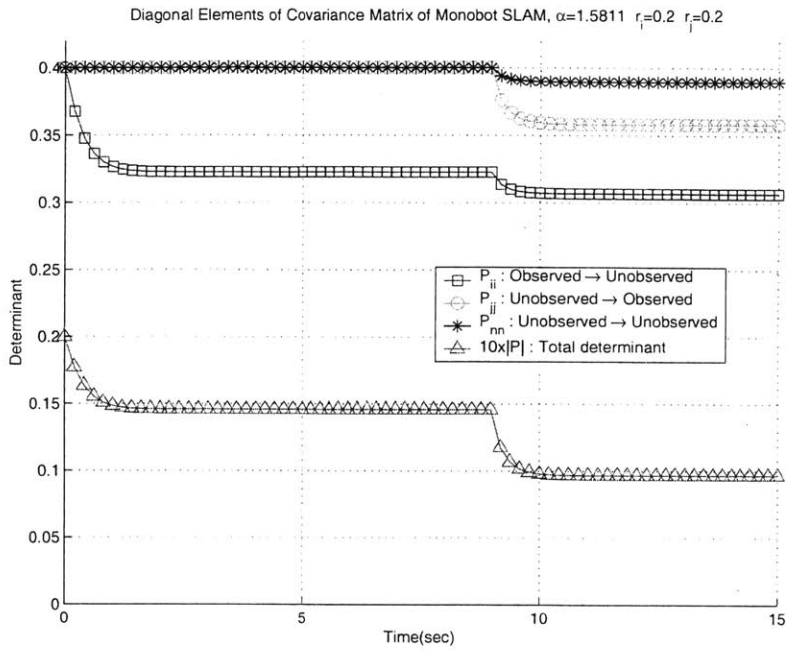

Figure 3-3: The terminal determinant of the covariance matrix must not necessarily to be zero. If the robot switches the observing feature, the determinant converges to another non-zero steady state.

In the case that the robot keeps observing only one feature for a long time, the limiting value must not necessarily to be zero for the feature covariance matrix to become steady state.

From this conclusion, we face another question, which can be stated as "Which condition causes the terminal determinant of the feature covariance matrix equal to zero?" In order to provide the answer to the question, let us examine a situation in which the robot keeps observing features i and j simultaneously. In this situation, the corresponding conditions for the steady state become:

$$\mathbf{H}\begin{bmatrix}\mathbf{P}_{VF}\\ \mathbf{P}_{FF}\end{bmatrix} = \begin{bmatrix}\mathbf{H}_{vi} & 0 & \cdots & \mathbf{H}_I & \cdots & 0 & \cdots & 0\\ \mathbf{H}_{vj} & 0 & \cdots & 0 & \cdots & \mathbf{H}_J & \cdots & 0\end{bmatrix} \left[\begin{array}{cccc} \mathbf{P}_{v1} & \cdots & \mathbf{P}_{vi} & \cdots & \mathbf{P}_{vj} & \cdots & \mathbf{P}_{vn} \\ \hline \mathbf{P}_{11} & \cdots & \mathbf{P}_{1i} & \cdots & \mathbf{P}_{1j} & \cdots & \mathbf{P}_{1n} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{P}_{i1} & \cdots & \mathbf{P}_{ii} & \cdots & \mathbf{P}_{ij} & \cdots & \mathbf{P}_{in} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{P}_{j1} & \cdots & \mathbf{P}_{ji} & \cdots & \mathbf{P}_{jj} & \cdots & \mathbf{P}_{jn} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{P}_{n1} & \cdots & \mathbf{P}_{ni} & \cdots & \mathbf{P}_{nj} & \cdots & \mathbf{P}_{nn} \end{array}\right] \tag{3.23}$$

$$= \mathbf{0} \quad ,$$

which implies

$$\begin{cases} \mathbf{H}_{vi}\mathbf{P}_{v1} + \mathbf{H}_I\mathbf{P}_{i1} = \mathbf{0}, & \mathbf{H}_{vj}\mathbf{P}_{v1} + \mathbf{H}_J\mathbf{P}_{j1} = \mathbf{0} \\ \vdots & \vdots \\ \mathbf{H}_{vi}\mathbf{P}_{vi} + \mathbf{H}_I\mathbf{P}_{ii} = \mathbf{0}, & \mathbf{H}_{vj}\mathbf{P}_{vi} + \mathbf{H}_J\mathbf{P}_{ji} = \mathbf{0} \\ \vdots & \vdots \\ \mathbf{H}_{vi}\mathbf{P}_{vj} + \mathbf{H}_I\mathbf{P}_{ij} = \mathbf{0}, & \mathbf{H}_{vj}\mathbf{P}_{vj} + \mathbf{H}_J\mathbf{P}_{jj} = \mathbf{0} \\ \vdots & \vdots \\ \mathbf{H}_{vi}\mathbf{P}_{vn} + \mathbf{H}_I\mathbf{P}_{in} = \mathbf{0}, & \mathbf{H}_{vj}\mathbf{P}_{vn} + \mathbf{H}_J\mathbf{P}_{jn} = \mathbf{0} \end{cases} . \tag{3.24}$$

These series of conditions are simplified if two features are the same type, which makes $\mathbf{H}_I = \mathbf{H}_J (= \mathbf{H}_F)$. In the following, we will see the above conditions (3.24) induce the determinant of the feature block to be zero in certain measurement models.

Suppose the measurement model is written in the following form [1]:

$$\mathbf{h} = \begin{pmatrix} (x_f - x_v)\cos(\phi_v) + (y_f - y_v)\sin(\phi_v) \\ -(x_f - x_v)\sin(\phi_v) + (y_f - y_v)\cos(\phi_v) \\ \theta_f - \phi_v \end{pmatrix} . \tag{3.25}$$

From this measurement model, the corresponding Jacobians $H_{vf}\Big|_{(f=i,j)}$ and H_F become

[1] This is the measurement model that is employed to process the post-processing result of a SLAM experiment in the next chapter.

$$H_{vf}\Big|_{(f=i,j)} = \begin{bmatrix} -\cos(\phi_v) & -\sin(\phi_v) & (-(x_f - x_v)\sin(\phi_v) + (y_f - y_v)\cos(\phi_v)) \\ \sin(\phi_v) & -\cos(\phi_v) & (-(x_f - x_v)\cos(\phi_v) - (y_f - y_v)\sin(\phi_v)) \\ 0 & 0 & -1 \end{bmatrix}$$

$$H_F = \begin{bmatrix} \cos(\phi_v) & \sin(\phi_v) & 0 \\ -\sin(\phi_v) & \cos(\phi_v) & 0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{3.26}$$

It should be noted that the third row of $H_{vf}\Big|_{(f=i,j)}$ is not a function of feature i nor feature j. With these facts, the k-th row of the conditions (3.24) is now written as

$$\begin{array}{l|l}
\mathbf{H}_{vi}\mathbf{P}_{vk} + \mathbf{H}_I\mathbf{P}_{ik} = \mathbf{0}, & \mathbf{H}_{vj}\mathbf{P}_{vk} + \mathbf{H}_J\mathbf{P}_{jk} = \mathbf{0} \\
\begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{vk}^{11} & \mathbf{P}_{vk}^{12} & \mathbf{P}_{vk}^{13} \\ \mathbf{P}_{vk}^{21} & \mathbf{P}_{vk}^{22} & \mathbf{P}_{vk}^{23} \\ \mathbf{P}_{vk}^{31} & \mathbf{P}_{vk}^{32} & \mathbf{P}_{vk}^{33} \end{bmatrix} & \begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{vk}^{11} & \mathbf{P}_{vk}^{12} & \mathbf{P}_{vk}^{13} \\ \mathbf{P}_{vk}^{21} & \mathbf{P}_{vk}^{22} & \mathbf{P}_{vk}^{23} \\ \mathbf{P}_{vk}^{31} & \mathbf{P}_{vk}^{32} & \mathbf{P}_{vk}^{33} \end{bmatrix} \\
+ \begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ik}^{11} & \mathbf{P}_{ik}^{12} & \mathbf{P}_{ik}^{13} \\ \mathbf{P}_{ik}^{21} & \mathbf{P}_{ik}^{22} & \mathbf{P}_{ik}^{23} \\ \mathbf{P}_{ik}^{31} & \mathbf{P}_{ik}^{32} & \mathbf{P}_{ik}^{33} \end{bmatrix} = \mathbf{0} & + \begin{bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{jk}^{11} & \mathbf{P}_{jk}^{12} & \mathbf{P}_{jk}^{13} \\ \mathbf{P}_{jk}^{21} & \mathbf{P}_{jk}^{22} & \mathbf{P}_{jk}^{23} \\ \mathbf{P}_{jk}^{31} & \mathbf{P}_{jk}^{32} & \mathbf{P}_{jk}^{33} \end{bmatrix} = \mathbf{0} \\
\begin{bmatrix} \bullet & \bullet & \bullet \\ -\mathbf{P}_{vk}^{31} & -\mathbf{P}_{vk}^{32} & -\mathbf{P}_{vk}^{33} \end{bmatrix} = \begin{bmatrix} \bullet & \bullet & \bullet \\ \mathbf{P}_{ik}^{31} & \mathbf{P}_{ik}^{32} & \mathbf{P}_{ik}^{33} \end{bmatrix} & \begin{bmatrix} \bullet & \bullet & \bullet \\ -\mathbf{P}_{vk}^{31} & -\mathbf{P}_{vk}^{32} & -\mathbf{P}_{vk}^{33} \end{bmatrix} = \begin{bmatrix} \bullet & \bullet & \bullet \\ \mathbf{P}_{jk}^{31} & \mathbf{P}_{jk}^{32} & \mathbf{P}_{jk}^{33} \end{bmatrix}
\end{array} \tag{3.27}$$

$$\therefore \begin{bmatrix} \mathbf{P}_{ik}^{31} & \mathbf{P}_{ik}^{32} & \mathbf{P}_{ik}^{33} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{jk}^{31} & \mathbf{P}_{jk}^{32} & \mathbf{P}_{jk}^{33} \end{bmatrix} \tag{3.28}$$

where $k = 1, \cdots, i, \cdots, j, \cdots, n$. Consequently, two columns in the feature covariance matrix are identical, which causes the feature covariance matrix to be singular; that is, the determinant to be zero. From these examinations, we can conclude that

The terminal value of the determinant of the feature covariance matrix approaches zero when at least two same type features are observed simul-

taneously as well as infinitely many times.

The following figure from the next subsection illustrates this result.

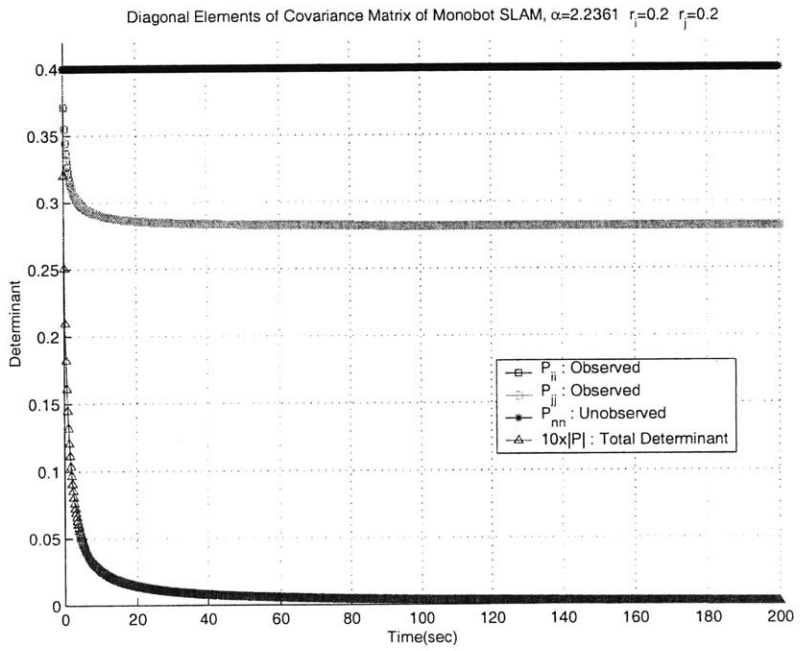

Figure 3-4: If two features of the same type are observed together, the terminal determinant of the covariance matrix converges to zero. This does not imply that the determinant of other feature block also converges to zero.

It should also be noted that this condition cannot still constrain all the features in the state vector to become fully correlated when the determinant becomes zero. Although the cross correlation coefficient between the two features being simultaneously observed becomes one, the cross correlation coefficients with the other features will not necessarily have an absolute value of unity. Figure 3.2.3 captures this situation.

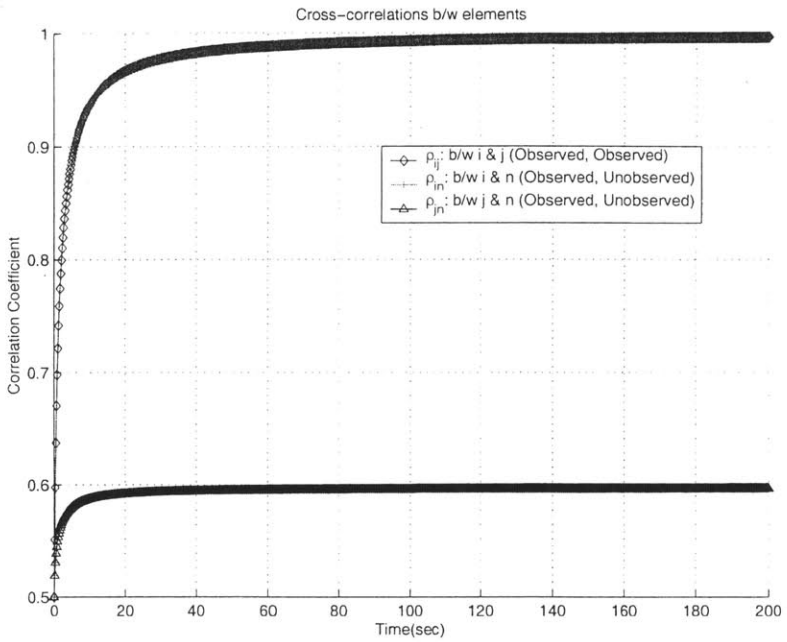

Figure 3-5: Even though the terminal determinant converges to zero, the cross correlations between unobserved features are not necessarily one.

While the cross correlation coefficient of the observed feature pair converges to one, the coefficients associated with unobserved features do not converge to one. This can be interpreted as indicating that it is not possible to fully correlate the covariance matrix unless every feature is observed continually.

e. Behavior during the compounding process

The compounding process is one of the key operations of the feature-based SLAM estimation using the Kalman filter and is employed to maintain the spatial relationships between the global origin and the features' estimated locations [76]. During a SLAM mission, all the observations are obtained through the surveying robot's sensors. The acquired spatial measurements, therefore, are always referenced with respect to the robot's pose. In order to maintain the estimated state vector of the Kalman filter to have globally referenced values, those local measurements should be transformed to the global coordinate frame. The compounding process combines two local spatial relationships and constructs a merged relationship; That is, when there are two local

relationships from i to j and from j to k, the compounding process constructs the relation from i to k as shown in Figure 3-6.

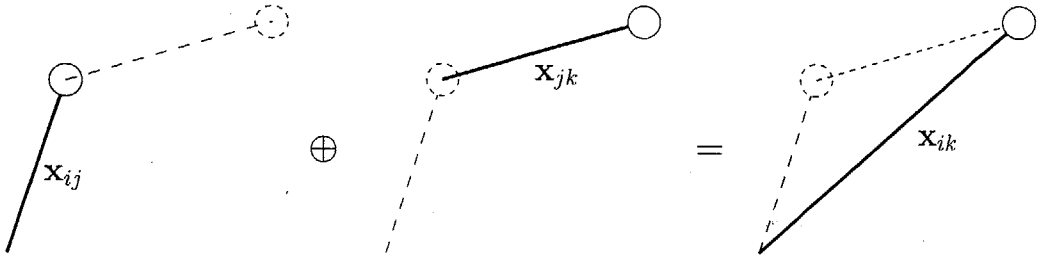

Figure 3-6: The compounding process is a binary operation that combines two spatial relationships. In this figure, $\mathbf{x}_{ik}$ is evaluated by compounding $\mathbf{x}_{ij}$ and $\mathbf{x}_{jk}$. This operation is also written in the form: $T_j^i \oplus T_k^j = T_k^i$ to emphasize that it is a coordinate transformation.

In this subsection, two important properties of the compounding operation are investigated. First, in linear cases and in nonlinear cases after linearization, the spatial relationships are associative under the compounding operation ($\oplus$ is used in this thesis). This means that if we define the pose of j relative to i as $\mathbf{x}_{ij}$,

$$(\mathbf{x}_{ij} \oplus \mathbf{x}_{jk}) \oplus \mathbf{x}_{kl} = \mathbf{x}_{ij} \oplus (\mathbf{x}_{jk} \oplus \mathbf{x}_{kl}) \tag{3.29}$$

holds. The corresponding covariance matrix is also *associative*. In other words, the covariance matrix $\mathbf{P}_{il}$ by compounding $\mathbf{P}_{ik}$ and $\mathbf{P}_{kl}$ is the same as the covariance matrix $\mathbf{P}_{il}$ by compounding $\mathbf{P}_{ij}$ and $\mathbf{P}_{jl}$. Consequently, the evaluating order does not affect the final result.

Secondly, the relative spatial relationship $\mathbf{x}_{il}$ (or T_l^i) is uniquely determined by feature i and feature j if they are maintained in one state vector of a SLAM solution; that is,

$$\begin{aligned} \mathbf{x}_{il} &= \mathbf{x}_{ij} \oplus \mathbf{x}_{jk} \oplus \mathbf{x}_{kl} \\ &= \mathbf{x}_{ik} \oplus \mathbf{x}_{kl} \\ &= \mathbf{x}_{ij} \oplus \mathbf{x}_{jk} \end{aligned} \tag{3.30}$$

holds. This property implies that every pairwise relationship in a map is unique and independent of the route connecting those two features. This is a very useful property and is employed in comparison to the uncertainties in the algorithm suggested in the next chapter.

f. Behavior during root shifting

The state vector of a SLAM solution contains spatial poses of the robot and the features with respect to the origin of a Cartesian coordinate frame. If the origin is set on top of a feature (*root feature*), the elements of the state vector become the relative poses of the other features to the root feature. Since the root feature adheres to the origin, its spatial pose and uncertainty are always zero and therefore are excluded from the state vector.

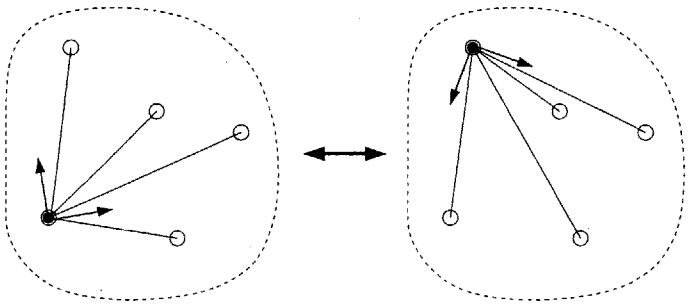

Figure 3-7: The determinant of the covariance matrix of the features is not changed by root shifting.

A special operation, termed root shifting [50], is a transformation operation that changes the origin of a local frame from one feature to another feature in the map. During this swapping of the origin, the values of the state vector as well as the values of the covariance matrix are changed so that all the information is referenced with respect to the new origin. (Root shifting is discussed in much more detail in Chapter 4.)

g. Level-wise update

Another important property of the covariance matrix to be reviewed is that the updated information flow during the Kalman update step has two special characteristics: (1) one way and (2) level-wise.

Suppose the surveying robot observes one feature i, while there are two other features (feature j and feature k) in the map. When the update step is executed in the Kalman filter, each element in the covariance matrix is updated according to the following rules.

- The updated covariance block of the robot $\mathbf{P}_{vv}^{+}$, the observed feature $\mathbf{P}_{ii}^{+}$, and their cross correlation $\mathbf{P}_{vi}^{+}$ depends only on the previous values of ($\mathbf{P}_{vv}^{-}$, $\mathbf{P}_{ii}^{-}$, and $\mathbf{P}_{vi}^{-}$). For example, $\mathbf{P}_{ij}^{-}$ is not necessary to update $\mathbf{P}_{vi}$.

- The cross correlations between the observed feature and the other unobserved features ($\mathbf{P}_{ij}^{+}$ and $\mathbf{P}_{ik}^{+}$) and between the robot and unobserved features ($\mathbf{P}_{vj}^{+}$ and $\mathbf{P}_{vk}^{+}$) depend only on the previous values of ($\mathbf{P}_{ij}^{-}$, $\mathbf{P}_{ik}^{-}$, $\mathbf{P}_{vj}^{-}$ and $\mathbf{P}_{vk}^{-}$) as well as the above information ($\mathbf{P}_{vv}^{-}$, $\mathbf{P}_{ii}^{-}$, and $\mathbf{P}_{vi}^{-}$).

- Finally, the covariance block of unobserved features ($\mathbf{P}_{jj}^{+}$ and $\mathbf{P}_{kk}^{+}$) and their cross correlations ($\mathbf{P}_{jk}^{+}$) depend on all of the previous values associated with the robot and the observed feature.

Figure 3-8 shows these update dependencies. It should be noted that the update is executed level-wise and the information dependencies always flow from the upper layers to the lower layers; that is, no upward information flow in the update step exists during a SLAM estimation. This means that it is not necessary to keep updating an unobserved feature to update an observed feature and the robot unless the robot sees the unobserved feature again. This is the key insight behind the postponement method of Davison and Knight and the Compressed Filter of Guivant and Nebot.

Figure 3-8 and the table that follows it show the dependencies between the elements. The first layer includes $\mathbf{P}_{vv}$, $\mathbf{P}_{ii}$, and $\mathbf{P}_{vi}$ and are updated first. The cross correlations $\mathbf{P}_{vj}$, $\mathbf{P}_{vk}$, $\mathbf{P}_{ij}$, and $\mathbf{P}_{ik}$ form the second layer and their updates depend

on the information of the first group. Finally, the remaining blocks are updated using all the information of the preceding groups.

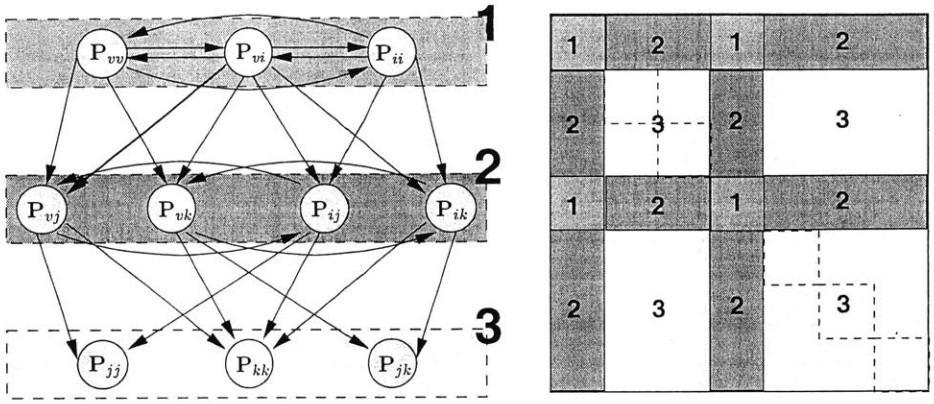

Figure 3-8: Level-wise update and dependencies in the Kalman filter update. The updated covariance block of the robot $\mathbf{P}^+_{vv}$, the observed feature $\mathbf{P}^+_{ii}$, and their cross correlation $\mathbf{P}^+_{vi}$ depends only on the previous values of ($\mathbf{P}^-_{vv}$, $\mathbf{P}^-_{ii}$, and $\mathbf{P}^-_{vi}$). For example, $\mathbf{P}^-_{ij}$ is not necessary to update $\mathbf{P}_{vi}$. The cross correlations between the observed feature and the other unobserved features ($\mathbf{P}^+_{ij}$ and $\mathbf{P}^+_{ik}$) and between the robot and unobserved features ($\mathbf{P}^+_{vj}$ and $\mathbf{P}^+_{vk}$) depend only on the previous values of ($\mathbf{P}^-_{ij}$, $\mathbf{P}^-_{ik}$, $\mathbf{P}^-_{vj}$ and $\mathbf{P}^-_{vk}$) as well as the above information ($\mathbf{P}^-_{vv}$, $\mathbf{P}^-_{ii}$, and $\mathbf{P}^-_{vi}$). Finally, the covariance block of unobserved features ($\mathbf{P}^+_{jj}$ and $\mathbf{P}^+_{kk}$) and their cross correlations ($\mathbf{P}^+_{jk}$) depend on all of the previous values associated to the robot and the observed feature.

	$\mathbf{P}^-_{vv}$	$\mathbf{P}^-_{vi}$	$\mathbf{P}^-_{ii}$	$\mathbf{P}^-_{vj}$	$\mathbf{P}^-_{vk}$	$\mathbf{P}^-_{ij}$	$\mathbf{P}^-_{ik}$	$\mathbf{P}^-_{jj}$	$\mathbf{P}^-_{kk}$	$\mathbf{P}^-_{jk}$
$\mathbf{P}^+_{vv}$	○	○	○							
$\mathbf{P}^+_{vi}$	○	○	○							
$\mathbf{P}^+_{ii}$	○	○	○							
$\mathbf{P}^+_{vj}$	○	○		○		○				
$\mathbf{P}^+_{vk}$	○	○			○	○				
$\mathbf{P}^+_{ij}$		○	○	○		○				
$\mathbf{P}^+_{ik}$		○	○		○		○			
$\mathbf{P}^+_{jj}$				○		○		○		
$\mathbf{P}^+_{kk}$					○		○		○	
$\mathbf{P}^+_{jk}$				○	○	○	○			○

3.3 Analytical Solution of a More General Case for the MonoRob SLAM Problem

So far, several important properties of the covariance matrix have been examined. Though all the properties discussed above have been proven theoretically, it is always desirable to illustrate that the properties hold in practice. In this section, a simple SLAM problem is solved analytically and the time-domain behavior of the covariance matrix is investigated. The problem considered in this section is the continuous-time, one-dimensional SLAM problem with n features. A solution is obtained by solving the corresponding Riccati equation in an explicit form. This enables a close examination of the time-evolution of the solution properties described in Section 3.2.

As mentioned in Section 3.1, the closed-form solution to the MonoRob problem was first published by Gibbens *et al.* [34] for the special case in which the robot observes all features of the time. In this section, we present a new solution with more complicated initial conditions and with several key processes included such as the initialization of new features. Most importantly, we can consider the partial observation case, in which not all features are observed at each time step.

We begin by reviewing the statement of the problem provided in [34]. We assume that the environment is one-dimensional and the robot's velocity only depends on the control input and the system noise; that is, it can be written as $\dot{x}_v(t) = u(t) + w$. In the following expressions, $\mathbf{x}(t)$ is the state vector defined as $[x_v(t) \quad x_1 \quad \cdots \quad x_n]^T$. $\mathbf{Q} = \mathbf{E}[\mathbf{w}\mathbf{w}^T] = q$ and $\mathbf{R} = \mathbf{E}[\mathbf{v}\mathbf{v}^T]$ designate the system noise and the measurement noise, respectively.

- Dynamic Model

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\left(\mathbf{u}(t) + \mathbf{w}(t)\right) \tag{3.31}$$

  where $\mathbf{F} = \mathbf{0}$ and $\mathbf{G} = [1 \ 0 \ \cdots \ 0]^T$.

- Measurement Model

$$\hat{\mathbf{z}}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{v} \quad . \tag{3.32}$$

An observation $z_i(t)$ is defined as the distance from feature i to the robot's location $x_v(t)$, that is, $z_i(t) = x_i - x_v(t) + v_i$. The measurement vector $\mathbf{z}(t)$ is composed of these measurements. Therefore, $\mathbf{H}(t)$ forms an $m \times (n+1)$ matrix where m is the number of

measurements available at time t. In [34], this observation matrix is assumed to always be an $n \times (n+1)$ matrix. This means that all features are always observed during a mission, an assumption that is typically not true for most SLAM problems. By considering the more general case, we will be able to examine several properties associated with the partially observable situation such as the terminal uncertainty for the covariance matrix.

We consider the following two observation matrices:

$$\mathbf{H} = \begin{bmatrix} -1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad \text{(only one feature is observed)}$$
$$\mathbf{H} = \begin{bmatrix} -1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}, \quad \text{(two features are observed).} \tag{3.33}$$

The closed form solution of the covariance matrix is constructed by solving the following differential Riccati equation [34]:

$$\begin{aligned} \dot{\mathbf{P}}(t) &= \mathbf{F}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T - \mathbf{P}(t)^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}(t) \\ \mathbf{P}(t) &= \mathbf{U}(t)\mathbf{V}^{-1}(t) \quad . \end{aligned} \tag{3.34}$$

As indicated above, the resulting solution is evaluated by the product of the two matrices $\mathbf{U}(t)$ and $\mathbf{V}^{-1}(t)$ satisfying the following conditions:

$$\begin{bmatrix} \dot{\mathbf{U}}(t) \\ \dot{\mathbf{V}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{F} & \mathbf{G}\mathbf{Q}\mathbf{G}^T \\ \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} & -\mathbf{F}^T \end{bmatrix} \begin{bmatrix} \mathbf{U}(t) \\ \mathbf{V}(t) \end{bmatrix} \tag{3.35}$$

with

$$\begin{bmatrix} \mathbf{U}(0) \\ \mathbf{V}(0) \end{bmatrix} = \begin{bmatrix} \mathbf{P}(0) \\ \mathbf{I} \end{bmatrix} \quad , \tag{3.36}$$

where $\mathbf{P}(0)$ is the initial state covariance matrix and $\mathbf{I}$ is a identity matrix of an appropriate dimension.

In Gibbens *et al.* [34], the initial condition $\mathbf{P}(0)$ is set to a diagonal matrix, which is not applicable to more general situations. In this thesis, an arbitrary initial covariance matrix

is considered so that the behavior of the solution can be investigated for more realistic situations, such as changing the identity of the observed feature (for example, from feature i to feature j).

Under this more general formulation (partial observations and an arbitrary initial covariance matrix), the following scenarios can be implemented:

- The robot observes only one feature from a time step t_0.
- The robot observes the feature i for a long time and switches its observation to the feature j.
- The robot observes two features (i and j) simultaneously and the switches observe feature k instead of feature i, while continuing to observe feature j. (Features i and k are never observed simultaneously.)

Each of these scenarios can be executed by taking the appropriate observation matrix $\mathbf{H}$ and the proper initial conditions. For example, the closed form of the second case in the list utilizes two observation matrices, $\mathbf{H} = [-1 \quad 0 \quad \cdots \quad 1 \quad \cdots \quad 0 \quad \cdots \quad 0]$ and $\mathbf{H} = [-1 \quad 0 \quad \cdots \quad 0 \quad \cdots \quad 1 \quad \cdots \quad 0]$ with appropriate initial conditions. In Appendix A, the solution procedures for the the two cases — (1) one feature observed and (2) two features observed — are briefly summarized. We now present the resulting values for the covariance matrix for each of these two cases.

3.3.1 Closed form solutions

a. Observation of only one feature

To solve the case in which the robot observes only one feature, the observation matrix $\mathbf{H}$ is defined to be

$$\mathbf{H} = \begin{bmatrix} -1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} . \tag{3.37}$$

For the purpose of simplicity, the index i is set to the observed feature and the indices j and n are assigned to the unobserved features in the following solutions. Therefore, $\mathbf{P}_{ii}(t)$ is the variance of the observed feature and $\mathbf{P}_{jn}(t)$ is the cross correlation between two unobserved features. All the covariance forms in the right hand side ($\mathbf{P}_{vv}$, $\mathbf{P}_{vi}$, etc) are

the initial values. The symbols q and α $(= \frac{1}{r_i})$ represent the system noise variance and the inverse of the measurement noise variance respectively.

$$\mathbf{P}_{vv}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right)+\sqrt{\frac{\alpha}{q}}\left[P^0_{ii}P^0_{vv}-{P^0_{iv}}^2+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}+e^{-\sqrt{q\alpha}t}\left\{\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right)+\sqrt{\frac{\alpha}{q}}\left[-P^0_{ii}P^0_{vv}+{P^0_{iv}}^2+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}+\left\{P^0_{vi}-P^0_{ii}\right\}}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.38}$$

$$\mathbf{P}_{vi}(t) = \frac{e^{\sqrt{q\alpha}t}\left(P^0_{ii}P^0_{vv}-P^0_{iv}P^0_{vi}+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right)+e^{-\sqrt{q\alpha}t}\left(-P^0_{ii}P^0_{vv}+P^0_{iv}P^0_{vi}+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right)+2\sqrt{\frac{q}{\alpha}}(P^0_{vi}-P^0_{ii})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.39}$$

$$\mathbf{P}_{vj}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{jv}P^0_{vi}+P^0_{jv}P^0_{ii}+P^0_{ji}P^0_{vv}-P^0_{ji}P^0_{iv}+P^0_{ji}\sqrt{\frac{q}{\alpha}}\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{jv}P^0_{vi}-P^0_{jv}P^0_{ii}-P^0_{ji}P^0_{vv}+P^0_{ji}P^0_{iv}+P^0_{ji}\sqrt{\frac{q}{\alpha}}\right\}+2\sqrt{\frac{q}{\alpha}}(P^0_{jv}-P^0_{ji})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.40}$$

$$\mathbf{P}_{vn}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{nv}P^0_{vi}+P^0_{nv}P^0_{ii}+P^0_{ni}P^0_{vv}-P^0_{ni}P^0_{iv}+P^0_{ni}\sqrt{\frac{q}{\alpha}}\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{nv}P^0_{vi}-P^0_{nv}P^0_{ii}-P^0_{ni}P^0_{vv}+P^0_{ni}P^0_{iv}+P^0_{ni}\sqrt{\frac{q}{\alpha}}\right\}+2\sqrt{\frac{q}{\alpha}}(P^0_{nv}-P^0_{ni})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.41}$$

$$\mathbf{P}_{iv}(t) = \frac{e^{\sqrt{q\alpha}t}\left(P^0_{ii}P^0_{vv}-P^0_{iv}P^0_{vi}+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right)+e^{-\sqrt{q\alpha}t}\left(-P^0_{ii}P^0_{vv}+P^0_{iv}P^0_{vi}+P^0_{ii}\sqrt{\frac{q}{\alpha}}\right)+2\sqrt{\frac{q}{\alpha}}(P^0_{vi}-P^0_{ii})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.42}$$

$$\mathbf{P}_{ii}(t) = \frac{e^{\sqrt{q\alpha}t}\left[P^0_{vv}P^0_{ii}-{P^0_{iv}}^2+P^0_{ii}\sqrt{\frac{q}{\alpha}})\right]+e^{-\sqrt{q\alpha}t}\left[{P^0_{iv}}^2-P^0_{vv}P^0_{ii}+P^0_{ii}\sqrt{\frac{q}{\alpha}})\right]}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.43}$$

$$\mathbf{P}_{ij}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{jv}(P^0_{vi}-P^0_{ii})+P^0_{ji}(P^0_{vv}-P^0_{iv}+\sqrt{\frac{q}{\alpha}})\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{jv}(P^0_{vi}-P^0_{ii})-P^0_{ji}(P^0_{vv}-P^0_{iv}-\sqrt{\frac{q}{\alpha}})\right\}}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.44}$$

$$\mathbf{P}_{in}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{nv}(P^0_{vi}-P^0_{ii})+P^0_{ni}(P^0_{vv}-P^0_{iv}+\sqrt{\frac{q}{\alpha}})\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{nv}(P^0_{vi}-P^0_{ii})-P^0_{ni}(P^0_{vv}-P^0_{iv}-\sqrt{\frac{q}{\alpha}})\right\}}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.45}$$

$$\mathbf{P}_{jv}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{jv}P^0_{vi}+P^0_{jv}P^0_{ii}+P^0_{ji}P^0_{vv}-P^0_{ji}P^0_{iv}+P^0_{ji}\sqrt{\frac{q}{\alpha}}\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{jv}P^0_{vi}-P^0_{jv}P^0_{ii}-P^0_{ji}P^0_{vv}+P^0_{ji}P^0_{iv}+P^0_{ji}\sqrt{\frac{q}{\alpha}}\right\}+2\sqrt{\frac{q}{\alpha}}(P^0_{jv}-P^0_{ji})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.46}$$

$$\mathbf{P}_{ji}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{jv}(P^0_{vi}-P^0_{ii})+P^0_{ji}(P^0_{vv}-P^0_{iv}+\sqrt{\frac{q}{\alpha}})\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{jv}(P^0_{vi}-P^0_{ii})-P^0_{ji}(P^0_{vv}-P^0_{iv}-\sqrt{\frac{q}{\alpha}})\right\}}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.47}$$

$$\mathbf{P}_{jj}(t) = \frac{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}(P^0_{ji}-P^0_{jv})^2+\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}(P^0_{ji}-P^0_{jv})^2}{e^{\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]}+P^0_{jj} \tag{3.48}$$

$$\mathbf{P}_{jn}(t) = \frac{(P^0_{ji}-P^0_{jv})\left[\sqrt{\frac{\alpha}{q}}(P^0_{vn}-P^0_{in})e^{\sqrt{q\alpha}t}-\sqrt{\frac{\alpha}{q}}(P^0_{vn}-P^0_{in})e^{-\sqrt{q\alpha}t}\right]}{\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right)-\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right)}+P^0_{jn} \tag{3.49}$$

And their limiting values can be written as follows:

$$\mathbf{P}_{nv}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{nv}P^0_{vi}+P^0_{nv}P^0_{ii}+P^0_{ni}P^0_{vv}-P^0_{ni}P^0_{iv}+P^0_{ni}\sqrt{\frac{q}{\alpha}}\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{nv}P^0_{vi}-P^0_{nv}P^0_{ii}-P^0_{ni}P^0_{vv}+P^0_{ni}P^0_{iv}+P^0_{ni}\sqrt{\frac{q}{\alpha}}\right\}+2\sqrt{\frac{q}{\alpha}}(P^0_{nv}-P^0_{ni})}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.50}$$

$$\mathbf{P}_{ni}(t) = \frac{e^{\sqrt{q\alpha}t}\left\{-P^0_{nv}(P^0_{vi}-P^0_{ii})+P^0_{ni}(P^0_{vv}-P^0_{iv}+\sqrt{\frac{q}{\alpha}})\right\}+e^{-\sqrt{q\alpha}t}\left\{P^0_{nv}(P^0_{vi}-P^0_{ii})-P^0_{ni}(P^0_{vv}-P^0_{iv}-\sqrt{\frac{q}{\alpha}})\right\}}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \tag{3.51}$$

$$\mathbf{P}_{nj}(t) = \frac{(P^0_{ni}-P^0_{nv})\left[(P^0_{vj}-P^0_{ij})e^{\sqrt{q\alpha}t}-(P^0_{vj}-P^0_{ij})e^{-\sqrt{q\alpha}t}\right]}{e^{\sqrt{q\alpha}t}\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right)-e^{-\sqrt{q\alpha}t}\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right)} + P^0_{nj} \tag{3.52}$$

$$\mathbf{P}_{nn}(t) = \frac{-e^{\sqrt{q\alpha}t}(P^0_{ni}-P^0_{nv})^2+e^{-\sqrt{q\alpha}t}(P^0_{ni}-P^0_{nv})^2}{e^{\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{\frac{q}{\alpha}}\right]-e^{-\sqrt{q\alpha}t}\left[P^0_{vv}-2P^0_{iv}+P^0_{ii}-\sqrt{\frac{q}{\alpha}}\right]} + P^0_{nn} \tag{3.53}$$

$$\lim_{t\to\infty} P_{vi}(t) = \frac{\left(P_{ii}^0 P_{vv}^0 - P_{iv}^0 P_{vi}^0 + P_{ii}^0\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.54}$$

$$\lim_{t\to\infty} P_{vj}(t) = \frac{\left\{-P_{jv}^0 P_{vi}^0 + P_{jv}^0 P_{ii}^0 + P_{ji}^0 P_{vv}^0 - P_{ji}^0 P_{iv}^0 + P_{ji}^0\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.55}$$

$$\lim_{t\to\infty} P_{vn}(t) = \frac{\left\{-P_{nv}^0 P_{vi}^0 + P_{nv}^0 P_{ii}^0 + P_{ni}^0 P_{vv}^0 - P_{ni}^0 P_{iv}^0 + P_{ni}^0\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.56}$$

$$\lim_{t\to\infty} P_{iv}(t) = \frac{\left(P_{ii}^0 P_{vv}^0 - P_{iv}^0 P_{vi}^0 + P_{ii}^0\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.57}$$

$$\lim_{t\to\infty} P_{ii}(t) = \frac{\left[P_{vv}^0 P_{ii}^0 - {P_{iv}^0}^2 + P_{ii}^0\sqrt{\frac{q}{\alpha}})\right]}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.58}$$

$$\lim_{t\to\infty} P_{ij}(t) = \frac{\left\{-P_{jv}^0(P_{vi}^0 - P_{ii}^0) + P_{ji}^0(P_{vv}^0 - P_{iv}^0 + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.59}$$

$$\lim_{t\to\infty} P_{in}(t) = \frac{\left\{-P_{nv}^0(P_{vi}^0 - P_{ii}^0) + P_{ni}^0(P_{vv}^0 - P_{iv}^0 + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.60}$$

$$\lim_{t\to\infty} P_{jv}(t) = \frac{\left\{-P_{jv}^0 P_{vi}^0 + P_{jv}^0 P_{ii}^0 + P_{ji}^0 P_{vv}^0 - P_{ji}^0 P_{iv}^0 + P_{ji}^0\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.61}$$

$$\lim_{t\to\infty} P_{ji}(t) = \frac{\left\{-P_{jv}^0(P_{vi}^0 - P_{ii}^0) + P_{ji}^0(P_{vv}^0 - P_{iv}^0 + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.62}$$

$$\lim_{t\to\infty} P_{jj}(t) = \frac{-(P_{ji}^0 - P_{jv}^0)^2}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} + P_{jj}^0 \tag{3.63}$$

$$\lim_{t\to\infty} P_{jn}(t) = \frac{(P_{ji}^0 - P_{jv}^0)(P_{vn}^0 - P_{in}^0)}{\left(P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right)} + P_{jn}^0 \tag{3.64}$$

$$\lim_{t\to\infty} P_{nv}(t) = \frac{\left\{-P^0_{nv}P^0_{vi} + P^0_{nv}P^0_{ii} + P^0_{ni}P^0_{vv} - P^0_{ni}P^0_{iv} + P^0_{ni}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P^0_{vv} - 2P^0_{iv} + P^0_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.65}$$

$$\lim_{t\to\infty} P_{ni}(t) = \frac{\left\{-P^0_{nv}(P^0_{vi} - P^0_{ii}) + P^0_{ni}(P^0_{vv} - P^0_{iv} + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P^0_{vv} - 2P^0_{iv} + P^0_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{3.66}$$

$$\lim_{t\to\infty} P_{jn}(t) = \frac{(P^0_{ni} - P^0_{nv})(P^0_{vj} - P^0_{ij})}{\left(P^0_{vv} - 2P^0_{iv} + P^0_{ii} + \sqrt{\frac{q}{\alpha}}\right)} + P^0_{nj} \tag{3.67}$$

$$\lim_{t\to\infty} P_{nn}(t) = \frac{-(P^0_{ni} - P^0_{nv})^2}{\left[P^0_{vv} - 2P^0_{iv} + P^0_{ii} + \sqrt{\frac{q}{\alpha}}\right]} + P^0_{nn} \tag{3.68}$$

b. Observation of two features simultaneously

To solve the case in which the robot observes two feature simultaneously, the observation matrix $\mathbf{H}$ is defined to be

$$\mathbf{H} = \begin{bmatrix} -1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix} . \tag{3.69}$$

The indices i and j are assigned to the observed features and the index n represents the unobserved feature. The system noise variance q is identical with the case of one observed feature, but the parameter α is now defined as $\sqrt{q\left(\frac{1}{r_i} + \frac{1}{r_j}\right)}$.

- $P_{vv}(t)$

$$
\begin{aligned}
P_{vv}(t) = {} & \sinh(\alpha t)\frac{q}{\alpha} + \frac{t\sinh(\alpha t)}{(r_i+r_j)}\frac{q}{\alpha}\left(P^0_{ii} - 2P^0_{ij} + P^0_{jj}\right) \\
& + \cosh(\alpha t)P^0_{vv} \\
& + \frac{t\cosh(\alpha t)}{(r_i+r_j)}\left\{P^0_{ii}P^0_{vv} + P^0_{vv}P^0_{jj} + 2P^0_{vj}P^0_{vi} - 2P^0_{ij}P^0_{vv} - {P^0_{vi}}^2 - {P^0_{vj}}^2\right\} \\
& - \frac{\left(1-e^{\alpha t}\right)\left(1-e^{-\alpha t}\right)}{(r_i+r_j)^2}\left\{\begin{array}{c}(r_i+r_j)t\left[-{P^0_{ij}}^2+P^0_{ij}P^0_{vi}-P^0_{vj}P^0_{ii}+P^0_{vj}P^0_{ij}+P^0_{jj}P^0_{ii}-P^0_{jj}P^0_{vi}\right] \\ +r_i^2\left[P^0_{jj}-P^0_{vj}\right]+r_j^2\left[P^0_{ii}-P^0_{vi}\right]+r_ir_j(2P^0_{ij}-P^0_{vi}-P^0_{vj})\end{array}\right\} \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q}\left\{\begin{array}{c}t(r_i+r_j)\left\{{P^0_{ij}}^2P^0_{vv}+{P^0_{vj}}^2P^0_{ii}+{P^0_{vi}}^2P^0_{jj}-2P^0_{vi}P^0_{vj}P^0_{ij}-P^0_{jj}P^0_{vv}P^0_{ii}\right\} \\ +2r_ir_j\left\{P^0_{vi}P^0_{vj}-P^0_{ij}P^0_{vv}\right\}+r_i^2\left({P^0_{vj}}^2-P^0_{jj}P^0_{vv}\right)+r_j^2\left({P^0_{vi}}^2-P^0_{ii}P^0_{vv}\right)\end{array}\right\}
\end{aligned}
\tag{3.70}
$$

- $P_{vi}(t)$

$$
\begin{aligned}
P_{vi}(t) = {} & P^0_{vi} + \frac{e^{-\alpha t}}{2(r_i+r_j)}\left(\begin{array}{c}(1-e^{\alpha t})^2\left\{P^0_{ii}r_j + P^0_{ij}r_i + (P^0_{ii}P^0_{jj} - {P^0_{ij}}^2)t\right\} \\ +2te^{\alpha t}(+P^0_{vi}P^0_{jj} + P^0_{ii}P^0_{vj} - P^0_{ij}P^0_{vi} - P^0_{ij}P^0_{vj})\end{array}\right) \\
& - \frac{(e^{\alpha t}-e^{-\alpha t})}{2(r_i+r_j)^2}\frac{\alpha}{q}\left(\begin{array}{c}r_i^2\left(P^0_{vi}P^0_{vj}+P^0_{ij}P^0_{vj}-P^0_{jj}P^0_{vi}-P^0_{ij}P^0_{vv}\right) \\ +r_j^2\left({P^0_{vi}}^2-P^0_{ii}P^0_{vv}\right) \\ +r_ir_j\left({P^0_{vi}}^2+P^0_{ii}P^0_{vj}+P^0_{vi}P^0_{vj}-P^0_{ii}P^0_{vv}-P^0_{ij}P^0_{vi}-P^0_{ij}P^0_{vv}\right) \\ +t(r_i+r_j)\left(P^0_{ii}{P^0_{vj}}^2+{P^0_{ij}}^2P^0_{vv}+P^0_{jj}{P^0_{vi}}^2-P^0_{jj}P^0_{ii}P^0_{vv}-2P^0_{ij}P^0_{vj}P^0_{vi}\right)\end{array}\right)
\end{aligned}
\tag{3.71}
$$

- $P_{vj}(t)$

$$
\begin{aligned}
P_{vj}(t) = {} & P^0_{vj} + \frac{e^{-\alpha t}}{2(r_i+r_j)}\left(\begin{array}{c}(1-e^{\alpha t})^2\left\{P^0_{ij}r_j + P^0_{jj}r_i + (P^0_{ii}P^0_{jj} - {P^0_{ij}}^2)t\right\} \\ +2te^{\alpha t}\left\{P^0_{vi}P^0_{jj} + P^0_{vj}P^0_{ii} - P^0_{vj}P^0_{ij} - P^0_{ij}P^0_{vi}\right\}\end{array}\right) \\
& - \frac{(e^{\alpha t}-e^{-\alpha t})}{2(r_i+r_j)^2}\frac{\alpha}{q}\left(\begin{array}{c}+r_i^2\left\{{P^0_{vj}}^2-P^0_{jj}P^0_{vv}\right\} \\ +r_j^2\left\{P^0_{ij}P^0_{vi}+P^0_{vj}P^0_{vi}-P^0_{vj}P^0_{ii}-P^0_{ij}P^0_{vv}\right\} \\ +r_ir_j\left\{{P^0_{vj}}^2+P^0_{jj}P^0_{vi}+P^0_{vj}P^0_{vi}-P^0_{ij}P^0_{vj}-P^0_{jj}P^0_{vv}-P^0_{ij}P^0_{vv}\right\} \\ +t(r_i+r_j)\left\{P^0_{ii}{P^0_{vj}}^2+{P^0_{ij}}^2P^0_{vv}+P^0_{jj}{P^0_{vi}}^2-2P^0_{ij}P^0_{vj}P^0_{vi}-P^0_{ii}P^0_{jj}P^0_{vv}\right\}\end{array}\right)
\end{aligned}
\tag{3.72}
$$

- $P_{vn}(t)$

$$
\begin{aligned}
P_{vn}(t) = & P^0_{vn} \\
& + \frac{e^{-\alpha t}}{2(r_i+r_j)}(1-e^{\alpha t})^2 \left\{ P^0_{in} r_j + P^0_{jn} r_i + t\left(P^0_{in} P^0_{jj} + P^0_{ii} P^0_{jn} - P^0_{ij} P^0_{jn} - P^0_{in} P^0_{ij} \right) \right\} \\
& + \frac{t}{(r_i+r_j)} \left\{ P^0_{vi} P^0_{jn} + P^0_{vj} P^0_{in} + P^0_{vn} P^0_{ii} + P^0_{vn} P^0_{jj} - 2P^0_{vn} P^0_{ij} - P^0_{vj} P^0_{jn} - P^0_{vi} P^0_{in} \right\} \\
& - \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i+r_j)^2} \frac{\alpha}{q} \left(\begin{array}{c}
+r_i^2 \left\{ P^0_{jn} P^0_{vj} + P^0_{vj} P^0_{vn} - P^0_{jj} P^0_{vn} - P^0_{jn} P^0_{vv} \right\} \\
+r_j^2 \left\{ P^0_{in} P^0_{vi} + P^0_{vi} P^0_{vn} - P^0_{ii} P^0_{vn} - P^0_{in} P^0_{vv} \right\} \\
+r_i r_j \left\{ P^0_{in} P^0_{vj} + P^0_{jn} P^0_{vi} + P^0_{vi} P^0_{vn} + P^0_{vj} P^0_{vn} - P^0_{in} P^0_{vv} - P^0_{jn} P^0_{vv} - 2P^0_{ij} P^0_{vn} \right\} \\
+t(r_i+r_j) \left\{ \begin{array}{c}
+P^0_{jn} {P^0_{vi}}^2 + P^0_{jn} P^0_{vj} P^0_{ii} + P^0_{jn} P^0_{vv} P^0_{ij} \\
+P^0_{in} P^0_{jj} P^0_{vi} + P^0_{in} {P^0_{vj}}^2 + {P^0_{ij}}^2 P^0_{vn} + P^0_{jj} P^0_{vi} P^0_{vn} \\
+P^0_{ii} P^0_{vj} P^0_{vn} + P^0_{ij} P^0_{in} P^0_{vv} - P^0_{ij} P^0_{vj} P^0_{vn} - P^0_{in} P^0_{jj} P^0_{vv} \\
-P^0_{ij} P^0_{in} P^0_{vj} - P^0_{ii} P^0_{jj} P^0_{vn} - P^0_{ij} P^0_{vi} P^0_{vn} - P^0_{in} P^0_{vi} P^0_{vj} \\
-P^0_{jn} P^0_{ij} P^0_{vi} - P^0_{jn} P^0_{vj} P^0_{vi} - P^0_{jn} P^0_{vv} P^0_{ii}
\end{array} \right\}
\end{array} \right)
\end{aligned}
\tag{3.73}
$$

- $P_{iv}(t) = P_{vi}(t)$
- $P_{ii}(t)$

$$
\begin{aligned}
P_{ii}(t) = & \cosh(\alpha t) P^0_{ii} \\
& + \frac{\cosh(\alpha t)}{(r_i+r_j)} t \left(P^0_{ii} P^0_{jj} - {P^0_{ij}}^2 \right) \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2} \frac{\alpha}{q} \left\{ \begin{array}{c}
t(r_i+r_j) \left\{ {P^0_{vi}}^2 P^0_{jj} - 2P^0_{ij} P^0_{vi} P^0_{vj} + {P^0_{ij}}^2 P^0_{vv} + P^0_{ii} {P^0_{vj}}^2 - P^0_{ii} P^0_{jj} P^0_{vv} \right\} \\
+(r_i+r_j)^2 \left({P^0_{vi}}^2 - P^0_{ii} P^0_{vv} \right) + 2r_i(r_i+r_j) \left(P^0_{ii} P^0_{vj} - P^0_{ij} P^0_{vi} \right) \\
+r_i^2 \left({P^0_{ij}}^2 - P^0_{ii} P^0_{jj} \right)
\end{array} \right\}
\end{aligned}
\tag{3.74}
$$

- $P_{ij}(t)$

$$
\begin{aligned}
P_{ij}(t) &= \cosh(\alpha t) P^0_{ij} \\
&\quad + \frac{\cosh(\alpha t)}{(r_i + r_j)} t \left(P^0_{ii} P^0_{jj} - {P^0_{ij}}^2 \right) \\
&\quad - \frac{\sinh(\alpha t)}{(r_i + r_j)^2} \frac{\alpha}{q} \left\{ \begin{array}{c} t(r_i + r_j) \left[P^0_{vv} {P^0_{ij}}^2 + P^0_{ii} {P^0_{vj}}^2 + P^0_{jj} {P^0_{vi}}^2 - 2 P^0_{ij} P^0_{vi} P^0_{vj} - P^0_{vv} P^0_{ii} P^0_{jj} \right] \\ + (r_i + r_j)^2 \left[P^0_{vi} P^0_{vj} - P^0_{vv} P^0_{ij} \right] + r_i r_j (P^0_{jj} P^0_{ii} - {P^0_{ij}}^2) \\ + (r_i + r_j) \left[r_i (P^0_{ij} P^0_{vj} - P^0_{jj} P^0_{vi}) + r_j (P^0_{ij} P^0_{vi} - P^0_{vj} P^0_{ii}) \right] \end{array} \right\}
\end{aligned}
\tag{3.75}
$$

- $P_{in}(t)$

$$
\begin{aligned}
P_{in}(t) &= \frac{(e^{\alpha t} + e^{-\alpha t})}{2} P^0_{in} \\
&\quad + \frac{(e^{\alpha t} + e^{-\alpha t})}{2(r_i + r_j)} t \left(P^0_{ii} P^0_{jn} + P^0_{in} P^0_{jj} - P^0_{ij} P^0_{jn} - P^0_{in} P^0_{ij} \right) \\
&\quad + \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i + r_j)^2} \frac{\alpha}{q} \left\{ \begin{array}{c} r_i^2 \left\{ P^0_{jn} P^0_{vi} + P^0_{in} P^0_{vv} + P^0_{in} P^0_{jj} + P^0_{ij} P^0_{vn} - P^0_{vi} P^0_{vn} - 2 P^0_{in} P^0_{vj} - P^0_{ij} P^0_{jn} \right\} \\ + r_j^2 \left\{ P^0_{ii} P^0_{vn} + P^0_{in} P^0_{vv} - P^0_{vi} P^0_{vn} - P^0_{in} P^0_{vi} \right\} \\ + r_i r_j \left\{ \begin{array}{c} P^0_{jn} P^0_{vi} + P^0_{ii} P^0_{vn} + P^0_{in} P^0_{ij} + 2 P^0_{in} P^0_{vv} \\ + P^0_{ij} P^0_{vn} - 2 P^0_{vi} P^0_{vn} - P^0_{ii} P^0_{jn} - P^0_{in} P^0_{vi} - 2 P^0_{in} P^0_{vj} \end{array} \right\} \\ + t(r_i + r_j) \left\{ \begin{array}{c} P^0_{in} P^0_{vj} P^0_{ij} + P^0_{in} P^0_{vj} P^0_{vi} + P^0_{in} P^0_{jj} P^0_{vv} + P^0_{ij} P^0_{vn} P^0_{vi} + P^0_{ij} P^0_{vn} P^0_{vj} + P^0_{jn} P^0_{vi} P^0_{vj} \\ + P^0_{ii} P^0_{jn} P^0_{vv} + P^0_{ij} P^0_{jn} P^0_{vi} + P^0_{ii} P^0_{jj} P^0_{vn} - P^0_{jn} {P^0_{vi}}^2 - P^0_{ii} P^0_{jn} P^0_{vj} - P^0_{jj} P^0_{vi} P^0_{vn} \\ - P^0_{ii} P^0_{vj} P^0_{vn} - {P^0_{ij}}^2 P^0_{vn} - P^0_{in} P^0_{ij} P^0_{vv} - P^0_{in} {P^0_{vj}}^2 - P^0_{in} P^0_{jj} P^0_{vi} - P^0_{ij} P^0_{jn} P^0_{vv} \end{array} \right\} \end{array} \right\}
\end{aligned}
\tag{3.76}
$$

- $P_{jv}(t) = P_{vj}(t), \qquad P_{ji}(t) = P_{ij}(t)$
- $P_{jj}(t)$

$$
\begin{aligned}
P_{jj}(t) &= \cosh(\alpha t) P^0_{jj} \\
&\quad + \frac{\cosh(\alpha t)}{(r_i + r_j)} t \left(P^0_{ii} P^0_{jj} - {P^0_{ij}}^2 \right) \\
&\quad - \frac{\sinh(\alpha t)}{(r_i + r_j)^2} \frac{\alpha}{q} \left\{ \begin{array}{c} t(r_i + r_j) \left\{ {P^0_{vj}}^2 P^0_{ii} - 2 P^0_{ij} P^0_{vj} P^0_{vi} + {P^0_{ij}}^2 P^0_{vv} + P^0_{jj} {P^0_{vi}}^2 - P^0_{vv} P^0_{ii} P^0_{jj} \right\} \\ (r_i + r_j)^2 \left({P^0_{vj}}^2 - P^0_{jj} P^0_{vv} \right) + 2 r_j (r_i + r_j) \left(P^0_{jj} P^0_{vi} - P^0_{ij} P^0_{vj} \right) + r_j^2 \left({P^0_{ij}}^2 - P^0_{jj} P^0_{ii} \right) \end{array} \right\}
\end{aligned}
\tag{3.77}
$$

- $P_{jn}(t)$

$$
\begin{aligned}
P_{jn}(t) &= \frac{(e^{\alpha t}+e^{-\alpha t})}{2}P^0_{jn} \\
&+ \frac{(e^{\alpha t}+e^{-\alpha t})}{2(r_i+r_j)}t\left\{P^0_{in}P^0_{jj}+P^0_{jn}P^0_{ii}-P^0_{in}P^0_{ij}-P^0_{jn}P^0_{ij}\right\} \\
&- \frac{(e^{\alpha t}-e^{-\alpha t})}{2(r_i+r_j)^2}\frac{\alpha}{q}\left\{\begin{array}{c}
r_i^2\left\{P^0_{vj}P^0_{vn}+P^0_{jn}P^0_{vj}-P^0_{jn}P^0_{vv}-P^0_{jj}P^0_{vn}\right\} \\
+r_j^2\left\{P^0_{ij}P^0_{in}+P^0_{vj}P^0_{vn}+2P^0_{jn}P^0_{vi}-P^0_{in}P^0_{vj}-P^0_{ij}P^0_{vn}-P^0_{jn}P^0_{vv}-P^0_{jn}P^0_{ii}\right\} \\
+r_ir_j\left\{\begin{array}{c}2P^0_{vj}P^0_{vn}-2P^0_{jn}P^0_{vv}+2P^0_{jn}P^0_{vi}+P^0_{jn}P^0_{vj}+P^0_{jj}P^0_{in} \\ -P^0_{in}P^0_{vj}-P^0_{ij}P^0_{vn}-P^0_{jn}P^0_{ij}-P^0_{jj}P^0_{vn}\end{array}\right\} \\
+t(r_i+r_j)\left\{\begin{array}{c}P^0_{in}{P^0_{vj}}^2+{P^0_{ij}}^2P^0_{vn}+P^0_{ii}P^0_{vj}P^0_{vn}+P^0_{jn}{P^0_{vi}}^2+P^0_{jn}P^0_{vv}P^0_{ij} \\ +P^0_{jn}P^0_{vj}P^0_{ii}+P^0_{jj}P^0_{vn}P^0_{vi}+P^0_{jj}P^0_{in}P^0_{vi}+P^0_{ij}P^0_{in}P^0_{vv}-P^0_{ij}P^0_{in}P^0_{vj} \\ -P^0_{in}P^0_{vi}P^0_{vj}-P^0_{ij}P^0_{vi}P^0_{vn}-P^0_{ij}P^0_{vj}P^0_{vn}-P^0_{jn}P^0_{ij}P^0_{vi}-P^0_{jn}P^0_{vv}P^0_{ii} \\ -P^0_{jn}P^0_{vj}P^0_{vi}-P^0_{jj}P^0_{vn}P^0_{ii}-P^0_{jj}P^0_{in}P^0_{vv}\end{array}\right\}
\end{array}\right\}
\end{aligned}
\tag{3.78}
$$

- $P_{nv}(t) = P_{vn}(t)$

- $P_{ni}(t) = P_{in}(t)$

- $P_{nj}(t) = P_{jn}(t)$

- $P_{nn}(t)$

$$
\begin{aligned}
P_{nn}(t) &= \cosh(\alpha t)P^0_{nn} \\
&+ \frac{\cosh(\alpha t)}{(r_i+r_j)}t\left\{P^0_{nn}P^0_{ii}+P^0_{nn}P^0_{jj}+2P^0_{in}P^0_{jn}-2P^0_{ij}P^0_{nn}-{P^0_{in}}^2-{P^0_{jn}}^2\right\} \\
&- \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q}\left(\begin{array}{c}
r_i^2({P^0_{vn}}^2+{P^0_{jn}}^2+2P^0_{nn}P^0_{vj}-P^0_{nn}P^0_{vv}-P^0_{jj}P^0_{nn}-2P^0_{jn}P^0_{vn}) \\
+r_j^2({P^0_{vn}}^2+{P^0_{in}}^2+2P^0_{nn}P^0_{vi}-P^0_{nn}P^0_{vv}-P^0_{ii}P^0_{nn}-2P^0_{in}P^0_{vn}) \\
+2r_ir_j({P^0_{vn}}^2+P^0_{nn}P^0_{vi}+P^0_{nn}P^0_{vj}+P^0_{in}P^0_{jn}-P^0_{ij}P^0_{nn}-P^0_{nn}P^0_{vv}-P^0_{in}P^0_{vn}-P^0_{jn}P^0_{vn}) \\
+t(r_i+r_j)\left\{\begin{array}{c}{P^0_{ij}}^2P^0_{nn}-2P^0_{ij}P^0_{nn}P^0_{vi}+P^0_{nn}{P^0_{vi}}^2-2{P^0_{in}}^2P^0_{vj} \\ +2P^0_{ii}P^0_{nn}P^0_{vj}-2P^0_{ij}P^0_{nn}P^0_{vj}-2P^0_{nn}P^0_{vi}P^0_{vj}+P^0_{nn}{P^0_{vj}}^2 \\ +2P^0_{ij}P^0_{in}P^0_{vn}-2P^0_{in}P^0_{vi}P^0_{vn}+2P^0_{in}P^0_{vj}P^0_{vn}+P^0_{ii}{P^0_{vn}}^2 \\ -2P^0_{ij}{P^0_{vn}}^2+{P^0_{in}}^2P^0_{vv}-P^0_{ii}P^0_{nn}P^0_{vv}+2P^0_{ij}P^0_{nn}P^0_{vv} \\ +P^0_{jn}{}^2P^0_{ii}-2P^0_{jn}{}^2P^0_{vi}+P^0_{jn}{}^2P^0_{vv}+P^0_{jj}{P^0_{in}}^2 \\ -2P^0_{jj}P^0_{in}P^0_{vn}+P^0_{jj}{P^0_{vn}}^2-P^0_{jj}P^0_{nn}P^0_{ii}+2P^0_{jj}P^0_{nn}P^0_{vi}-P^0_{jj}P^0_{nn}P^0_{vv} \\ -2P^0_{jn}P^0_{vn}P^0_{ii}+2P^0_{jn}P^0_{vn}P^0_{ij}+2P^0_{jn}P^0_{vn}P^0_{vi}-2P^0_{jn}P^0_{vn}P^0_{vj} \\ -2P^0_{jn}P^0_{in}P^0_{ij}+2P^0_{jn}P^0_{in}P^0_{vi}+2P^0_{jn}P^0_{in}P^0_{vj}-2P^0_{jn}P^0_{in}P^0_{vv}\end{array}\right\}
\end{array}\right)
\end{aligned}
\tag{3.79}
$$

3.3.2 Analysis of the closed form solution

In this subsection, the properties of the covariance matrix are investigated through the closed form solution summarized in the previous sections. Though the given problem is a continuous case, it suffices to show the behaviors of the covariance matrix [34].

a. Monotonicity

Let us examine the monotonicity of the determinant feature covariance block. As described in the previous section, it has been proven that the uncertainty of a feature block monotonically decreases during a SLAM mission. This property is true for every feature in the map whether the feature is observed or unobserved. Equations 3.80 and 3.81 illustrate the monotonicity of these two cases: (1) an observed feature and (2) an unobserved feature.

First, the covariance for the observed feature, specified by

$$\begin{aligned}\mathbf{P}_{ii}(t) &= \frac{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 P_{ii}^0 - {P_{iv}^0}^2 + P_{ii}^0\sqrt{qr_i}\right] + e^{-\sqrt{\frac{q}{r_i}}t}\left[{P_{iv}^0}^2 - P_{vv}^0 P_{ii}^0 + P_{ii}^0\sqrt{qr_i}\right]}{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right] - e^{-\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{qr_i}\right]} \\ &= P_{ii}^0 - \frac{(P_{iv}^0 - P_{ii}^0)^2\left(e^{\sqrt{\frac{q}{r_i}}t} - e^{-\sqrt{\frac{q}{r_i}}t}\right)}{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right] - e^{-\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{qr_i}\right]},\end{aligned} \tag{3.80}$$

clearly monotonically decreases with time, because its first derivative is always non-positive. Thus, the determinant of the observed feature block decreases monotonically from the initial uncertainty P_{ii}^0.

Similarly, the solution for the unobserved feature (feature n) illustrates the monotonically decreasing property through its first derivative with time

$$\mathbf{P}_{nn}(t) = P_{nn}^0 - \frac{(P_{ni}^0 - P_{nv}^0)^2\left(e^{\sqrt{\frac{q}{r_i}}t} - e^{-\sqrt{\frac{q}{r_i}}t}\right)}{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right] - e^{-\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{qr_i}\right]} \tag{3.81}$$

One interesting fact is that the amount of the decrement for the unobserved feature does not depend on its initial uncertainty, while the amount of the observed feature does.

b. Terminal uncertainty

So far, the non-zero terminal uncertainty has been a property not well known to the SLAM research community. Many papers have assumed that the terminal form of the covariance matrix is fully correlated (the absolute value of every cross correlation becomes 1.) without specifying a rigorous condition for this assumption. This has possibly misled the SLAM research community to conclude that if the survey is executed forever, the solution will become perfect even when a part of the features are observable only for a finite time duration.

In this section, the terminal uncertainty for the following cases are examined through the closed form solutions: (1) only one feature is observed and (2) two features are observed simultaneously.

First, if only one feature (feature i) is observed, the feature covariance is

$$\begin{aligned}\mathbf{P}_{ii}(t) &= \frac{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 P_{ii}^0 - {P_{iv}^0}^2 + P_{ii}^0\sqrt{qr_i})\right] + e^{-\sqrt{\frac{q}{r_i}}t}\left[{P_{iv}^0}^2 - P_{vv}^0 P_{ii}^0 + P_{ii}^0\sqrt{qr_i})\right]}{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right] - e^{-\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{qr_i}\right]} \\ &= P_{ii}^0 - \frac{(P_{iv}^0 - P_{ii}^0)^2\left(e^{\sqrt{\frac{q}{r_i}}t} - e^{-\sqrt{\frac{q}{r_i}}t}\right)}{e^{\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right] - e^{-\sqrt{\frac{q}{r_i}}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{qr_i}\right]}\end{aligned} \tag{3.82}$$

and its limiting form

$$\lim_{t\to\infty} P_{ii}(t) = \frac{\left[P_{vv}^0 P_{ii}^0 - {P_{iv}^0}^2 + P_{ii}^0\sqrt{qr_i})\right]}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{qr_i}\right]} \tag{3.83}$$

shows the non-zero limit value. This limit is the uncertainty of the observed feature when t goes to infinity. The cross correlation of the block of an observed feature (feature i) and an unobserved feature (feature n), $\begin{bmatrix} P_{ii} & P_{in} \\ P_{ni} & P_{nn} \end{bmatrix}$, does not converge to one, either. It can be easily checked from the limiting value:

$$\lim_{t\to\infty} \frac{P_{ni}(t)^2}{P_{ii}(t)\cdot P_{nn}(t)}$$
$$= \frac{\left\{-P^0_{nv}(P^0_{vi}-P^0_{ii})+P^0_{ni}(P^0_{vv}-P^0_{iv}+\sqrt{qr_i})\right\}^2}{\left[P^0_{vv}P^0_{ii}-P^{0\,2}_{iv}+P^0_{ii}\sqrt{qr_i})\right]\left[-(P^0_{ni}-P^0_{nv})^2+P^0_{nn}\left(P^0_{vv}-2P^0_{iv}+P^0_{ii}+\sqrt{qr_i}\right)\right]}$$
$$\neq 1 \quad . \tag{3.84}$$

When two features (here, i and j) are observed simultaneously, the behavior of their limiting values is shown in Figure 3-9. The limiting values are

$$\begin{bmatrix} P_{ii}(\infty) \\ P_{ji}(\infty) \\ P_{ni}(\infty) \end{bmatrix} = \begin{bmatrix} P_{ij}(\infty) \\ P_{jj}(\infty) \\ P_{nj}(\infty) \end{bmatrix} \quad , \tag{3.85}$$

and hence, the matrix is singular. Consequently, the determinant of the whole feature block converges to zero, even though a block associated with an unobserved feature remains non-zero.

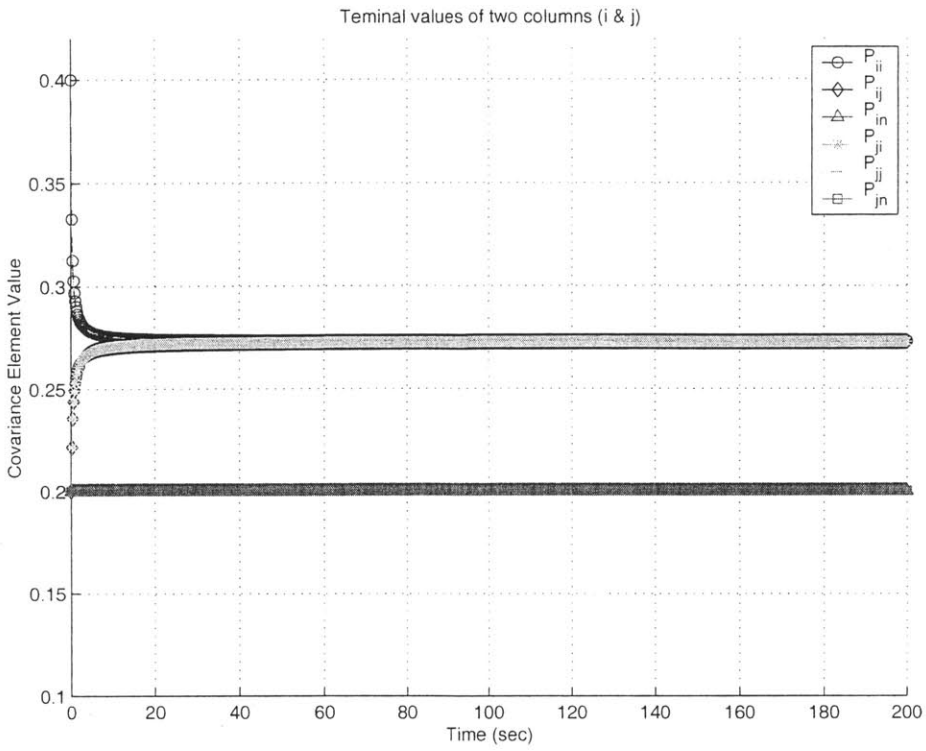

Figure 3-9: If two features (i and j) of the same type are simultaneously observed, their terminal values become $P_{ii}(\infty) = P_{ij}(\infty)$, $P_{ji}(\infty) = P_{jj}(\infty)$, and $P_{ni}(\infty) = P_{nj}(\infty)$.

c. Level-wise update

All the closed form solutions illustrate the level-wise update behavior. Every element of the covariance matrix follows the level-wise update rule described in the previous subsection. For example,

$$\mathbf{P}_{ii}(t) = \frac{e^{\sqrt{q\alpha}t}\left[P_{vv}^0 P_{ii}^0 - P_{iv}^{0\,2} + P_{ii}^0\sqrt{\frac{q}{\alpha}})\right] + e^{-\sqrt{q\alpha}t}\left[P_{iv}^{0\,2} - P_{vv}^0 P_{ii}^0 + P_{ii}^0\sqrt{\frac{q}{\alpha}})\right]}{e^{\sqrt{q\alpha}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 - \sqrt{\frac{q}{\alpha}}\right]} \tag{3.86}$$

shows that the update of an observed feature (P_{ii}) requires only the previous values of P_{ii}, P_{vi}, and P_{vv}.

d. Lower bound of the uncertainty

The limiting forms of the feature in the covariance matrix capture the lower bounds of the feature uncertainty. Suppose, for example, that only one feature exists in the region being mapped. When it is initialized, the covariance matrix is:

$$\begin{bmatrix} \mathbf{P}_{vv} & (\mathbf{G}_x\mathbf{P}_x)^T \\ \mathbf{G}_x\mathbf{P}_x & \mathbf{G}_x\mathbf{P}_x\mathbf{G}_x^T + \mathbf{G}_z r_i \mathbf{G}_z^T \end{bmatrix} . \tag{3.87}$$

Since $\mathbf{G}_x = 1$, $\mathbf{G}_z = 1$, and $\mathbf{P}_x = \mathbf{P}_{vv}$ for our situation, the covariance matrix ends will be

$$\begin{bmatrix} \mathbf{P}_{vv} & \mathbf{P}_{vv}^T \\ \mathbf{P}_{vv} & \mathbf{P}_{vv} + r_i \end{bmatrix} . \tag{3.88}$$

Now, let us substitute these elements into the initial values of Equation 3.83. Then,

$$\begin{aligned} \mathbf{P}_{ii}(\infty) &= \frac{\left[P_{vv}^0 P_{ii}^0 - P_{iv}^{0\,2} + P_{ii}^0\sqrt{\frac{q}{\alpha}})\right]}{\left[P_{vv}^0 - 2P_{iv}^0 + P_{ii}^0 + \sqrt{\frac{q}{\alpha}}\right]} \\ &= \frac{\left[P_{vv}(P_{vv}+r_i) - P_{vv}^2 + (P_{vv}+r_i)\sqrt{qr_i})\right]}{\left[P_{vv} - 2P_{vv} + (P_{vv}+r_i) + \sqrt{qr_i}\right]} \\ &= P_{vv} + \frac{r_i\sqrt{qr_i}}{\left[r_i + \sqrt{qr_i}\right]} \quad . \end{aligned} \tag{3.89}$$

Therefore, the uncertainty converges to the uncertainty of the robot when the robot initializes the first feature when $q = 0$ or $r_i = 0$, which corresponds to the case of no system noise or no measurement noise, respectively. If noise values are non-zero, then the limiting value of the uncertainty is a function of these noise variances.

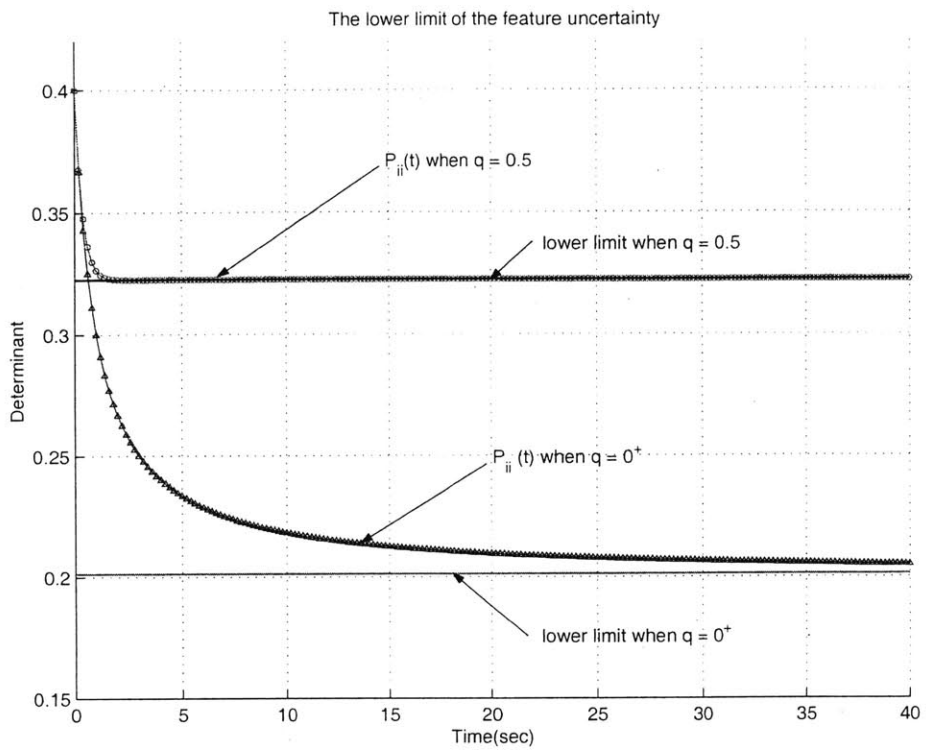

Figure 3-10: The lower bound of the feature uncertainty depends on the noise levels.

3.3.3 Case study

In this subsection, we investigate the solution behavior of several cases of the MonoRob SLAM problem. Through these examples, the described properties in this chapter such as the monotonicity, the non-negative definiteness, and non-zero terminal uncertainty, are illustrated.

As the default configuration, the initial covariance matrix is set to be

$$\mathbf{P}(0) = \begin{bmatrix} \mathbf{P}_{RR}(0) & \mathbf{P}_{Ri}(0) & \mathbf{P}_{Rj}(0) & \mathbf{P}_{Rn}(0) \\ \mathbf{P}_{iR}(0) & \mathbf{P}_{ii}(0) & \mathbf{P}_{ij}(0) & \mathbf{P}_{in}(0) \\ \mathbf{P}_{jR}(0) & \mathbf{P}_{ji}(0) & \mathbf{P}_{jj}(0) & \mathbf{P}_{jn}(0) \\ \mathbf{P}_{nR}(0) & \mathbf{P}_{ni}(0) & \mathbf{P}_{nj}(0) & \mathbf{P}_{nn}(0) \end{bmatrix} = \begin{bmatrix} 0.4 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.4 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.4 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.4 \end{bmatrix}, \tag{3.90}$$

and the noise variances

$$q = 0.5, \qquad r_i = r_j = r_n = 0.2 \tag{3.91}$$

are used [2].

For the purpose of the comparison the following four scenarios are examined:

1. The robot observes the feature i for $t \leq 20$.
   The other features are not observed at all.

2. The robot observes the feature i for $t \leq 10$ and switches observing
   to the feature j for $10 < t < 20$. The feature n is not observed.

3. The robot keeps switching between feature i and feature j every 2 seconds
   until $t = 20$.

4. The robot observes both feature i and feature j simultaneously. This condition is
   maintained until $t = 200$.

In all cases, the monotonicity and non-negativeness of the determinants are easily examined. The results of the first scenario illustrate a case in which the terminal value of the uncertainty is non-zero (Figure 3-11). The determinant converges to its limit very quickly and stops decreasing because useful observations are no longer provided. This behavior is observed in the second scenario, too. The determinant converges to the first limit and stops decreasing as it does in the first case. Once the robot begins observing feature j, the determinant decreases again and converges to a new limiting value. The new convergence limit, however, is not zero even though its value is smaller than the value of the first scenario. This stair-wise decrement is also seen in the third case. Whenever new useful information is provided, the determinant starts decreasing to the next limiting value which is still not zero (Figure 3-15). Although the limiting value is getting smaller, the reductions stop once the robot switches to observe another feature. Also, the cross correlation between an observed feature and an unobserved feature does not converge to 1 as shown in Figure 3-16. When two features are observed simultaneously, as in fourth scenario, the limiting value of the

[2] The initial covariance matrix can be constructed as follows: (1) the robot moves for 0.4 sec without observing any features, (2) the robot initializes three features, and (3) the robot moves for 0.4 sec again without re-observing any features. During the robot's motion, the covariance matrix follows $\hat{\mathbf{P}} = \mathbf{G}\mathbf{Q}\mathbf{G}^T$, according to Equation 3.34.

determinant becomes zero. The cross correlation between unobserved features, however, still does not converge to 1.

- scenario 1

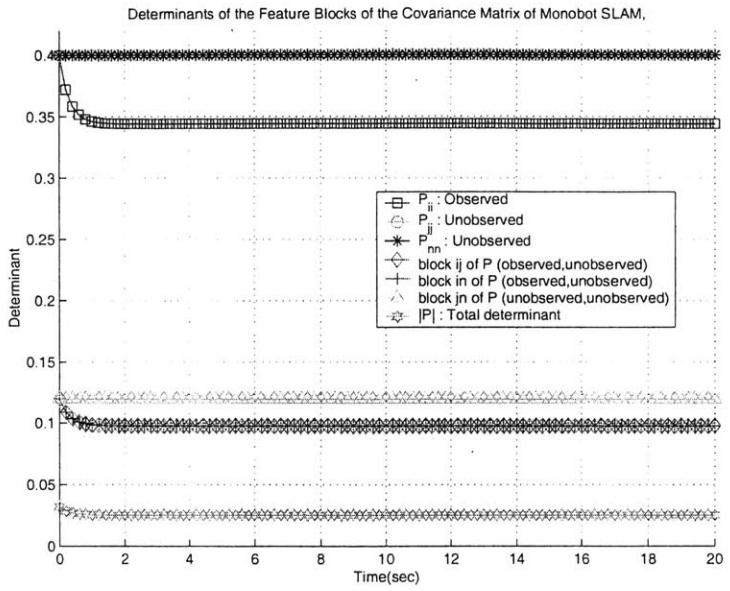

Figure 3-11: The determinants of the feature blocks when only one feature is observed.

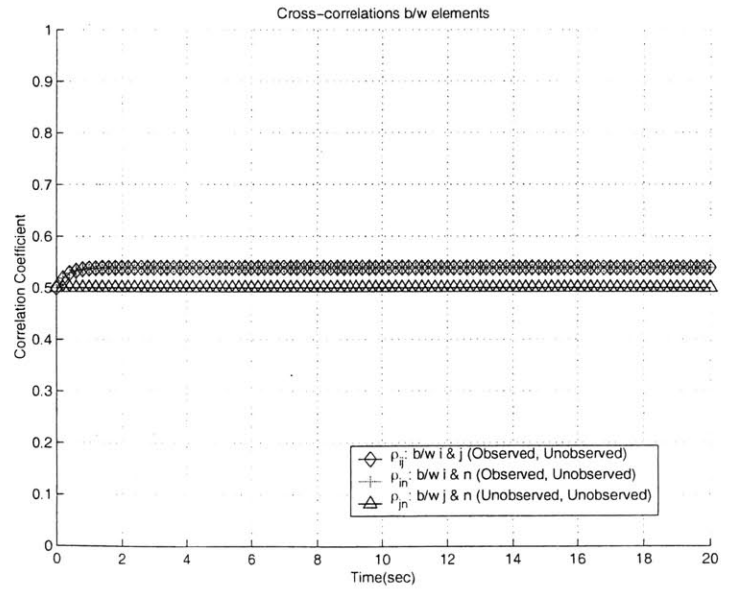

Figure 3-12: The cross correlation behavior when only one feature is observed.

- scenario 2

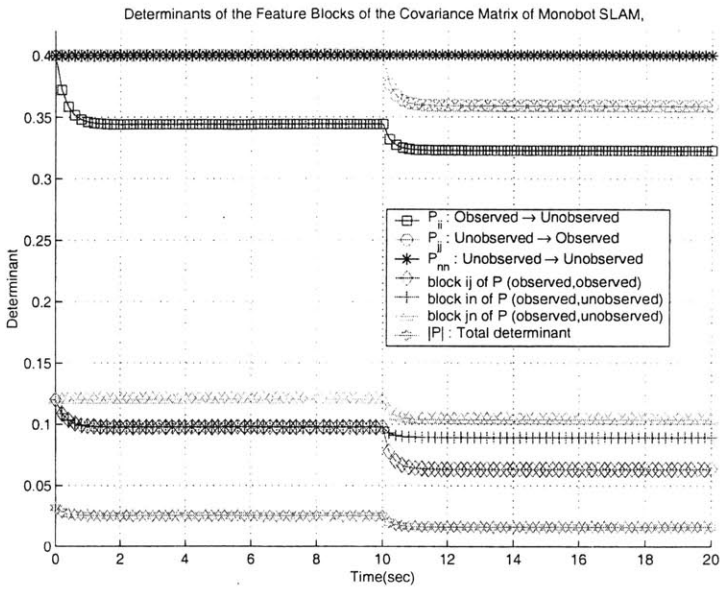

Figure 3-13: The determinants of the feature blocks when the feature observed is switched once.

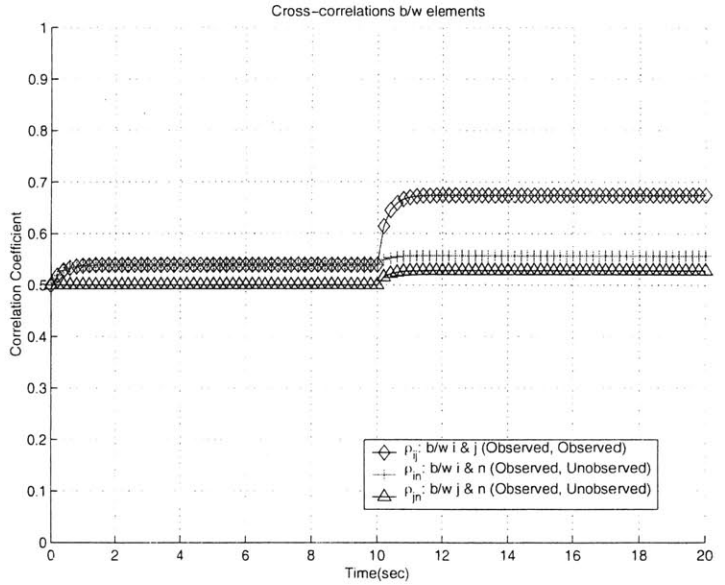

Figure 3-14: The cross correlation behavior when the feature observed is switched once.

- scenario 3

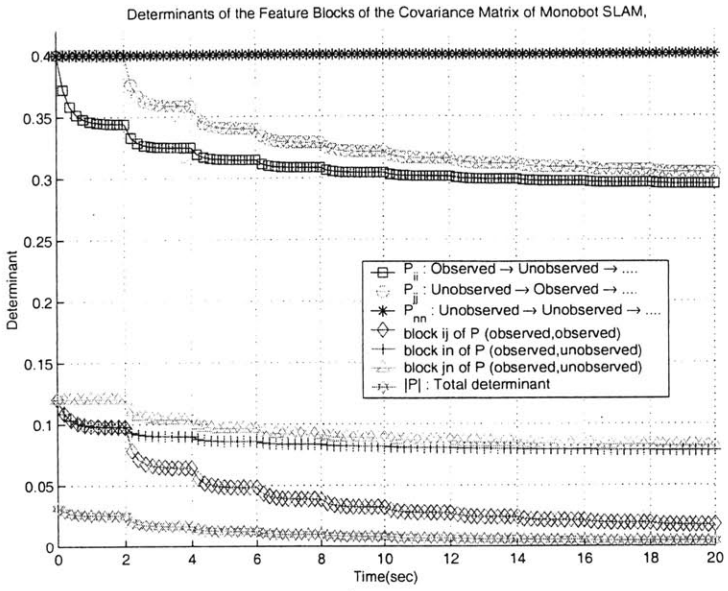

Figure 3-15: The determinants of the feature blocks when the feature observed is switched every two seconds.

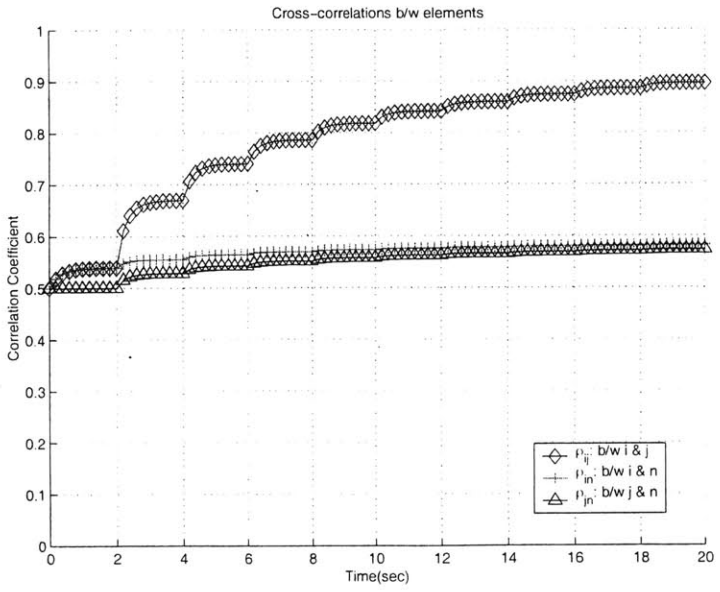

Figure 3-16: The cross correlation behavior when the feature observed is switched every two seconds.

- scenario 4

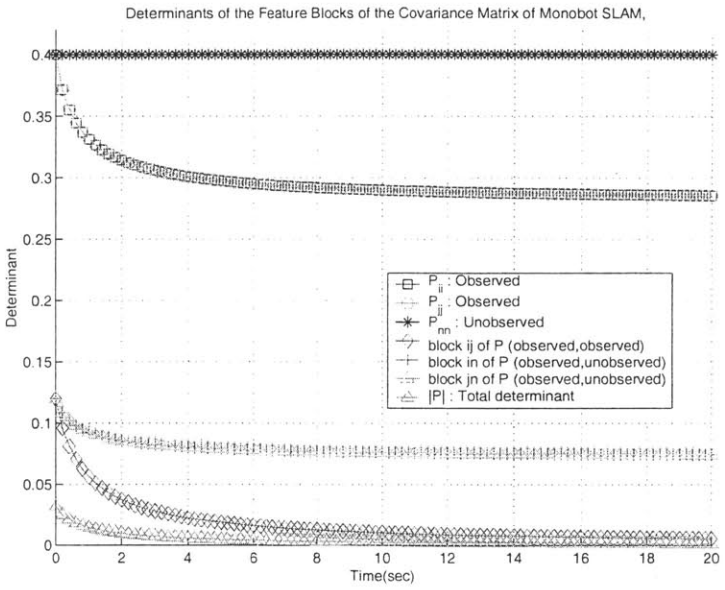

Figure 3-17: The determinants of the feature blocks when two features are simultaneously observed.

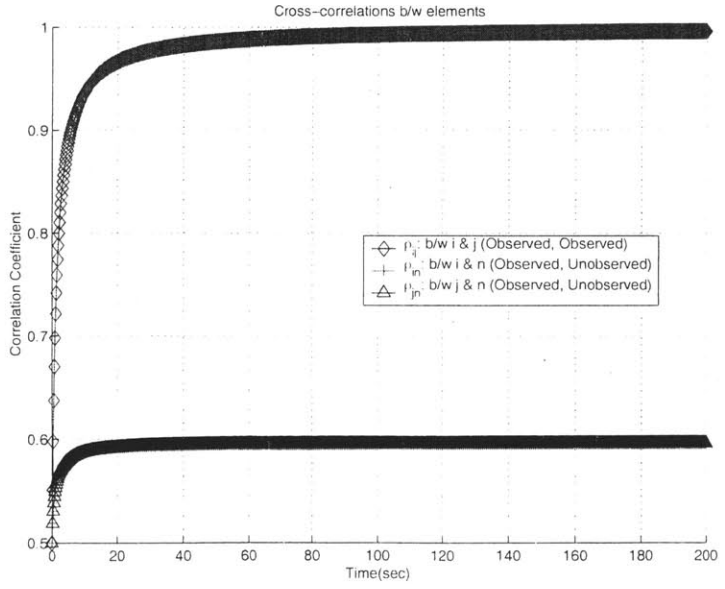

Figure 3-18: The cross correlation behavior when two features are simultaneously observed.

3.4 Conclusion

In Chapter 3, several properties of the covariance matrix have been investigated in detail. Theoretical proofs of the properties have been provided and the time domain behaviors of the covariance matrix have been examined through analytical solution of one dimensional SLAM problem.

The resultant properties include: (1) a non-zero terminal uncertainty, (2) the condition for achieving a fully correlated covariance matrix, (3) exposition of the level-wise update structure in the Kalman filter based SLAM solution, and (4) the uncertainty projection during compounding and root shifting. The first two properties clarify the conditions that a SLAM solution should satisfy. The level-wise update provides clear dependencies between observed features and unobserved features during the Kalman update operation. The properties on the uncertainty projection are effectively used to analyze the Constant Time SLAM (CTS) algorithm and to develop the CTS 2.0 and Network Optimized SLAM (NOS) algorithms in Chapter 4.

In addition to the examination of these properties, the closed form solution for one dimensional a MonoRob SLAM problem has been provided. The given problem considers arbitrary initial values for the covariance matrix and partial observations (not all features observed at all time steps). Most of the properties found in the first part of Chapter 3 have been examined through the closed form solution behaviors. We checked the non-zero terminal uncertainty, cross correlation behavior, level-wise update dependencies, and the achievable lowest bound for the feature uncertainty.

In the next chapter, we will utilize the properties found in this chapter to improve the converging rate of the global uncertainties of local maps and to develop new algorithms that provide better performance than the current state-of-the-art for the scaling problem.

Chapter 4

Efficient SLAM Algorithms Using Submap Networks

In the previous chapter, we reviewed the map-scaling problem in SLAM and considered a variety of methods proposed to increase the efficiency of SLAM algorithms. In this chapter, we focus more intently on one of these approaches, the *Constant Time SLAM* (CTS) algorithm [50]. While CTS succeeds in achieving consistency and convergence with $\mathcal{O}(1)$ order of growth, it suffers from a slower rate of convergence in comparison to full covariance SLAM. In this chapter, we present two new algorithms — CTS 2.0 and Network-Optimized SLAM (NOS) — that build on the original CTS algorithm to achieve an improved rate of convergence. All three algorithms are implemented with real data and compared to the full covariance solution. Our results provide insights into the choice of error metrics for measuring the global uncertainty of SLAM algorithms that use submap network representations.

4.1 Constant Time SLAM (CTS) Algorithm

We begin by reviewing CTS and investigating its properties. Constant Time SLAM [50] is the first algorithm that achieves three vital requirements for efficient large-scale mapping: consistency, convergence, and constant time updates. It decouples the global map into multiple, locally referenced submaps and maintains estimates for entities in each submap that are uncorrelated to the other submaps. Since each estimate is never fused with the others, and measurements are used in only one submap, the CTS algorithm can guarantee

the consistency of each submap. Also, the method runs in constant time because estimates of the global locations of submaps can be evaluated locally and independently from the number of submaps. Finally, CTS achieves convergence to the full covariance solution by chaining together the local submaps, each of which converge under the assumptions described in Chapter 3 [50]. Although CTS is one of the best algorithms for the map-scaling problem, there considerable room for improvement remains. In this section, we examine how the CTS algorithm works and follow that in the next section with a consideration of what can be done to improve the current state of the art.

Let us begin with a description of the operations performed by the CTS algorithm in each individual local map. In a local region, CTS solves the full covariance SLAM problem using any kind of consistent solution[1]. The state vector is composed of the locations of the surveying robot and a number of static features. The robot is controlled by a series of control inputs, $\mathbf{u}[k]$, and obtains a set of sensor measurements, $\mathbf{z}[k]$, at each discrete time step k. The estimation process seeks to find the best description of the following conditional probability density (which is assumed to be jointly Gaussian):

$$p\left(\mathbf{x}[k]\middle|\mathbf{Z}^k, \mathbf{U}^k\right) \quad , \tag{4.1}$$

where $\mathbf{Z}^k$ and $\mathbf{U}^k$ are all the measurements and inputs up to time k from the start.

The theoretical development of CTS for local SLAM processing is generally the same as in the standard Kalman filter SLAM formulation. The only difference to be considered in CTS is the coordinate frame of each local map, as shown in Figure 4-1. There are two kinds of frames, one for the first local map and one for the other local maps. First, the origin of the first map serves simultaneously as its own origin as well as the origin of all the submaps, and is set from the beginning. Usually, its location is established as the pose of the robot at the instant the robot begins to move, but there is no limitation for its location. Since when a survey begins, no features are known *a priori*, the state vector at the onset includes only the robot's pose. This coordinate system for this submap is identical to that of a typical full covariance SLAM system.

Each of the remaining local maps, however, has a special local coordinate system. In each local submap, the local origin is chosen to coincide with a local feature in the map.

[1]In this thesis, the extended Kalman filter [76] is employed for the local solution.

Hence, the locations of all features in the submap are described with respect to the location of this feature. This feature is called the *root feature* or *root entity* [50] of the submap and is assumed to be perfectly known in a local sense. (Another term that is widely used is the *base reference* of a stochastic map [79].) Because we know the pose root feature exactly in the local frame, the robot does not include the root feature in the local SLAM state vector. The management of these root entities is the key idea of the CTS algorithm, which is examined in detail in the next subsection.

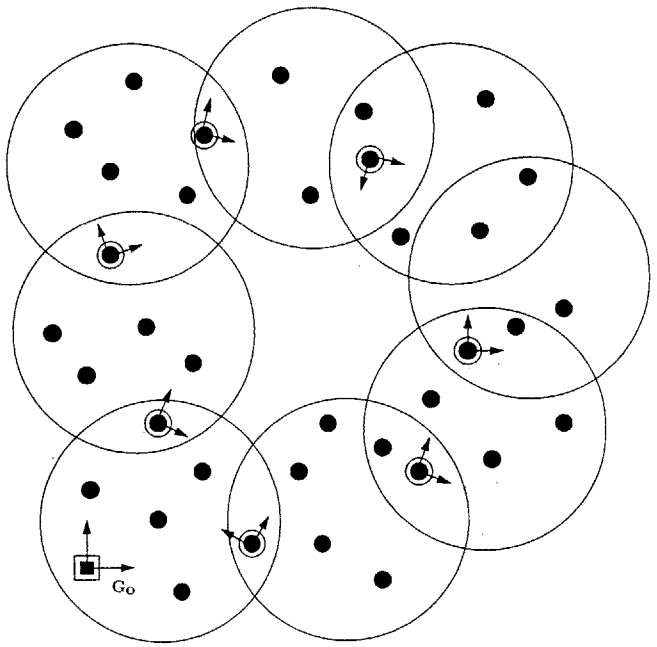

Figure 4-1: The origin of the local coordinate frame is chosen to coincide with one of the features. We call the feature whose pose is located at the local origin the root feature of its submap. The origin of the first submap, however, is not located on top of a feature; it serves as the global origin during the CTS mission.

4.1.1 Manipulating local maps in CTS

Following [50], we now introduce several notational conventions to describe the operation of the CTS algorithm. First, we define T_j^i to be the relative pose of feature j with respect to the feature i. Here, we assume that the feature i has an orientation so that the vector T_j^i is uniquely determined. T_j^i is applied to any two features if they are in the same local map. If two features are included in more than one local map, there exist two T_j^i's and

they are distinct because CTS does not share any estimated information between maps and relies instead on a local independent SLAM solution. In order to explicitly distinguish between the two T_j^i's, a map ID is associated with each instance of T_j^i. Consequently, ${}^{[m,k]}T_j^i$ represents the relative pose of j with respect to i when the two features are in the local map m whose root feature is the feature k. For simplicity, if the local map and its root are obvious or do not matter, we sometimes omit the upper-left indexing in the remainder of the chapter.

According to this notation, the operations of *compounding* and *inversion* of approximate coordinate transformation can be written in the following form:

$$\begin{aligned} T_i^j &= \ominus T_j^i && \text{(inversion)} \\ T_k^i &= T_j^i \oplus T_k^j && \text{(compounding)} \end{aligned} \quad . \tag{4.2}$$

Also, a composite relationship between T_k^i and T_k^j when T_j^i is given can be easily calculated by

$$T_k^j \;=\; T_i^j \oplus T_k^i \;=\; \ominus T_j^i \oplus T_k^i \quad . \tag{4.3}$$

The linearized relationships in the covariance matrices associated with the above operation can be written in the following form [74, 76]:

$$\mathbf{P}_{ji,ji} \approx \mathbf{J}_{\ominus}\mathbf{P}_{ij,ij}\mathbf{J}_{\ominus}^{T} \qquad (inversion) \quad . \quad (4.4)$$

$$\mathbf{P}_{ik,ik} \approx \mathbf{J}_{\oplus}\begin{bmatrix}\mathbf{P}_{ij,ij} & \mathbf{P}_{ij,jk}\\ \mathbf{P}_{ij,jk}^{T} & \mathbf{P}_{jk,jk}\end{bmatrix}\mathbf{J}_{\oplus}^{T} \qquad (compounding) \quad . \quad (4.5)$$

$$\begin{aligned}\mathbf{P}_{jk,jk} &\approx \mathbf{J}_{\oplus}\begin{bmatrix}\mathbf{P}_{ji,ji} & \mathbf{P}_{ji,ik}\\ \mathbf{P}_{ji,ik}^{T} & \mathbf{P}_{ik,ik}\end{bmatrix}\mathbf{J}_{\oplus}^{T}\\ &= \mathbf{J}_{\oplus}\begin{bmatrix}\mathbf{J}_{\ominus}\mathbf{P}_{ij,ij}\mathbf{J}_{\ominus}^{T} & \mathbf{J}_{\ominus}\mathbf{P}_{ij,ik}\\ \mathbf{P}_{ij,ik}^{T}\mathbf{J}_{\ominus}^{T} & \mathbf{P}_{ik,ik}\end{bmatrix}\mathbf{J}_{\oplus}^{T}\\ &= \mathbf{J}_{\oplus}\begin{bmatrix}\mathbf{J}_{\ominus} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{P}_{ij,ij} & \mathbf{P}_{ij,ik}\\ \mathbf{P}_{ij,ik}^{T} & \mathbf{P}_{ik,ik}\end{bmatrix}\begin{bmatrix}\mathbf{J}_{\ominus}^{T} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\mathbf{J}_{\oplus}^{T}\\ &= \begin{bmatrix}\mathbf{J}_{1\oplus}\mathbf{J}_{\ominus} & \mathbf{J}_{2\oplus}\end{bmatrix}\begin{bmatrix}\mathbf{P}_{ij,ij} & \mathbf{P}_{ij,ik}\\ \mathbf{P}_{ij,ik}^{T} & \mathbf{P}_{ik,ik}\end{bmatrix}\begin{bmatrix}\mathbf{J}_{\ominus}^{T}\mathbf{J}_{1\oplus}^{T}\\ \mathbf{J}_{2\oplus}^{T}\end{bmatrix} \quad (composite\ \ relationship) \quad . \quad (4.6)\end{aligned}$$

In the above, $\mathbf{J}_{\ominus}$ and $\mathbf{J}_{\oplus} = \begin{bmatrix}\mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus}\end{bmatrix}$ are appropriate Jacobians of the inversion and compounding operations, respectively.

As a key manipulation in CTS, a local origin is shifted from one feature to another feature according to the features' global uncertainties. Leonard and Newman [50] define this operation as *root shifting*, $\mathcal{S}$. The process can be written as

$$\begin{aligned}{}^{[m,j]}T_{1:n} &= \mathcal{S}_{i\to j}\left({}^{[m,i]}T_{1:n}\right)\\ &= \begin{bmatrix}\ominus^{[m,i]}T_j \oplus^{[m,i]} T_1\\ \ominus^{[m,i]}T_j \oplus^{[m,i]} T_2\\ \vdots\\ \ominus^{[m,i]}T_j \oplus^{[m,i]} T_n\end{bmatrix} \quad .\end{aligned} \qquad (4.7)$$

Figure 3-7 shows this root shifting operation. We should note that root shifting does not change the determinant of the covariance matrix of the local map. That is, in a local map, the choice of root feature does not improve or corrupt the estimated uncertainty. A proof of this property follows:

Proof. As shown above in Equation 4.7, root shifting is a combination of two

basic operations: Inversion (unitary) and compounding (binary). By root shifting, every local pose with respect to the old root is transformed to that of the new root. The corresponding covariance matrices (${}^{[\cdot,i]}\mathbf{P}$ for the old covariance matrix and ${}^{[\cdot,j]}\mathbf{P}$ for new one) have the following relationships:

$$
{}^{[\cdot,j]}\mathbf{P} = \begin{bmatrix}
\mathbf{P}_{j1,j1} & \mathbf{P}_{j1,j2} & \cdots & \mathbf{P}_{j1,ji} & \cdots & \mathbf{P}_{j1,jn} \\
\mathbf{P}_{j2,j1} & \mathbf{P}_{j2,j2} & \cdots & \mathbf{P}_{j2,ji} & \cdots & \mathbf{P}_{j2,jn} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\mathbf{P}_{j(i-1),j1} & \mathbf{P}_{j(i-1),i2} & \cdots & \mathbf{P}_{j(i-1),ji} & \cdots & \mathbf{P}_{j(i-1),jn} \\
\mathbf{P}_{ji,j1} & \mathbf{P}_{ji,j2} & \cdots & \mathbf{P}_{ji,ji} & \cdots & \mathbf{P}_{ji,jn} \\
\mathbf{P}_{j(i+1),j1} & \mathbf{P}_{j(i+1),i2} & \cdots & \mathbf{P}_{j(i+1),ji} & \cdots & \mathbf{P}_{j(i+1),jn} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\mathbf{P}_{jn,j1} & \mathbf{P}_{jn,j2} & \cdots & \mathbf{P}_{jn,ji} & \cdots & \mathbf{P}_{jn,jn}
\end{bmatrix}
$$

$$
\begin{aligned}
{}^{[\cdot,j]}\mathbf{P} &= \mathbf{J}_{\mathcal{S}}\,{}^{[\cdot,i]}\mathbf{P}\mathbf{J}_{\mathcal{S}}^{T} \\
&= \begin{bmatrix}
\mathbf{J}^1_{2\oplus} & 0 & \cdots & 0 & \mathbf{J}^1_{1\oplus}\mathbf{J}^1_{\ominus} & 0 & \cdots & 0 \\
0 & \mathbf{J}^2_{2\oplus} & \cdots & 0 & \mathbf{J}^2_{1\oplus}\mathbf{J}^2_{\ominus} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \mathbf{J}^{j-1}_{2\oplus} & \mathbf{J}^{j-1}_{1\oplus}\mathbf{J}^{j-1}_{\ominus} & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & \mathbf{J}^{j}_{\ominus} & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & \mathbf{J}^{j+1}_{1\oplus}\mathbf{J}^{j+1}_{\ominus} & \mathbf{J}^{j+1}_{2\oplus} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & \mathbf{J}^{n}_{1\oplus}\mathbf{J}^{n}_{\ominus} & 0 & \cdots & \mathbf{J}^{n}_{2\oplus}
\end{bmatrix}
\begin{bmatrix}
\mathbf{P}_{i1,i1} & \mathbf{P}_{i1,i2} & \cdots & \mathbf{P}_{i1,ij} & \cdots & \mathbf{P}_{i1,in} \\
\mathbf{P}_{i2,i1} & \mathbf{P}_{i2,i2} & \cdots & \mathbf{P}_{i2,ij} & \cdots & \mathbf{P}_{i2,in} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\mathbf{P}_{i(j-1),i1} & \mathbf{P}_{i(j-1),i2} & \cdots & \mathbf{P}_{i(j-1),ij} & \cdots & \mathbf{P}_{i(j-1),in} \\
\mathbf{P}_{ij,i1} & \mathbf{P}_{ij,i2} & \cdots & \mathbf{P}_{ij,ij} & \cdots & \mathbf{P}_{ij,in} \\
\mathbf{P}_{i(j+1),i1} & \mathbf{P}_{i(j+1),i2} & \cdots & \mathbf{P}_{i(j+1),ij} & \cdots & \mathbf{P}_{i(j+1),in} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\mathbf{P}_{in,i1} & \mathbf{P}_{in,i2} & \cdots & \mathbf{P}_{in,ij} & \cdots & \mathbf{P}_{in,in}
\end{bmatrix}
\mathbf{J}_{\mathcal{S}}^{T}
\end{aligned}
\tag{4.8}
$$

In Equation (4.8), the Jacobian matrix $\mathbf{J}_{\mathcal{S}}$ forms a triangular matrix after appropriate row and column interchanges. With the following lemma relating to the determinant of a matrix [41],

$$
\begin{vmatrix}
\mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1n} \\
\mathbf{0} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_{nn}
\end{vmatrix}
= |\mathbf{A}_{11}|\,|\mathbf{A}_{22}|\cdots|\mathbf{A}_{nn}| \quad . \tag{4.9}
$$

the relationship for the determinants can be calculated from the equation (4.8), as

$$\begin{aligned}\left|{}^{[\cdot,j]}\mathbf{P}\right| &= |\mathbf{J}_{\mathcal{S}}| \ \left|{}^{[\cdot,i]}\mathbf{P}\right| \ |\mathbf{J}_{\mathcal{S}}^{T}| \\ &= \left(\prod_{k=1}^{n} |\mathbf{J}_{k\oplus}|\right) \ \left|{}^{[\cdot,i]}\mathbf{P}\right| \ \left(\prod_{k=1}^{n} |\mathbf{J}_{k\oplus}^{T}|\right) \\ &= 1 \cdot \left|{}^{[\cdot,i]}\mathbf{P}\right| \cdot 1 \\ &= \left|{}^{[\cdot,i]}\mathbf{P}\right| \quad .\end{aligned} \tag{4.10}$$

In the above, we use $|\mathbf{J}| = 1$ for all Jacobians because they are merely coordinate transformation operations [2]. Therefore, in a local map, the choice of root feature does not improve or corrupt the estimated uncertainty. The determinants before and after root shifting are always the same.

In addition to this property for the determinant of the coordinate transformation operation, we should also note that the global location of any local feature can be constructed by a simple compounding operation. Since CTS maintains the global pose estimate of the root feature, we can choose any local feature location as the other input of the binary operation $\oplus$ and it will evaluate the global location estimate of the local feature. This is illustrated in Figure 4-2.

[2]This constant determinant holds in a two-dimensional environment, such as mapping objects on the seafloor when the depth is known. If we use Cartesian coordinates in a three-dimensional environment, the determinant is not generally 1. This topic will be investigated in future work. In the remainder of this thesis, we confine our focus to two dimensional environments.

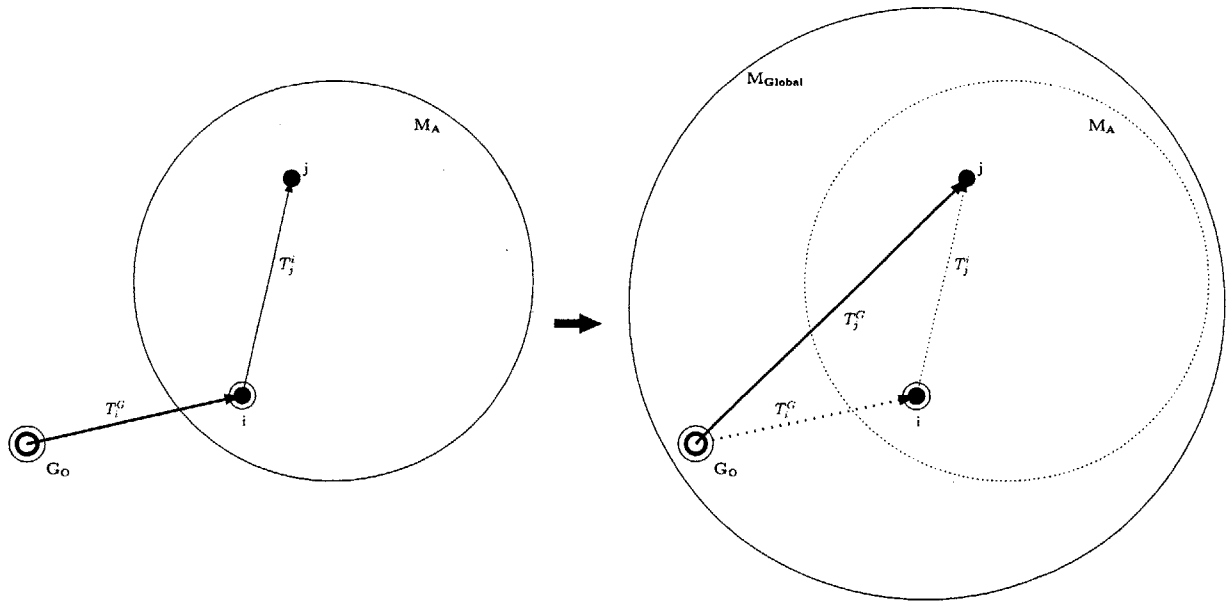

Figure 4-2: In the CTS algorithm, the global uncertainty of the feature j (T_j^G) is obtained by compounding the global uncertainty of the root feature i (T_i^G) and the local uncertainty of the feature j (T_j^i). That is, $T_j^G = T_i^G \oplus T_j^i$.

An interesting fact in this transformation is that the resultant uncertainty is easily calculated from the two input values ($\mathbf{P}_{Gi,Gi}$ and $\mathbf{P}_{ij,ij}$) for the compounding operation in CTS. Suppose, for example, there are two features in the global map. The covariance matrix of the features will be $\begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,Gj} \\ \mathbf{P}_{Gi,Gj}^T & \mathbf{P}_{Gj,Gj} \end{bmatrix}$. This can be constructed by the given uncertainties $\mathbf{P}_{Gi,Gi}$ and $\mathbf{P}_{ij,ij}$ through the relationship

$$\begin{aligned} \begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,Gj} \\ \mathbf{P}_{Gi,Gj}^T & \mathbf{P}_{Gj,Gj} \end{bmatrix} &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,ij} \\ \mathbf{P}_{Gi,ij}^T & \mathbf{P}_{ij,ij} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{J}_{1\oplus}^T \\ \mathbf{0} & \mathbf{J}_{2\oplus}^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{ij,ij} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{J}_{1\oplus}^T \\ \mathbf{0} & \mathbf{J}_{2\oplus}^T \end{bmatrix} \end{aligned} \quad . \tag{4.11}$$

Therefore, their determinant can be calculated as

$$\left|\begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,Gj} \\ \mathbf{P}^T_{Gi,Gj} & \mathbf{P}_{Gj,Gj} \end{bmatrix}\right| = |\mathbf{I}|\,|\mathbf{J}_{2\oplus}|\;\; |\mathbf{P}_{Gi,Gi}|\,|\mathbf{P}_{ij,ij}|\;\; |\mathbf{I}|\,|\mathbf{J}^T_{2\oplus}| \\ = |\mathbf{P}_{Gi,Gi}|\,|\mathbf{P}_{ij,ij}| \quad . \tag{4.12}$$

In the above equation (4.11), $\mathbf{P}_{Gi,ij} = \mathbf{0}$ is used because T_i^G and T_j^i are uncorrelated in CTS; that is, they do not share any information with each other. Consequently, the determinant can be evaluated by the product of two local values[3].

This transformation to the global frame can be extended to the transformation of all the features in a local map. Figure 4-3 shows this transformation and Equations (4.13 and 4.14) represent the corresponding determinant calculation.

$$\begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,GA} \\ \mathbf{P}^T_{Gi,GA} & \mathbf{P}_{GA,GA} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,A} \\ \mathbf{P}^T_{Gi,A} & \mathbf{P}_{A,A} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{J}^T_{1\oplus} \\ \mathbf{0} & \mathbf{J}^T_{2\oplus} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_{1\oplus} & \mathbf{J}_{2\oplus} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{A,A} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{J}^T_{1\oplus} \\ \mathbf{0} & \mathbf{J}^T_{2\oplus} \end{bmatrix} \quad . \tag{4.13}$$

And the corresponding determinant is written as

$$\left|\begin{bmatrix} \mathbf{P}_{Gi,Gi} & \mathbf{P}_{Gi,GA} \\ \mathbf{P}^T_{Gi,GA} & \mathbf{P}_{GA,GA} \end{bmatrix}\right| = |\mathbf{I}|\,|\mathbf{J}_{2\oplus}|\;\; |\mathbf{P}_{Gi,Gi}|\,|\mathbf{P}_{A,A}|\;\; |\mathbf{I}|\,|\mathbf{J}^T_{2\oplus}| \\ = |\mathbf{P}_{Gi,Gi}|\,|\mathbf{P}_{A,A}| \quad . \tag{4.14}$$

[3] T_i^G (with feature i) can be thought of as a local operation compared to the global region (with features i and j) described as a big circle in Figure 4-3

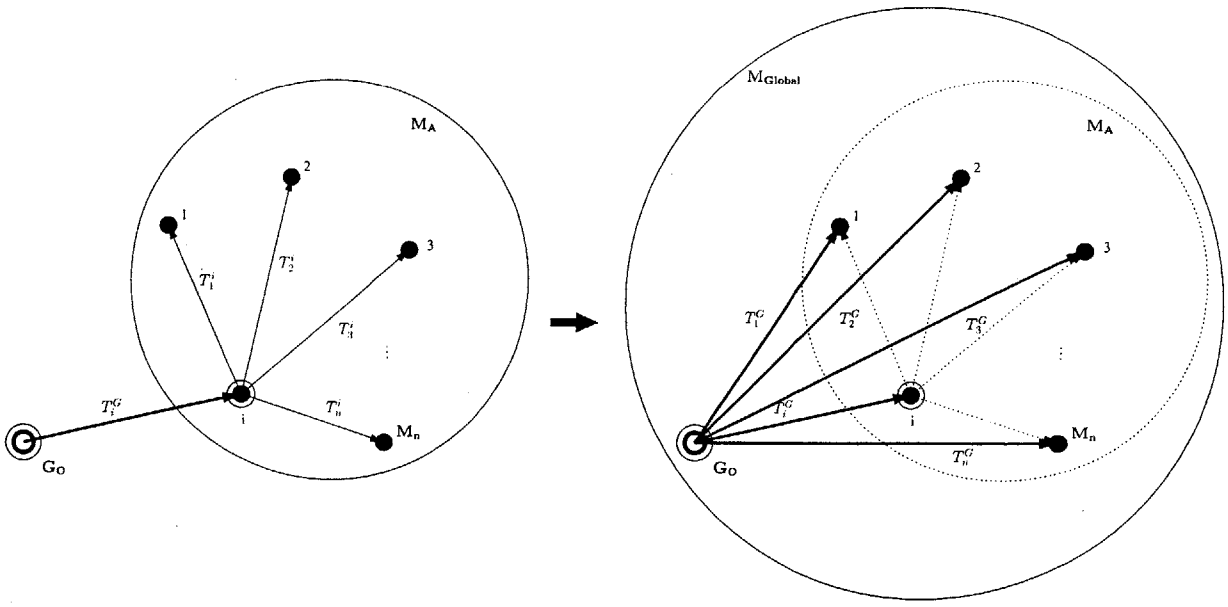

Figure 4-3: The spatial pose and the uncertainty of every feature in the submap $\mathbf{M}_A$ can be transformed to the global frame through the compounding processes. As the measure of the global uncertainty, the determinant of the features of $\mathbf{M}_A$ is calculated through the equation (4.14).

One very important result from the above relationship is that the uncertainty can be written as a product of two given determinants. Since the second term of the product is not changed by root shifting, we conclude that *the global uncertainty of a local map is determined by the root feature's global uncertainty.* Therefore, finding the best root feature for each submap is the key process of the CTS algorithm.

4.1.2 Algorithm description

As described in Leonard and Newman [50], the four basic parts of the CTS algorithm are (1) SLAM processing within local maps, (2) map management, (3) map location estimation, and (4) computation of global state estimates for all features in a map. Figure 4-4 shows the flowchart of the CTS algorithm.

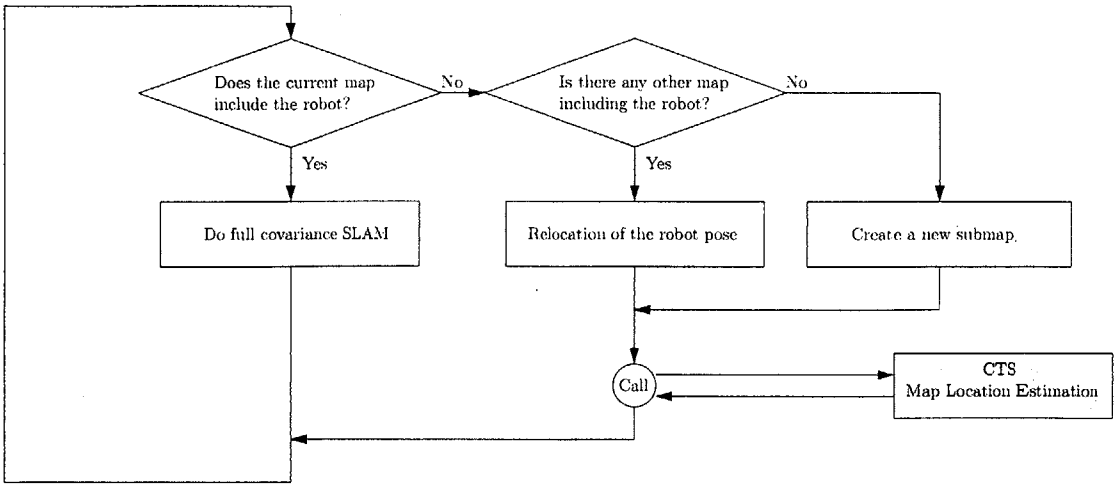

Figure 4-4: Flowchart of the CTS algorithm.

a. SLAM within local maps

As described earlier, CTS computes a full covariance SLAM solution in each local map in the same way that one would compute a full covariance solution for a single map SLAM problem. The robot moves according to the given control input, makes observations, and does either feature initialization or Kalman update for re-observed features (as described in Chapter 2). Every feature initialized within a local map is included in a state vector and its estimate is updated by the extended Kalman filter.

All elements of the state vector are location estimates with respect to the origin of the local map. The origin of each map, except the first local map, coincides with the location of a feature and is not included in the state vector (because in the local map, the pose of the root feature is perfectly known).

At any given moment, CTS decides in which map the robot is located. When the robot is in multiple maps, CTS recognizes the robot as being in the oldest map. This map assignment is also applied to the measurements obtained from the robot's sensors. If the robot is in map $\mathbf{M}_A$, the observation made by the robot at that moment can be used only for the map $\mathbf{M}_A$. This prevents observed information from being used more than once. Figure 4-5 illustrates the manner in which each measurement is used for local state estimation in a single submap.

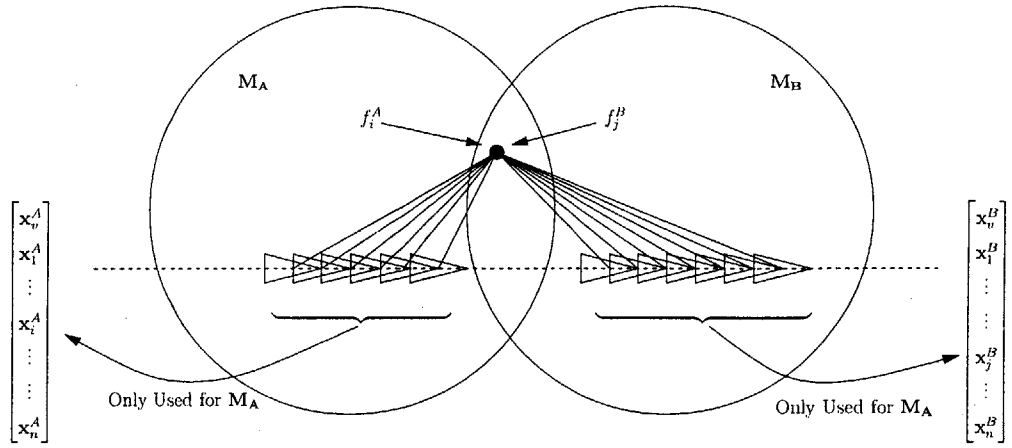

Figure 4-5: Each measurement is assigned to only one submap so that no information is shared by multiple local maps. Consequently, f_i^A and f_j^B are regarded as separate features.

Since no information is shared between submaps, some features can be initialized in multiple local maps. For example, in Figure 4-5, feature f_i^A in $\mathbf{M}_A$ and feature f_j^B in $\mathbf{M}_B$ have different estimated values and uncertainties. The fact that they correspond to the same feature, however, is used during the map matching process.

b. Map management

Before moving on, let us make the meaning of 'map' clear. The term *map* has two meanings in the SLAM community. The first meaning is as a set of the features and the robot as estimated states. When the term map is used in this context, it includes the estimated state vector and the corresponding covariance matrix. *Map building* implies the addition of a new estimated feature to the existing states and an improvement of the states through the use of various estimation techniques. The *uncertainty* of a map refers to the quality of the state estimation.

The other definition of map refers to geometric regions. When we say '*the robot is in a map*', we imagine that a robot is within a geometric region such as a rectangle or a circle. If we say '*the robot leaves the map*', it usually means that the robot transfers across a geometric boundary. Since these two meanings do not conflict with each other in most cases, we use 'map' for both cases without distinct clarification.

A local map in CTS environment is created whenever both of the following are satisfied: (1) no existing map contains the robot and (2) at least one feature exists within the range

of sensors attached the robot. The feature should also be included in the previous map [4]. Each map forms a circle whose center is located at the position of the robot when the map is created and whose radius is pre-set with the proper value for the environment. When a map is created, therefore, the robot is at the center of the map, and the origin of the local frame is set on top of a feature within the map. Since the map is defined by the robot's pose, some mapped features might actually be located outside of a local map if they can be seen by the robot that is in the map. This spatial decomposition representation, using a set of overlapping map regions defined by a set of vehicle poses, was first introduced in the SLAM literature by Leonard and Feder [54].

If the robot is in a local map, it computes a full covariance SLAM for those local regions described in the previous subsection. If the robot leaves a local map, CTS checks if there exists a local map that contains the current robot's location. If multiple maps containing the robot exist, the oldest map is chosen. If there is no existing map, CTS creates a new map [5].

When the robot re-enters a previously visited map, CTS performs relocation to compute a new pose estimate for the robot. That is, the CTS algorithm ignores any spatial information acquired outside of the map and re-localizes the robot whenever it visits this map.

c. Map location estimation

The following, quoted from the paper [50], shows the steps of CTS algorithm for *map location estimation.* It should be noted that this subroutine can be executed independently of the main process of the CTS algorithm described in the previous two subsections.

[4] There was no need for the second condition in [50] because the paper assumed a sufficient density of features in the environment.

[5] In fact, CTS waits until the robot finds at least one shared feature. In this case, the robot is in the previous map even though it leaves the geometric region associated with that map.

Algorithm ***Map Location Estimation***

1: Select a map p, to improve which is currently referenced to the root entity i
2: Create a set, $\mathcal{N}$, containing the ID's of all nearby maps including p – the map to be improved.
3: For each $q \in \mathcal{N}$, $q \neq p$ create a set $\mathcal{C}_{p,q}$ of ID's of features that are present in both maps.
4: For each frame $\mathcal{F}_k$ attached to feature $k \in \mathcal{C}_{p,q}$ calculate its globally referenced location ${}^{[q,\cdot]}T_k^G$ and uncertainty ${}^{[q,\cdot]}\mathbf{P}_k^G$ using the location estimate of feature k within map q and its current location estimate.
5: Pick the map $q^\star$ and entity ID $k^\star$ such that:
$[q^\star, k^\star] = \arg\min \left\{ \left| {}^{[q,\cdot]}\mathbf{P}_k^G \right| \right\}$
6: If $q^\star = p$ and $k^\star = i$ then stop. The map p cannot be improved.
7: Root shift map p to $k^\star$ from i — reference all entities in map p to a coordinate frame attached to entity $k^\star$.
8: Replace T_p^G and $\mathbf{P}_p^G$ with ${}^{[q^\star,\cdot]}T_{k^\star}^G$ and ${}^{[q^\star,\cdot]}\mathbf{P}_{k^\star}^G$ respectively.

Figure 4-6 illustrates how map location estimation works. Suppose the map $\mathbf{M}_P$ is about to be updated and the map shares features with three neighboring maps $\mathbf{M}_A$, $\mathbf{M}_B$, and $\mathbf{M}_C$. Figure 4-6.(a) illustrates this situation. It should be noted that every shared feature has two indices: (1) one for $\mathbf{M}_P$ as shown in the Figure 4-6.(a) and (2) the other for one of the neighboring maps $\mathbf{M}_A$, $\mathbf{M}_B$, and $\mathbf{M}_C$ as shown in Figure 4-6.(b). After identifying all shared features, CTS calculates the global uncertainty of each feature by using compounding. In Figure 4-6.(b), we can see which feature is combined with which root feature. Figure 4-6.(c) captures the results. All root feature candidates are compared through their global uncertainties. Finally, CTS appoints the feature with the least uncertainty as the root feature of the map $\mathbf{M}_P$.

The map location estimation is executed in constant time with reasonable assumptions: (1) the number of shared features and (2) the number of neighboring maps are bounded by appropriate constants. If features are equally distributed over the entire region and the density of features are neither too small nor too large, the given assumptions can be applied. The map location estimation chooses the best feature by comparing the determinant values of the candidate features. If we set the upper bound of the shared features as N_{sh} and the upper bound of the neighboring maps as N_{map}, the number of comparisons is bounded by the constant $N_{sh} \times N_{map}$, which implies the running time of the map location estimation is $\mathcal{O}(1)$. Also, since the map location estimation is executed only when the robot leaves a submap, it can be processed separately from the main processes of the CTS algorithm.

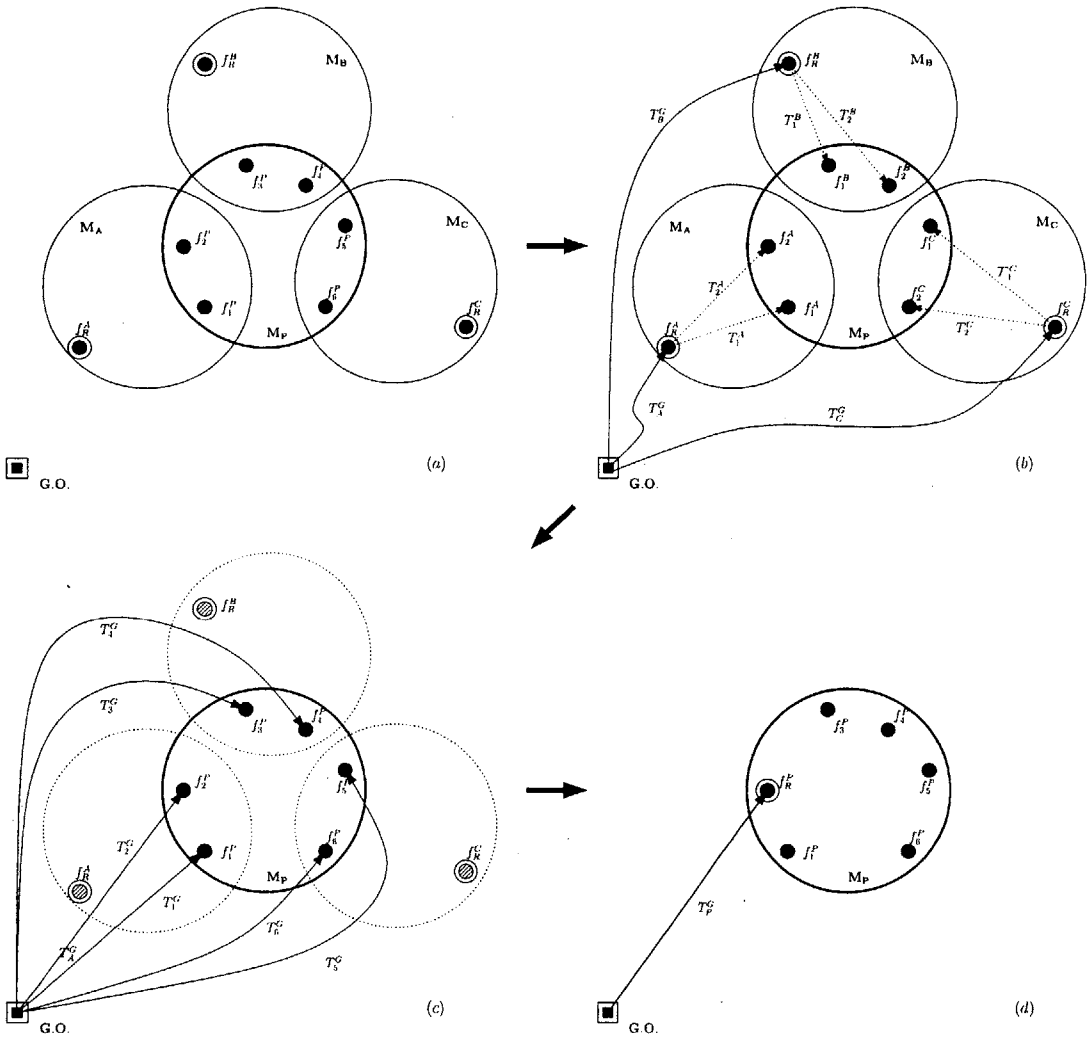

Figure 4-6: In order to update the global location of $\mathbf{M}_P$, the global uncertainties through the root features of $\mathbf{M}_P$'s neighboring maps ($\mathbf{M}_A$, $\mathbf{M}_B$, and $\mathbf{M}_C$) are compared. From this comparison, the feature providing the minimum uncertainty (T_p^G) is chosen as the root feature of $\mathbf{M}_P$.

d. Computation of global state estimates for all features in a map

As already described, the global state estimates for all features can be obtained through a simple compounding operation.(See Figure 4-2 and 4-3) Also, we know that every feature in a local map ($\mathbf{M}_A$) comes to be fully correlated when all features are observed infinitely many times. Since the location of the root feature is perfectly known, all features' estimates in the map $\mathbf{M}_A$ become perfect [6]. This means that the local map $\mathbf{M}_A$ is built without uncertainty

[6] Even though the root feature is not included in the local state vector, its cross correlations with the other features in the map can be improved while a different root feature is selected by CTS

if the root feature is correctly given.

While serving as the root entity of $\mathbf{M}_A$, the feature simultaneously appears in another local map ($\mathbf{M}_B$). Therefore, the feature will be perfectly known after infinite observation in the map $\mathbf{M}_B$, if the root feature of $\mathbf{M}_B$ is correctly given. Analogous reasoning can be applied to the other maps so that we can conclude that all the features become fully correlated and their uncertainties will converge to the lowest bound[7].

4.2 Review of CTS

In the previous section, we demonstrated what the constant time SLAM algorithm is, how it works, and why it satisfies three key requirements for large-scale SLAM problems. In this section, we examine the CTS algorithm for areas of improvement in two respects and prepare for the next section that will provide specific algorithms to increase the effectiveness of the CTS algorithm.

4.2.1 Nonlinearity in determinants

One of the key steps of the CTS algorithm is the estimation of the global location of submaps through the use of compounding operations. As discussed in Chapter 3, the determinant of a sum of two matrices is not equal to the sum of the determinants of the two matrics. That is,

$$|\mathbf{A}+\mathbf{B}| \neq |\mathbf{A}|+|\mathbf{B}| \tag{4.15}$$

for two square matrices $\mathbf{A}$ and $\mathbf{B}$ [41]. Also, for $n \times n$ non-negative definite matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, $|\mathbf{B}| > |\mathbf{C}|$ does not imply that

$$|\mathbf{A}+\mathbf{B}| > |\mathbf{A}+\mathbf{C}| \quad . \tag{4.16}$$

This inequality holds only when $\mathbf{B}-\mathbf{C}$ is also non-negative definite [41]. Similarly, for $n \times n$ non-negative matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$, $|\mathbf{A}| > |\mathbf{B}|$ and $|\mathbf{C}| > |\mathbf{D}|$ do not guarantee that

during mission. Or, a special treatment can be used to update the root feature's estimate when it is observed by the robot.

[7] See Chapter 3 for the description of the lowest bound.

$$|\mathbf{A}+\mathbf{C}| > |\mathbf{B}+\mathbf{D}| \quad . \tag{4.17}$$

Figure 4-7 illustrates an example in which this inequality does not hold.

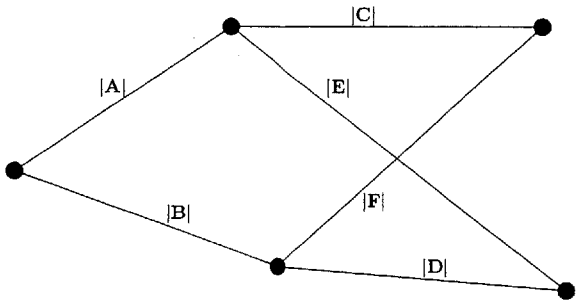

Figure 4-7: Even though $|\mathbf{A}| > |\mathbf{B}|$ and $|\mathbf{C}| > |\mathbf{D}|$, the determinants of their summation does not imply that $|\mathbf{A}+\mathbf{C}| > |\mathbf{B}+\mathbf{D}|$.

Now, let us apply this property to the root feature selection process in CTS. As shown in Figure 4-6, the global uncertainties of all shared features are compared in CTS to find the least uncertain feature. For the attainment of map location estimation, the global uncertainties of the neighboring maps need to be combined with each shared feature. One question this process raises is, "How can we evaluate the global uncertainty of the neighboring map?" Figure 4-8 illustrates the root feature selection steps of two maps, $\mathbf{M}_A$ and $\mathbf{M}_P$. Figure 4-8(a) represents a part of Figure 4-6 and illustrates two compounding results, $T_A^G \oplus T_1^A$ and $T_A^G \oplus T_2^A$, that are compared to find the path of the lower uncertainty. Now, let us examine how the feature f_R^A was chosen as the root feature of $\mathbf{M}_A$ in Figure 4-8(b). It should be clear that the same procedure is used as in map $\mathbf{M}_P$. That is, $T_B^G \oplus T_1^B$ and $T_B^G \oplus T_2^B$ are compared, and the path $G.O. \rightarrow f_R^B \rightarrow f_1^B$ is chosen. The global uncertainty via the path is assigned to the feature f_1^B, and the feature stores the value as T_R^A and is prepared for the neighbor map's effort to determine its root feature.

As already shown, only two paths $(G.O \rightarrow f_1^B(=f_R^A) \rightarrow f_1^A)$ and $(G.O \rightarrow f_1^B(=f_R^A) \rightarrow f_2^A)$ are compared when CTS searches the root feature of $\mathbf{M}_P$ [8].

[8] Those paths are identical with $(G.O \rightarrow f_R^B \rightarrow f_1^B \rightarrow f_1^A)$ and $(G.O \rightarrow f_R^B \rightarrow f_1^B \rightarrow f_2^A)$ respectively.

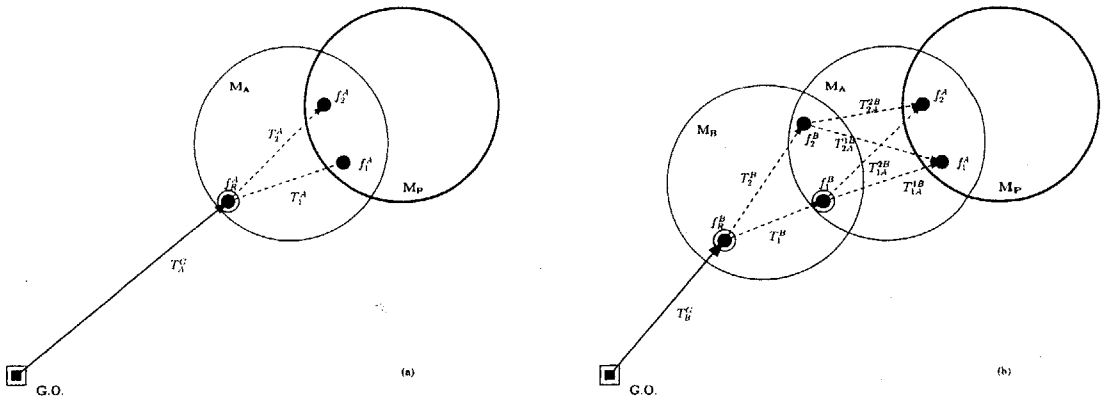

Figure 4-8: In order to choose the root feature of $\mathbf{M}_P$, CTS compares two possible paths through the root feature of its neighbor map, $\mathbf{M}_A$. However, if we include another shared features between $\mathbf{M}_A$ and its neighbor $\mathbf{M}_B$, another paths might exist in addition to these two paths and they might provide lower global uncertainties than the paths through the root feature of $\mathbf{M}_A$.

We should note that even though two more possible paths exist — $(G.O \rightarrow f_R^B \rightarrow f_2^B \rightarrow f_1^A)$ and $(G.O \rightarrow f_R^B \rightarrow f_2^B \rightarrow f_2^A)$ — from the global origin to $\mathbf{M}_P$, they are dismissed in CTS before the root feature of $\mathbf{M}_P$ is considered. These two paths, however, may yield lower uncertainties than the paths via f_R^A. Because the linearity does not hold for the determinants, that potential for a lower uncertainty is missed by the CTS algorithm.

The uncertainties of the compounding processes are the sum of the determinants, which do not hold linearity. As a result, the path that provides the least uncertainty might be included in the paths that are eliminated prior to their consideration. In this respect, there is room for improvement in the algorithm for finding a best root feature in CTS.

4.2.2 Information dissemination depth

Currently, the CTS algorithm activates the *map location estimation* routine only when a map transition occurs in order to maintain the constant time computation. When the routine is called, CTS algorithm searches a best root feature of the map to be improved. This means that whenever the routine is called one and only one map is improved. Thus, the improved information is propagated only one submap per one execution of the map location estimation. This limited propagation of the improved information may cause a

significant slow rate of convergence of the CTS algorithm.

To illustrate this point, let us consider an environment that consists of a big loop. As shown in Figure 4-9, all submaps have neighbor maps only in one direction until the robot closes the loop and revisits the previously created submaps. When the robot leaves the last submap($\mathbf{M}_m$) and re-enters the first submap, CTS activates the *map location estimation* routine and compares the root feature candidates. Suppose, for example, that one path directly connected to the global origin is chosen as the root feature and its corresponding uncertainty is much less than the uncertainty of the paths through $\mathbf{M}_{m-1}$. We desire that map $\mathbf{M}_{m-1}$ be improved by the new link via the map $\mathbf{M}_m$. Unfortunately, the CTS algorithm does not currently check this possibility because of the computational complexity. In order to improve the map location estimate of $\mathbf{M}_{m-1}$, the robot should re-visit the map all the way around the loop of the submaps.

Since the improvement depth of map location estimates is limited to one map, the rate of convergence might be very low in some cases.

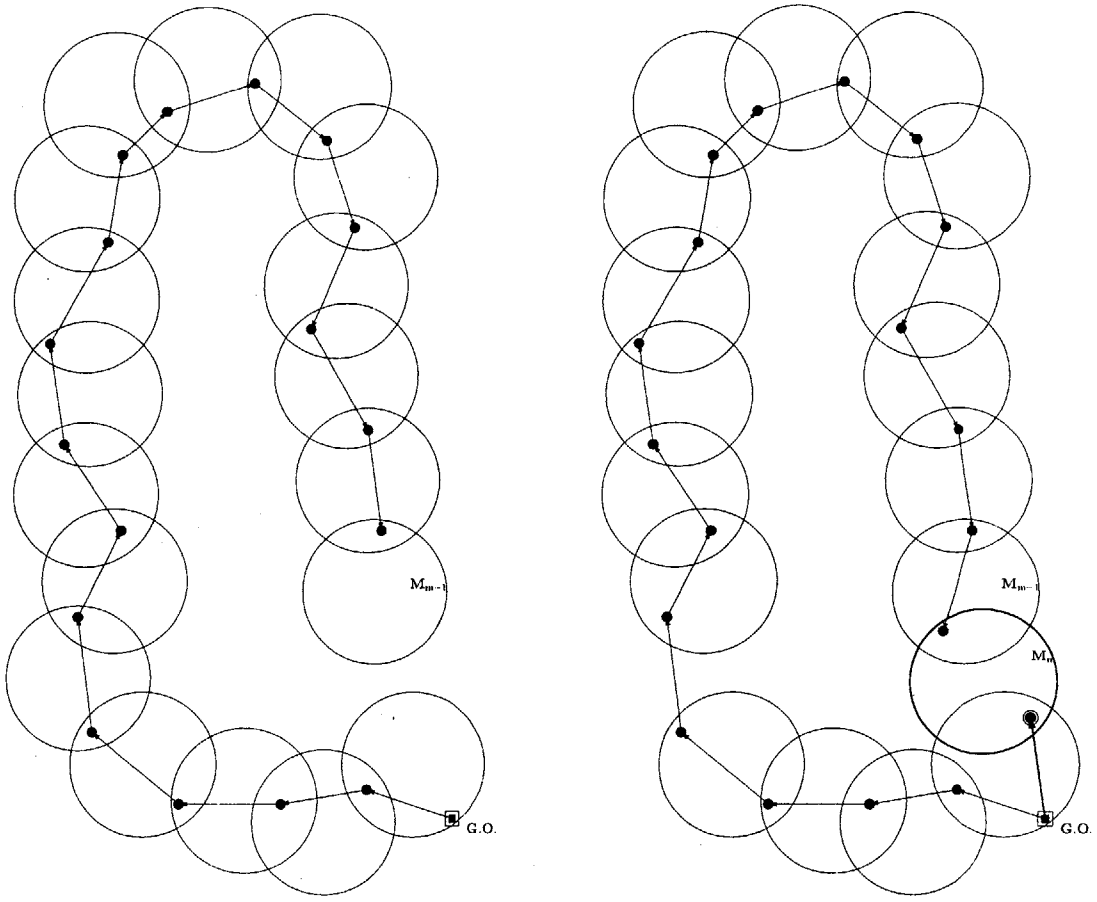

Figure 4-9: When the robot leaves $\mathbf{M}_{m-1}$, $\mathbf{M}_{m-1}$ has only one neighbor map. Therefore, the global uncertainty of $\mathbf{M}_{m-1}$ should be calculated all the way around the submaps. When the robot leaves $\mathbf{M}_m$, however, $\mathbf{M}_m$ has two neighbor maps and chooses the better path, which is directly connected to the global origin. As a result, the global uncertainty of $\mathbf{M}_m$ is much less than that of $\mathbf{M}_{m-1}$. Unfortunately, this less uncertain path through $\mathbf{M}_m$ does not propagate to $\mathbf{M}_{m-1}$ until the robot re-enters $\mathbf{M}_{m-1}$ again.

4.3 Efficient SLAM algorithms

So far, we have examined various methods to resolve the map-scaling issue in SLAM problems. In particular, the Constant Time SLAM algorithm [50] has been studied in detail and shown to satisfy three vital requirements for large-scale SLAM problems. In the previous two subsections, we have also demonstrated that a more effective means for approaching certain large-scale SLAM problem is possible. Now, we will introduce a graphical descrip-

tion of a submap system and suggest two new algorithms that enhance the performance of the CTS algorithm.

4.3.1 Graphical description

Let us define a graph $G(N, E)$ whose nodes and edges are written as follows:

$$\begin{aligned} N &= \{\mathbf{M}_i \mid i = 1, \cdots, n, \quad \text{where n is the number of submaps.}\} \\ E &= \{(i,j),(j,i) \mid \text{if } \mathbf{M}_i \text{ and } \mathbf{M}_j \text{ has at least one shared feature.}\} \quad . \end{aligned} \tag{4.18}$$

Henceforth, as CTS does, we assume that the feature density is high enough throughout the region being mapped so that every submap has at least one neighboring map with which it shares at least one feature. This assumption leads the graph G to be a *connected* graph. That is, *there exists a path between any two nodes in the graph.* Figure 4-10(a) shows an example of this type of graph. Now, let us define a sub-graph $G_T(N, E_T)$ induced from G. The definition of G_T is written as

$$\begin{aligned} N &= \{\mathbf{M}_i \mid i = 1, \cdots, n, \quad \text{where n is the number of submaps.}\} \\ E_T &= \{(i,j) \mid ((i,j) \in E) \wedge (\pi(j) = i)\} \quad . \end{aligned} \tag{4.19}$$

where $\pi(\cdot)$ represent a *parental* relationship. The sub-graph G_T has the same nodes set as G does, but possesses a different edge network from G. In G_T, an edge (i, j) exists only when $\mathbf{M}_i$ is the *parent* of $\mathbf{M}_j$. This parental relationship is established by the CTS algorithm whenever a submap location estimate is updated. If the global uncertainty of $\mathbf{M}_j$ is updated by compounding with the root feature of $\mathbf{M}_i$, $\mathbf{M}_i$ becomes the parent of $\mathbf{M}_j$. (See Figure 4-8. $\mathbf{M}_A$ is the parent of $\mathbf{M}_P$ and $\mathbf{M}_B$ is the parent of $\mathbf{M}_A$.) Of course, this parenthood can always be dismissed when the next map update occurs.

One interesting fact here is that G_T is guaranteed to be a tree. The root of the tree is the map that contains the global origin. No cycle is allowed in the graph G_T as illustrated through the following reasoning:

Proof. (By contradiction) Let us assume there is a cycle in G_T. Suppose $M_p \leftarrow$

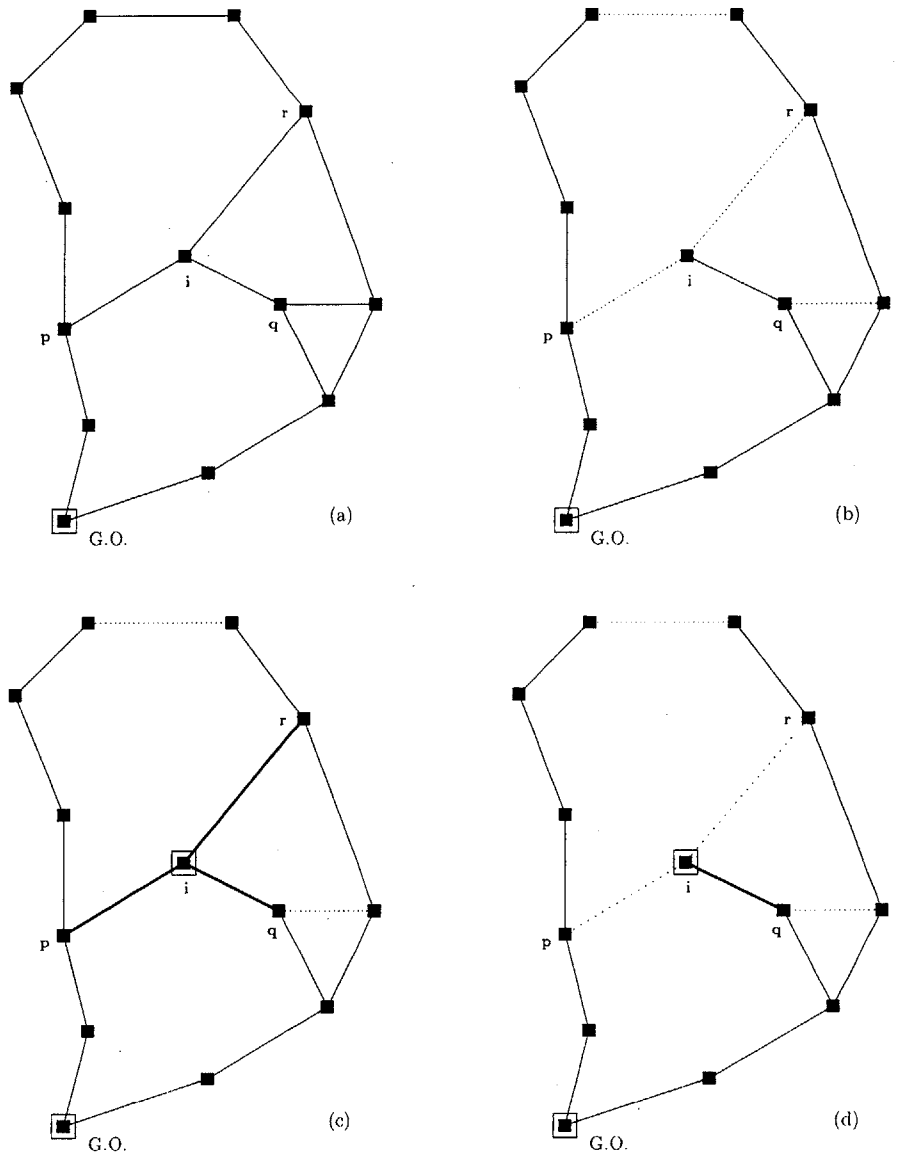

Figure 4-10: Figure (a) represents the graph produced through the neighborhood between submaps. If two submaps share at least one feature, one edge between those two submaps exists. Figure (b) illustrates the tree structure constructed by the CTS algorithm. Since the CTS algorithm makes every submap choose one parent map, no submap (node) can have more than one outgoing edge from itself. Figure (c) captures a map location estimation process when $\mathbf{M}_i$ is considered to be improved. The CTS algorithm compares three possible paths to the submap and chooses $\mathbf{M}_q$ as its parent map. (d) Once this selection process is finished, the other paths via $\mathbf{M}_p$ and $\mathbf{M}_r$ are ignored until another map location estimation for $\mathbf{M}_i$ occurs. Thus, the graph maintains the tree structure during a CTS mission.

$M_q \leftarrow \cdots \leftarrow M_k \leftarrow M_p$ are the nodes composing the cycle. It is well known that, for non-negative definite matrices A and B, $|A+B| \geq |A|+|B|$ [41]. For the compounding process $x_{ij} \oplus x_{jk} = x_{ik}$, the associated covariance matrices

satisfy

$$\mathbf{P}_{ik} = \mathbf{J}_{ij}\mathbf{P}_{ij}\mathbf{J}_{ij}^T + \mathbf{J}_{jk}\mathbf{P}_{jk}\mathbf{J}_{jk}^T \quad . \tag{4.20}$$

where $\mathbf{J}_{ij}$ and $\mathbf{J}_{jk}$ are corresponding Jacobians. Since the determinant of these Jacobians are 1, the following inequality holds:

$$\begin{aligned} \left|\mathbf{J}_{ij}\mathbf{P}_{ij}\mathbf{J}_{ij}^T + \mathbf{J}_{jk}\mathbf{P}_{jk}\mathbf{J}_{jk}^T\right| &\geq \left|\mathbf{J}_{ij}\mathbf{P}_{ij}\mathbf{J}_{ij}^T\right| + \left|\mathbf{J}_{jk}\mathbf{P}_{jk}\mathbf{J}_{jk}^T\right| \\ &= |\mathbf{J}_{ij}|\,|\mathbf{P}_{ij}|\,\left|\mathbf{J}_{ij}^T\right| + |\mathbf{J}_{jk}|\,|\mathbf{P}_{jk}|\,\left|\mathbf{J}_{jk}^T\right| \\ &= |\mathbf{P}_{ij}| + |\mathbf{P}_{jk}| \quad . \end{aligned}$$

$$\therefore \quad |\mathbf{P}_{ik}| \geq |\mathbf{P}_{ij}| + |\mathbf{P}_{jk}| \quad . \tag{4.21}$$

In the last relation, the equality holds only when either $|\mathbf{P}_{ij}|$ or $|\mathbf{P}_{jk}|$ is zero. Now, let us apply this to the nodes in the cycle. Since $\pi(q) = p$ (parent of q is p), the global uncertainty of $\mathbf{M}_q$ is larger than that of $\mathbf{M}_p$. This results from the fact that the global uncertainty of a child node is the compounding result of (1) the global uncertainty of the parent and (2) the local uncertainty between the parent and the child (See Figure 4-2 and 4-3) [9].

Therefore, a child map's global uncertainty is always bigger than its parent's. The remainder of the proof is straightforward. These inequalities contradict the existence of a cycle when the parenthood M_r and M_p is examined. That is,

$$|M_p| \;<\; |M_q| \;<\; \cdots \;<\; |M_r| \;\overset{?}{\lesseqgtr}\; |M_p| \quad . \tag{4.22}$$

This last relation contradicts the given assumption. Therefore, no cycle can exist in the graph G_T.

[9] Even though two uncertainties can be equal, that case is eliminated in the remainder of this thesis, because these compounding operations take place during the mission, and no perfectly known relation is possible until the mission is successfully completed.

Figure 4-10(b) provides an example of G_T. In order to distinguish the submap graph from the network of root features, in the figure squares symbolize the nodes (submaps) of the graph. We note that each node value is the global uncertainty of the submap, and the edges (solid lines) represent only the parenthood between submaps.

4.3.2 Improvement of the nonlinearity, Constant Time SLAM (CTS) 2.0

The improvement of the nonlinearity of the CTS algorithm originates from the fact that the compounding process in CTS is nonlinear. Instead of maintaining only one global connection between the root feature of a local submap and the global origin, the global uncertainties of all the root feature candidates of the submap are maintained. Figure 4-11 illustrates this new algorithm, which is named CTS 2.0. $\mathbf{M}_A$, for example, maintains not only f_R^A, but also the other candidate $f_\star^A$. When $\mathbf{M}_P$ is improved, the paths through $f_\star^A$ are also considered. Through this slight modification, we can reduce the possibility that the root feature found by CTS falls in a local minimum. It is certain that the performance of the new algorithm will always be better than or equal to those from CTS, because all the paths in CTS are still included in the new set of paths.

Furthermore, this new algorithm does not degrade the constant time computation of the CTS algorithm if the number of shared features is bounded by a constant number. The new *map location estimation* routine still examines only the adjacent neighbor maps. The paths between the global origin and the root feature candidates of the neighboring maps, T^Gs, are all encapsulated because the compounding process is associative as shown in Chapter 3. The number of comparisons in the CTS 2.0 algorithm is bounded by the constant, $N_{rc} \times N_{sh} \times N_{map}$ instead of $N_{sh} \times N_{map}$ as in the original map location estimation, where N_{rc} represents the largest possible number of the root feature candidates of a neighboring map. This constant N_{rc} is the same as N_{sh} because the algorithm uses the shared features as the candidates of the root feature. Consequently, the CTS 2.0 algorithm still runs in $\mathcal{O}(1)$ because the running time is bounded by a constant.

It should be noted that, however, this modified algorithm can not enhance the solution equivalent to an optimal solution because there is still a possibility that the solution misses

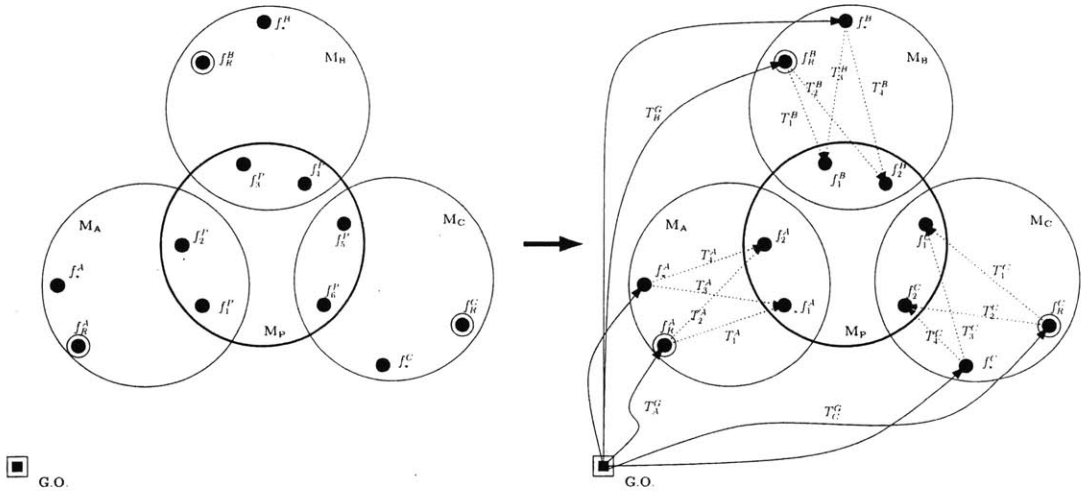

Figure 4-11: Instead of tracking only one global connection between a submap and the global origin, the CTS 2.0 maintains the global uncertainties of all the root feature candidates. Consequently, in order to choose the root feature of $\mathbf{M}_P$, the algorithm compares not only the paths via the root features of neighboring maps (f_R^A, f_R^B, and f_R^C), but also the paths through the root feature candidates of the neighboring maps ($f_\star^A$, $f_\star^B$, and $f_\star^C$). Maintaining multiple shared features reduces the possibility that the root feature found by CTS falls in a local minimum. The results are guaranteed to be better than or equal to those from CTS, because all the paths in CTS are still included in the new set of paths.

the global minimum value. In order to guarantee that the solution is the global minimum, all the possible paths from the global origin to the submap should be recalculated. However, this is computationally intractable because no greedy algorithm can be employed to this nonlinear summation of the determinants. The main contribution of this modification is improving the rate of convergence of the global poses of local maps. The performance is guaranteed to be better or equal to that of a CTS solution and the computation is still $\mathcal{O}(1)$. Later in this chapter, an experimental result will be provided to illustrate the performance of this algorithm. Figure 4-12 captures the flowchart of the CTS 2.0 algorithm.

4.3.3 Improvement of the information dissemination depth, Network Optimized SLAM (NOS)

As pointed in a previous subsection, the CTS algorithm updates only one local map at a time and this, sometimes, causes very slow improvements of local maps' pose estimations. In this subsection, we suggest an algorithm, called *Network Optimized SLAM* (NOS), that maximizes the improvement of local maps' location estimates without much loss in

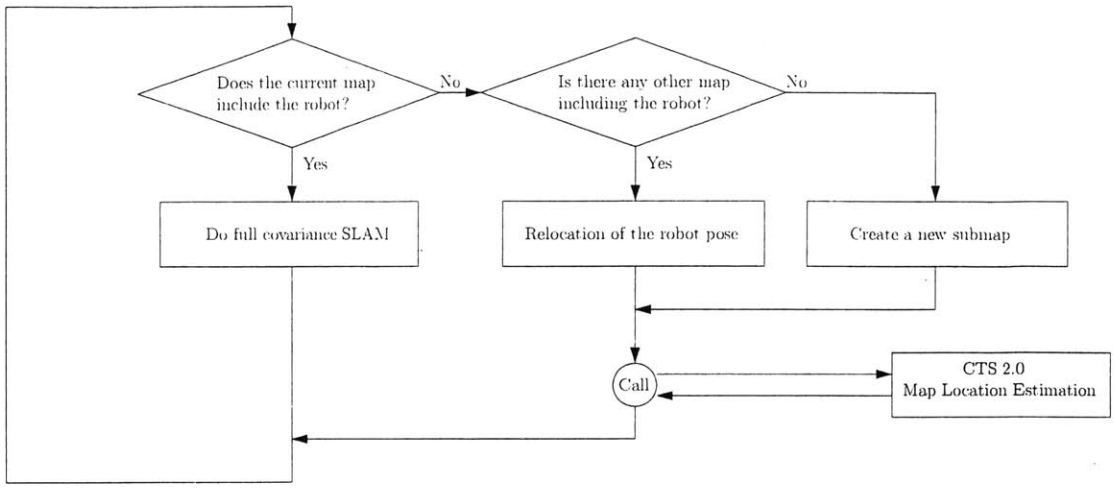

Figure 4-12: Flowchart of the CTS 2.0 algorithm.

efficiency.

To introduce the NOS algorithm more easily, let us begin with a brief description how the graph of submaps is transformed while the NOS algorithm is executed. Figure 4-10 represents the process in which the CTS algorithm improves the global location estimate of the submap $\mathbf{M}_i$. Figure 4-10(c) captures the moment that three possible paths are compared and Figure 4-10(d) illustrates that the path through $\mathbf{M}_q$ is chosen as the result of the comparison. Now, Figure 4-13(a) and 4-13(b) are re-drawn with explicit edge values. Even though the summation of the determinants does not hold the linearity, the values in the figure ignore this property for the purpose of easy illustration. The real code of the algorithm considers the nonlinearity. As shown in Figure 4-13(a), the submap $\mathbf{M}_i$ improves its global pose estimate. The Figure 4-13(a) represents the step just before the map location estimation chooses the best parent for the $\mathbf{M}_i$. The Figure 4-13(b) captures $\mathbf{M}_q$ as it is chosen as the parent map of $\mathbf{M}_i$ because the global uncertainty via $\mathbf{M}_q$ is the smallest among the three candidate parent maps $\mathbf{M}_p$, $\mathbf{M}_q$, and $\mathbf{M}_r$. This is the end stage of the map location estimation of the CTS algorithm. The new algorithm, NOS, starts from this stage and tries to disseminate the updated map's information to the other maps. In Figure 4-13(b) and 4-13(c), it can be seen that the uncertainty of $\mathbf{M}_r$ via the map $\mathbf{M}_t$ (=16) is worse than that via the $\mathbf{M}_i$ (=15) which has just been improved by the map location estimation. NOS connects $\mathbf{M}_i$ as the parent of $\mathbf{M}_r$ and eliminates the old parenthood between $\mathbf{M}_r$ and $\mathbf{M}_t$. This information flow keeps disseminating until it either reaches a leaf node($\mathbf{M}_s$)

or is blocked by a lower value node ($\mathbf{M}_p$). Consequently, NOS disseminates the updated information to the other maps as far as possible. The Figure 4-13(d) is the resultant tree after the NOS is executed. Every submap in this resultant tree has a global uncertainty lower or equal to that of the tree made by the CTS algorithm (Figure 4-13(b)).

The pseudo-code of the algorithm can be written as the following.

Algorithm *Linear-Time Network Optimized SLAM* (M_i);

1: do CTS (or CTS 2.0) ***map location estimation***
2: construct the transient graph G_{M_i}
3: do Depth First Search and disseminate the information obtained from the *map location estimation.*
4: eliminate redundant edges and construct new tree $G^*_{M_i}$.

It should be noted that the algorithm NOS is not fused with the map location estimation of the CTS (or CTS 2.0) algorithm. The NOS uses the output of the map pose estimation as its input and does not communicate with any other routines in the CTS until NOS finishes its calculations and returns the improved tree structure. This means that NOS can be performed separately from the main process of the CTS and, therefore, usually does not degrade the computation efficiency even though the NOS itself runs in $\mathcal{O}(N_m)$ where N_m is the number of the local maps. As long as the NOS algorithm can complete its calculation before another map update occurs, the total computation time of the whole SLAM solution is maintained as $\mathcal{O}(1)$. Figure 4-14 represents the flowchart of the NOS algorithm.

a. Depth first search (DFS) in NOS

The NOS uses the *Depth First Search* (DFS) technique to check if a submap is eligible to be updated. It is well known that *all nodes of a graph can be reached by DFS if the graph is connected* [22]. Though the searching technique of NOS slightly differs from that of a conventional DFS, the same reasoning applies to the NOS algorithm; that is, every node in the graph is checked for its eligibility. The following is a brief proof of that property.

> *Proof.* Every node is directly or indirectly examined by DFS in NOS. DFS leaves no node unchecked if the graph is connected. Let the graph for DFS starting from $\mathbf{M}_i$ be $G_{DFS}(i)$. The graph is connected and the nodes and edges are defined as

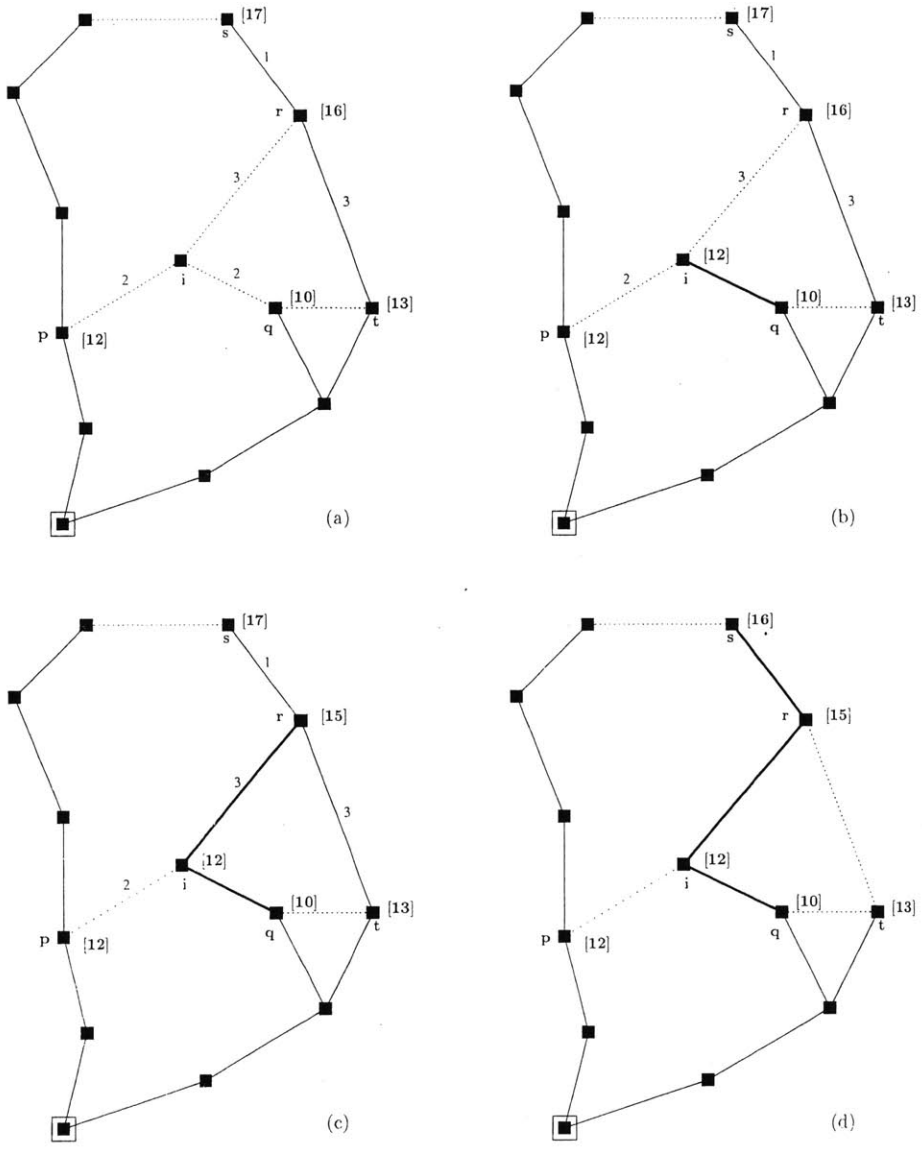

Figure 4-13: Figure (a) represents the map location estimation process in CTS. $\mathbf{M}_i$ is considered to be improved. The CTS algorithm compares three possible paths to the submap (via $\mathbf{M}_p$, $\mathbf{M}_q$, and $\mathbf{M}_r$) and chooses $\mathbf{M}_q$ as its parent map because the resultant uncertainty is the smallest among the three. (b) Once this selection process is finished, the CTS algorithm ignores the other paths via $\mathbf{M}_p$ and $\mathbf{M}_r$ until another map location estimation for $\mathbf{M}_i$ occurs. It should be noted, however, that $\mathbf{M}_r$ can be improved if $\mathbf{M}_r$ replaces its parent with $\mathbf{M}_i$. The resultant uncertainty (=15) is less than the current value (=16). This new relationships can be disseminated until no improvement can exist. Figure (c) and (d) illustrates the resultant tree structure after this improvement dissemination.

$$N = \left\{ \mathbf{M}_i \;\middle|\; i = 1, \cdots, n, \quad \text{where n is the number of submaps.} \right\}$$
$$E = E_T \cup \left\{ (i,j) \;\middle|\; \text{if } \mathbf{M}_j \text{ is a neighbor of } \mathbf{M}_i. \right\} \tag{4.23}$$

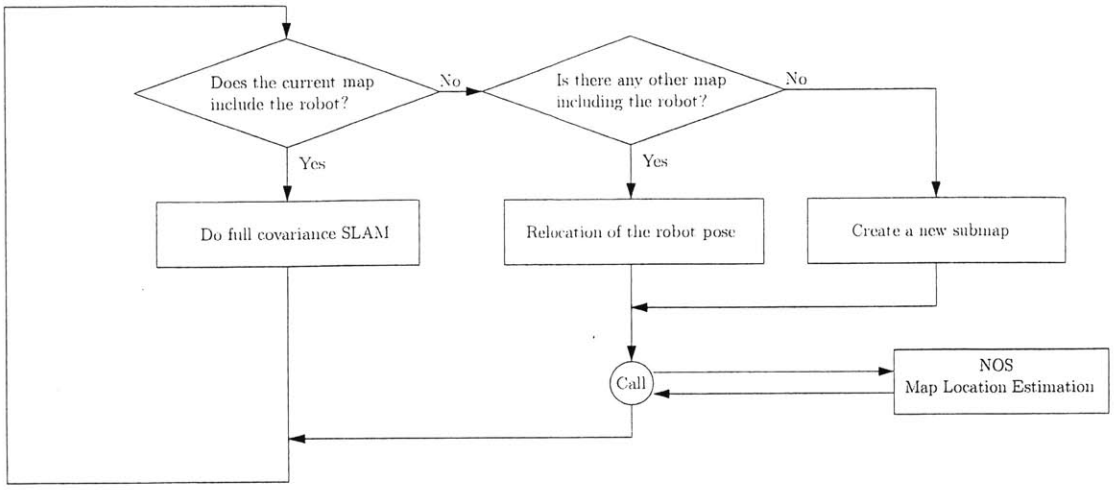

Figure 4-14: Flowchart of the NOS algorithm.

where E_T is the tree defined in (4.19). We know that every node of $G_{DFS}(i)$ is in a branch of the original tree. The branch can be classified into two groups: (1) a branch can be reached from the node i without passing the global root or (2) a branch cannot be reached unless the global root is passed through. First, it is not necessary to examine any node of the branches in the second group (branch 1 in Figure 4-15). Without passing through the global root, the updated information cannot be transferred to that node and the global root, having zero uncertainty, always resets the updated information while passing the root.

Thus, it suffices to show that every node of the branches in the first group (branch 2 in Figure 4-15) is examined during DFS in NOS. Let us examine one neighboring map (v in the figure) of M_i. The node always separates the branch containing v into two links. One is the link from the global root to the parent node of v (ancestor link in the figure) and the other link is from the child of v to the leaf of the branch if v is not a leaf (predecessor link in the figure). All nodes in the predecessor link are visited during DFS and updated if the new information is better than the old value. This information updates flow up to a leaf because the uncertainty of a child is always bigger than that of its parent. The nodes in the ancestor link from the parent of v to the global origin are examined if the parent-child relationship should be reversed. That is, if the

uncertainty of node v's parent is larger than that of v, the relation is reversed so that v is now the parent (See the right sub-figure of 4-15). This inversion occurs until a parent's uncertainty still beats its child's. In that case, all of the ancestor nodes also have less uncertainties than the child node, therefore, is not necessary for it to be examined. Consequently, all nodes in the original tree are examined either directly or indirectly.

□

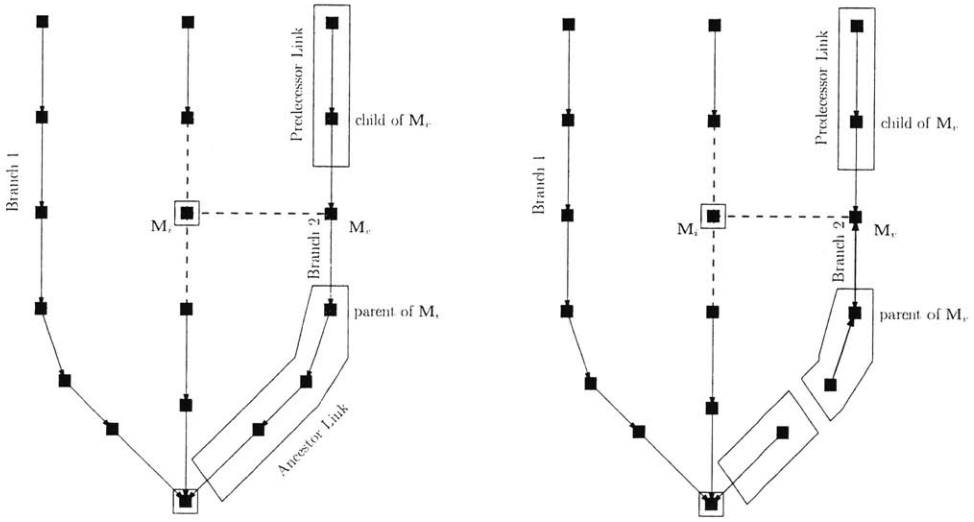

Figure 4-15: When $\mathbf{M}_i$ is improved, every node of the submap network is examined either directly or indirectly through DFS in the NOS algorithm. Every path between the $\mathbf{M}_i$ and the nodes in the type of branch 1 includes the global origin node. Therefore, the nodes in the type of branch 1 are not improved by NOS. The uncertainty values of nodes in the type of branch 2, however, are compared and updated through DFS of NOS algorithm.

b. The running time of NOS

It is well known that the running time of DFS is $\mathcal{O}(N+E)$ where N and E are the number of nodes and edges of a graph [22]. This results because the time required to call DFS routine is $\mathcal{O}(N)$ and the total number of checking whether the search goes deeper is $\sum_{v \in N} |Adj(v)|$.

The modified DFS in NOS process, however, runs in $\mathcal{O}(N_m)$ because the number of edges in a tree is $\mathcal{O}(N_m)$. Furthermore, as already described in the above proof, many nodes are excluded from the checking, so DFS in NOS runs much faster in most cases. Also, since the

process is executed independently from the main SLAM process, the running time does not degrade the running time of CTS until the number of submaps exceeds the product of the number of features in a map and the total time steps spent in one map. That is, in order to maintain CTS as a constant time algorithm, it suffices to complete one NOS process while the robot stays in the submap.

4.4 Global Uncertainty of the Whole Map

An important issue in considering a large-scale SLAM algorithm is to compare its performance to the full covariance solution. This section examines the global uncertainty of the whole map that is constructed through the union of all submaps, and shows that the CTS based algorithms search the best root feature not only the determinant but also the each traces of submaps.

All three of the algorithms described in this chapter — CTS, CTS 2.0, and NOS — follow a common strategy for estimating the global pose of features in a given local submap. The goal is to find a best root feature for each submap so that the global uncertainty of the submap is minimized. The performances of these algorithms can be compared to that of the full covariance SLAM solution by taking the corresponding sub-block of the covariance matrix that the full SLAM solution produces.

What is the global uncertainty of all features with respect to the global origin? That is, if all features are globally referenced in a single state vector, what is the determinant value of the corresponding covariance matrix? When we perform full covariance SLAM, the global uncertainty is merely the determinant of the covariance matrix because every feature in the survey region is estimated in one state vector. In CTS, however, we need to combine the locally processed information to determine the global uncertainty. For a local map, we have examined how to transform the local uncertainty to the global frame. As described in the previous subsections, each uncertainty is a product of (1) the uncertainty of the global pose of the local frame and (2) the uncertainty of the features in the local map with respect to the local origin. This relationship is demonstrated through

$$\left|\mathbf{P}^{G}_{M_i}\right| = \left|\mathbf{P}^{G}_{root_i}\right| \left|\mathbf{P}^{root_i}_{M_i}\right| \quad . \tag{4.24}$$

From the above equation and the fact that any root shifting between local features does not change the second determinant, minimizing the first quantity is the only way to reduce the product in Equation 4.24. The routines, map location estimation in CTS and network optimized SLAM, focus on this minimization. The goal of these algorithms is to provide more accurate global pose estimates for each individual local map.

However, in this subsection we want to calculate the global uncertainty of the entire region from the fragments of information obtained in local maps. Surprisingly, the result is simply the product of the determinants of every local map. That is,

$$\left|\mathbf{P}^{G}\right| = \prod_{i=0}^{N} \left|\mathbf{P}_{M_i}^{root_i}\right| \quad . \tag{4.25}$$

For convenience, we verify this result through one example. Although the map connections are slightly different according to which algorithm (CTS, CTS 2.0, or NOS) is chosen, they do not affect the result. Thus, we use simple map connections from the original CTS configuration, as shown in Figure 4-16. In this figure, there are thirteen features spread across five local maps. The first map ($\mathbf{M}_0$) has the global origin and two features (a and c). Therefore, when the robot is in $\mathbf{M}_0$, the state vector and the covariance matrix have the forms $\mathbf{x}_0 = [\mathbf{x}_{oa}\ \mathbf{x}_{oc}]^T$ and $\mathbf{P}_0 = \begin{bmatrix} \mathbf{P}_{oa,oa} & \mathbf{P}_{oa,oc} \\ \mathbf{P}_{oc,oa} & \mathbf{P}_{oc,oc} \end{bmatrix}$.

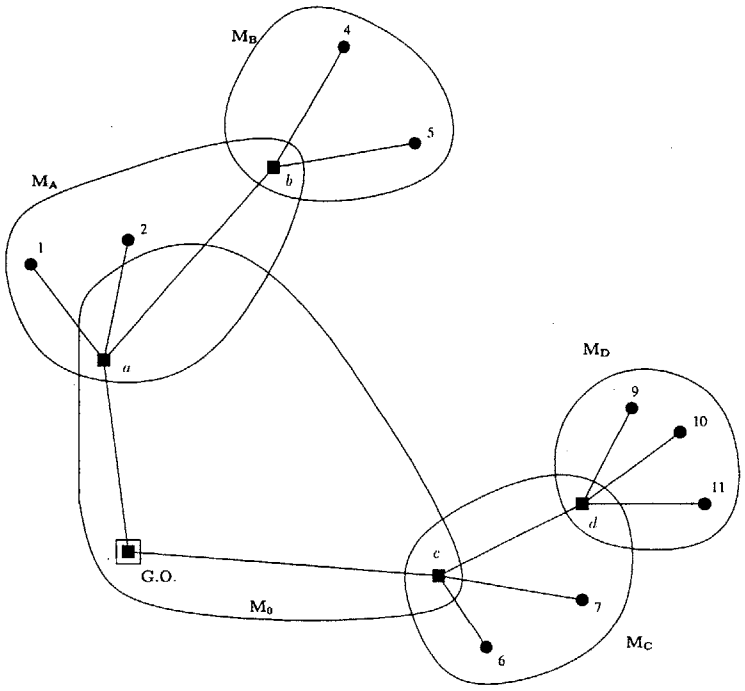

Figure 4-16: A typical network configuration for a CTS type algorithm is illustrated. There are five submaps including the first submap $\mathbf{M}_0$. Submaps are connected through the root feature and form a tree structure.

The second map ($\mathbf{M}_A$) is rooted at the feature a and has three features in it. The corresponding state vector and the covariance matrix of $\mathbf{M}_A$ are $\mathbf{x}_A = [\mathbf{x}_{a1}\ \mathbf{x}_{a2}\ \mathbf{x}_{a3}]^T$ and $\mathbf{P}_A = \begin{bmatrix} \mathbf{P}_{a1,a1} & \mathbf{P}_{a1,a2} & \mathbf{P}_{a1,a3} \\ \mathbf{P}_{a2,a1} & \mathbf{P}_{a2,a2} & \mathbf{P}_{a2,a3} \\ \mathbf{P}_{a3,a1} & \mathbf{P}_{a3,a2} & \mathbf{P}_{a3,a3} \end{bmatrix}$. The third map ($\mathbf{M}_B$) has one feature that is shared with $\mathbf{M}_A$ and chooses it as the root feature b. The features in $\mathbf{M}_B$ are referenced with respect to this local origin and are estimated in the form of $\mathbf{x}_B = [\mathbf{x}_{b4}\ \mathbf{x}_{b5}]^T$ and $\mathbf{P}_B = \begin{bmatrix} \mathbf{P}_{b4,b4} & \mathbf{P}_{b4,b5} \\ \mathbf{P}_{b5,b4} & \mathbf{P}_{b5,b5} \end{bmatrix}$. Feature b in map $\mathbf{M}_B$ is the same as feature number 3 in $\mathbf{M}_A$. Similarly, the maps $\mathbf{M}_C$ and $\mathbf{M}_D$ are written as:

$$\mathbf{M}_C : \mathbf{x}_C = \begin{bmatrix} \mathbf{x}_{c6} \\ \mathbf{x}_{c7} \\ \mathbf{x}_{c8}(=\mathbf{x}_{cd}) \end{bmatrix}, \quad \mathbf{P}_C = \begin{bmatrix} \mathbf{P}_{c6,c6} & \mathbf{P}_{c6,c7} & \mathbf{P}_{c6,c8} \\ \mathbf{P}_{c7,c6} & \mathbf{P}_{c7,c7} & \mathbf{P}_{c7,c8} \\ \mathbf{P}_{c8,c6} & \mathbf{P}_{c8,c7} & \mathbf{P}_{c8,c8} \end{bmatrix}$$

$$\mathbf{M}_D : \mathbf{x}_D = \begin{bmatrix} \mathbf{x}_{d9} \\ \mathbf{x}_{d10} \\ \mathbf{x}_{d11} \end{bmatrix}, \quad \mathbf{P}_D = \begin{bmatrix} \mathbf{P}_{d9,d9} & \mathbf{P}_{d9,d10} & \mathbf{P}_{d9,d11} \\ \mathbf{P}_{d10,d9} & \mathbf{P}_{d10,d10} & \mathbf{P}_{d10,d11} \\ \mathbf{P}_{d11,d9} & \mathbf{P}_{d11,d10} & \mathbf{P}_{d11,d11} \end{bmatrix}. \tag{4.26}$$

Now, let us combine the local information in these maps to compute the global uncertainty. The combination can be interpreted in terms of the following transformation.

$$\begin{bmatrix} \mathbf{x}_{oa} & \mathbf{x}_{oc} & \mathbf{x}_{a1} & \mathbf{x}_{a2} & \mathbf{x}_{a3} & \mathbf{x}_{b4} & \mathbf{x}_{b5} & \mathbf{x}_{c6} & \mathbf{x}_{c7} & \mathbf{x}_{c8} & \mathbf{x}_{d9} & \mathbf{x}_{d10} & \mathbf{x}_{d11} \end{bmatrix}^T$$
$$\Rightarrow \begin{bmatrix} \mathbf{x}_{oa} & \mathbf{x}_{oc} & \mathbf{x}_{o1} & \mathbf{x}_{o2} & \mathbf{x}_{o3} & \mathbf{x}_{o4} & \mathbf{x}_{o5} & \mathbf{x}_{o6} & \mathbf{x}_{o7} & \mathbf{x}_{o8} & \mathbf{x}_{o9} & \mathbf{x}_{o10} & \mathbf{x}_{o11} \end{bmatrix}^T \tag{4.27}$$

We should note that it is always possible to stack up the submaps in the order in which every parent is located above its predecessors since the feature connections form a tree that is always a directed acyclic graph. (See the topological sort alogorithm in [22].)

This transformation is done by series of compounding operations whose inputs are the feature itself and several ancestors. That is,

$$\begin{bmatrix} \mathbf{x}_{oa} \\ \mathbf{x}_{oc} \\ \mathbf{x}_{o1} \\ \mathbf{x}_{o2} \\ \mathbf{x}_{o3}(=\mathbf{x}_{oc}) \\ \mathbf{x}_{o4} \\ \mathbf{x}_{o5} \\ \mathbf{x}_{o6} \\ \mathbf{x}_{o7} \\ \mathbf{x}_{o8}(=\mathbf{x}_{od}) \\ \mathbf{x}_{o9} \\ \mathbf{x}_{o10} \\ \mathbf{x}_{o11} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{oa} \\ \mathbf{x}_{oc} \\ \mathbf{x}_{oa} \oplus \mathbf{x}_{a1} \\ \mathbf{x}_{oa} \oplus \mathbf{x}_{a2} \\ \mathbf{x}_{oa} \oplus \mathbf{x}_{ab}(=\mathbf{x}_{a3}) \\ \mathbf{x}_{oa} \oplus \mathbf{x}_{ab}(=\mathbf{x}_{a3}) \oplus \mathbf{x}_{b4} \\ \mathbf{x}_{oa} \oplus \mathbf{x}_{ab}(=\mathbf{x}_{a3}) \oplus \mathbf{x}_{b5} \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{c6} \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{c7} \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{c8}(=\mathbf{x}_{c8}) \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{cd}(=\mathbf{x}_{c8}) \oplus \mathbf{x}_{d9} \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{cd}(=\mathbf{x}_{c8}) \oplus \mathbf{x}_{d10} \\ \mathbf{x}_{oc} \oplus \mathbf{x}_{cd}(=\mathbf{x}_{c8}) \oplus \mathbf{x}_{d11} \end{bmatrix} . \tag{4.28}$$

Now, the globally referenced uncertainty is calculated through the proper Jacobian matrix. If the environment is not a linear Gaussian case, the value becomes the first order approximation of the uncertainty. In the following equations, $J_{o1|oa}$ means the Jacobian of transformation for $\mathbf{x}_{o1}$ with respect to $\mathbf{x}_{oa}$, that is, $\frac{\partial(\mathbf{x}_{oa} \oplus \mathbf{x}_{a1})}{\partial \mathbf{x}_{oa}}$. As shown in the equation, the Jacobian matrix forms a triangular matrix. This occurs because every element in the state vector can be calculated from the element itself and its ancestors, and the special tree structure of the CTS submap system makes this ordering possible at all times. It is well known that the determinant of a block triangular matrix is just the product of the determinants of diagonal blocks [41]. Also, the input covariance matrix forms a block diagonal matrix because the cross correlations between local maps are not maintained by the CTS in order to maintain the consistency. Therefore, the determinant of the original covariance matrix is also the product of the determinants of each local covariance matrix. Finally, since we know that the determinant of a Jacobian block element is always 1, the resultant determinant of the quadratic form can be written in the following form.

$$\mathbf{P}_G \approx \mathbf{J}\ \mathbf{P}_L\ \mathbf{J}^T$$

$$= \begin{bmatrix}
I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
J_{o1|oa} & 0 & J_{o1|a1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
J_{o2|oa} & 0 & 0 & J_{o2|a2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
J_{o3|oa} & 0 & 0 & 0 & J_{o3|a3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
J_{o4|oa} & 0 & 0 & 0 & J_{o4|a3} & J_{o4|b4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
J_{o5|oa} & 0 & 0 & 0 & J_{o5|a3} & 0 & J_{o5|b5} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & J_{o6|oc} & 0 & 0 & 0 & 0 & 0 & J_{o6|c6} & 0 & 0 & 0 & 0 & 0 \\
0 & J_{o7|oc} & 0 & 0 & 0 & 0 & 0 & 0 & J_{o7|c7} & 0 & 0 & 0 & 0 \\
0 & J_{o8|oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J_{o8|c8} & 0 & 0 & 0 \\
0 & J_{o9|oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J_{o9|c8} & J_{o9|d9} & 0 & 0 \\
0 & J_{o10|oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J_{o10|c8} & 0 & J_{o10|d10} & 0 \\
0 & J_{o11|oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J_{o11|c8} & 0 & 0 & J_{o11|d11}
\end{bmatrix}$$

$$\begin{bmatrix}
P_{oa,oa} & P_{oa,oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_{oc,oa} & P_{ob,oc} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & P_{a1,a1} & P_{a1,a2} & P_{a1,a3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & P_{a2,a1} & P_{a2,a2} & P_{a2,a3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & P_{a3,a1} & P_{a3,a2} & P_{a3,a3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & P_{b4,b4} & P_{b4,b5} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & P_{b5,b4} & P_{b5,b5} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{c6,c6} & P_{c6,c7} & P_{c6,c8} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{c7,c6} & P_{c7,c7} & P_{c7,c8} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{c8,c6} & P_{c8,c7} & P_{c8,c8} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{d9,d9} & P_{d9,d10} & P_{d9,d11} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{d10,d9} & P_{d10,d10} & P_{d10,d11} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & P_{d11,d9} & P_{d11,d10} & P_{d11,d11}
\end{bmatrix}$$

$$\begin{bmatrix}
I & 0 & J^T_{o1|oa} & J^T_{o2|oa} & J^T_{o3|oa} & J^T_{o4|oa} & J^T_{o5|oa} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & I & 0 & 0 & 0 & 0 & 0 & J^T_{o6|oc} & J^T_{o7|oc} & J^T_{o8|oc} & J^T_{o9|oc} & J^T_{o10|oc} & J^T_{o11|oc} \\
0 & 0 & J^T_{o1|a1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & J^T_{o2|a2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & J^T_{o3|a3} & J^T_{o4|a3} & J^T_{o5|a3} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & J^T_{o4|b4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & J^T_{o5|b5} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o6|c6} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o7|c7} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o8|c8} & J^T_{o9|c8} & J^T_{o10|c8} & J^T_{o11|c8} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o9|d9} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o10|d10} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & J^T_{o11|d11}
\end{bmatrix} \tag{4.29}$$

$$
\begin{aligned}
|\mathbf{P}_G| &= \left|\mathbf{J}\ \mathbf{P}_L\ \mathbf{J}^T\right| \\
&= |\mathbf{J}| \cdot |\mathbf{P}_L| \cdot \left|\mathbf{J}^T\right| \\
&= \left|\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{J}_{\oplus 1} & \mathbf{J}_{\oplus 2} \end{bmatrix}\right| \cdot \left|\begin{bmatrix} \mathbf{P}_0 & \mathbf{0} \\ \mathbf{0} & \begin{matrix} \mathbf{P}_A & 0 & 0 & 0 \\ 0 & \mathbf{P}_B & 0 & 0 \\ 0 & 0 & \mathbf{P}_C & 0 \\ 0 & 0 & 0 & \mathbf{P}_D \end{matrix} \end{bmatrix}\right| \cdot \left|\begin{bmatrix} \mathbf{I} & {\mathbf{J}_{\oplus 1}}^T \\ \mathbf{0} & {\mathbf{J}_{\oplus 2}}^T \end{bmatrix}\right| \\
&= |\mathbf{I}|\,|\mathbf{J}_{\oplus 2}| \cdot |\mathbf{P}_0|\,|\mathbf{P}_A|\,|\mathbf{P}_B|\,|\mathbf{P}_C|\,|\mathbf{P}_D| \cdot |\mathbf{I}|\,\left|{\mathbf{J}_{\oplus 2}}^T\right| \\
&= |\mathbf{P}_0|\,|\mathbf{P}_A|\,|\mathbf{P}_B|\,|\mathbf{P}_C|\,|\mathbf{P}_D| \quad .
\end{aligned}
\tag{4.30}
$$

Therefore, the global uncertainty of the entire region only depends on the determinant of each individual submap [10]. The form does not include any information concerning how the local maps are connected and which features are chosen as their local origins. Also, we can conclude that minimizing individual local uncertainties is the only way to maximize the quality of the global estimates if the estimation goal is to minimize the determinant.

Of course, this property does not degrade the importance of CTS based algorithms. Even though the global uncertainty (the determinant of the covariance matrix) of the entire region does not depend on which features are chosen as the root features, the trace (variances) of the global covariance matrix still depends on the qualities of the root features; That is, the CTS based algorithms provide a solution that minimizes the determinant as well as the trace. Figure 4-17 illustrates the effect of the root feature quality on the global map. Suppose, as shown in the figure, that there exist three features, and two features are shared by two submaps. Figure (a) shows the global map with the uncertainties of each feature when feature 1 is chosen as the root feature of submap B. Figure (b) illustrates the global map when feature 2, whose uncertainty is larger than feature 1's, is chosen as the root feature of submap B. As described, the resultant determinants of the whole state are the same. The uncertainty of each feature, however, in (a) differs from that of each feature in (b). Consequently, we can conclude that the CTS based algorithms search the best root feature that minimizes not only the determinant but also each variances.

[10] Again, this property relies on linearization and the assumption of a two-dimensional environment.

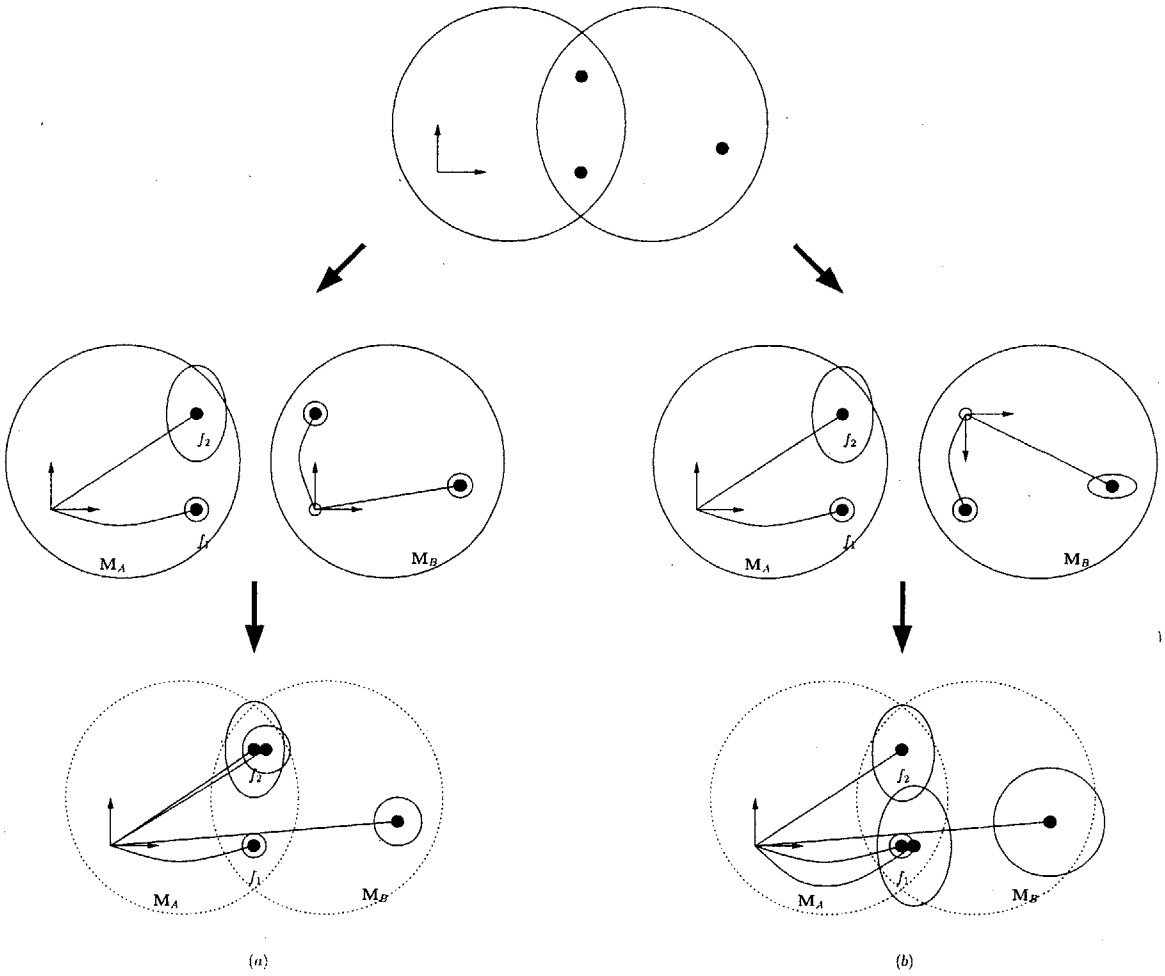

Figure 4-17: Though the determinant of the global state does not depend on which features are chosen as the root features, the variances (traces) of the global covariance matrix still depend on the qualities of the root features.

4.5 Experimental Results

In this section, the performances of all the described algorithms is compared through a postprocessing result of a challenging data set.

4.5.1 Experiment configuration

Four tennis courts in the middle of the indoor track (See Figure 4-23) are used for the environment of the experiment because the lines of the courts can be utilized as a good *ground-truth* of the features' locations. For each tennis courts, 16 hurdles are set along the line of the court. Figure 4-24 and Figure 4-19 illustrate the location setting of the hurdles. In the experiment, two poles of the hurdle define a point feature with an angle; that is, one pole defines the x and y positions and the angle is determined through the relative location of the other pole. Figure 4-18 represents how the point feature with an angle is defined in the experiment.

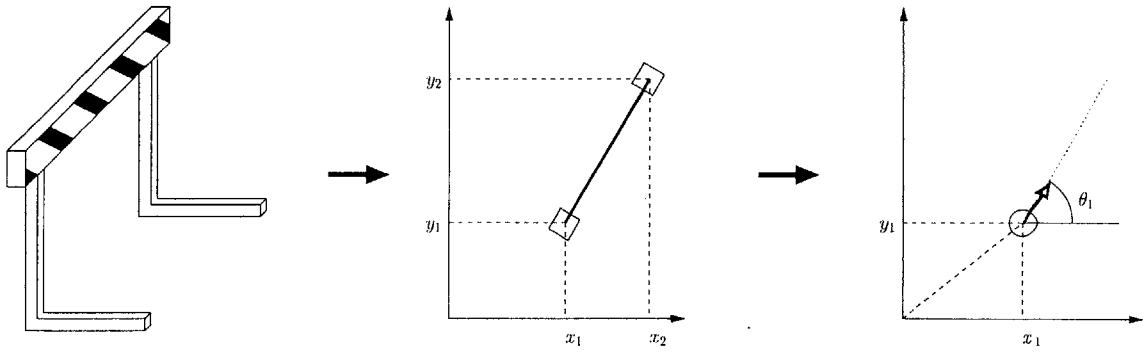

Figure 4-18: Two poles of the hurdle define a point feature that has three parameters: x position, y position and the angle.

Since the distance between the poles of a hurdle is known, this hurdle feature environment facilitates the data association. The data set, however, is still very challenging because the *dead reckoning* (DR) performance of the B21r robot is quite poor as shown in Figure 4-20.

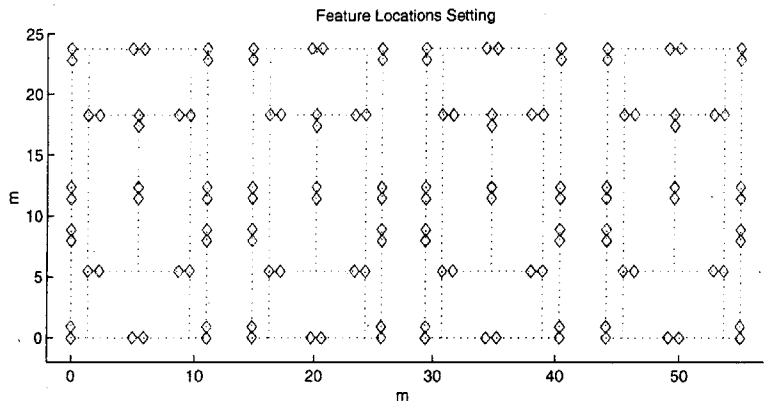

Figure 4-19: 64 hurdles are located along the lines of four tennis courts.

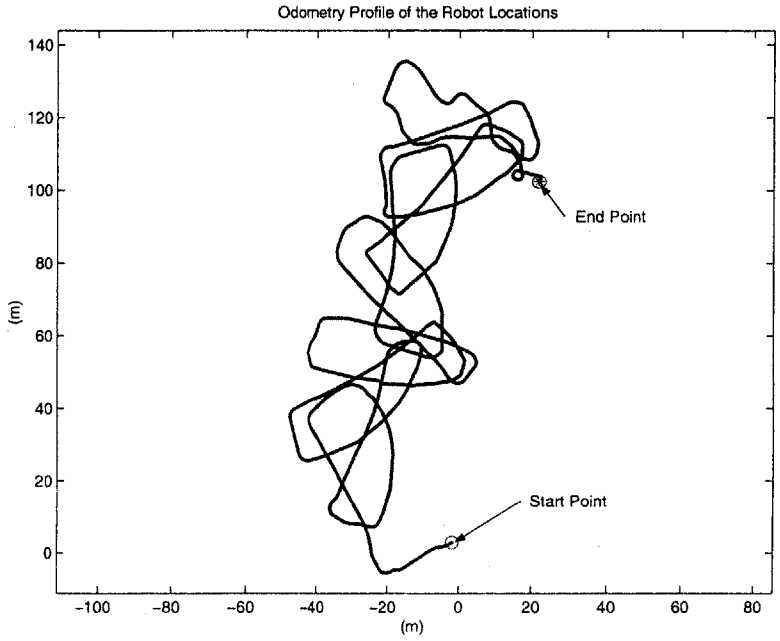

Figure 4-20: Dead reckoned robot trajectory shows the large bias error of the odometry.

The dynamic model and the measurement model for the experiment are defined as

- dynamic model

$$\begin{aligned} x_v[k+1] &= x_v[k] - u[k] \cdot \delta T \cdot \sin(\phi_v[k]) \\ y_v[k+1] &= y_v[k] + u[k] \cdot \delta T \cdot \cos(\phi_v[k]) \\ \phi_v[k+1] &= \phi_v[k] + \dot{\phi}[k] \cdot \delta T \end{aligned} \tag{4.31}$$

- measurement model

$$\begin{aligned} z_x[k] &= \quad (x_f[k] - x_v[k])\cos(\phi_v[k]) + (y_f[k] - y_v[k])\sin(\phi_v[k]) \\ z_y[k] &= -(x_f[k] - x_v[k])\sin(\phi_v[k]) + (y_f[k] - y_v[k])\cos(\phi_v[k]) \\ z_\theta[k] &= \theta_f[k] - \phi_v[k] \end{aligned} \tag{4.32}$$

Figure 4-21 and Figure 4-22 illustrate the definitions. It should be noted that the positive heading angle of the robot is defined as being counter-clockwise from the y axis. The robot navigates by two control inputs, the forward velocity $u[k]$ and the angular velocity $\dot{\phi}[k]$, and the measurements are obtained with respect to the robot's current pose. From the dynamic model, the corresponding prediction step of the SLAM is written in the form:

$$P_{vv}[k+1|k] = F_v[k]\, P_{vv}[k|k]\, F_v[k]^T + G_v[k]\, Q_v[k]\, G_v[k]^T \tag{4.33}$$

where the Jacobians are defined

$$\begin{aligned} F_v[k] &= \begin{bmatrix} 1 & 0 & -u[k]\cdot\delta T\cdot\cos(\phi_v[k]) \\ 0 & 1 & -u[k]\cdot\delta T\cdot\sin(\phi_v[k]) \\ 0 & 0 & 1 \end{bmatrix} \\ G_v[k] &= \begin{bmatrix} -\delta T\cdot\sin(\phi_v[k]) & 0 \\ \delta T\cdot\cos(\phi_v[k]) & 0 \\ 0 & \delta T \end{bmatrix} . \end{aligned} \tag{4.34}$$

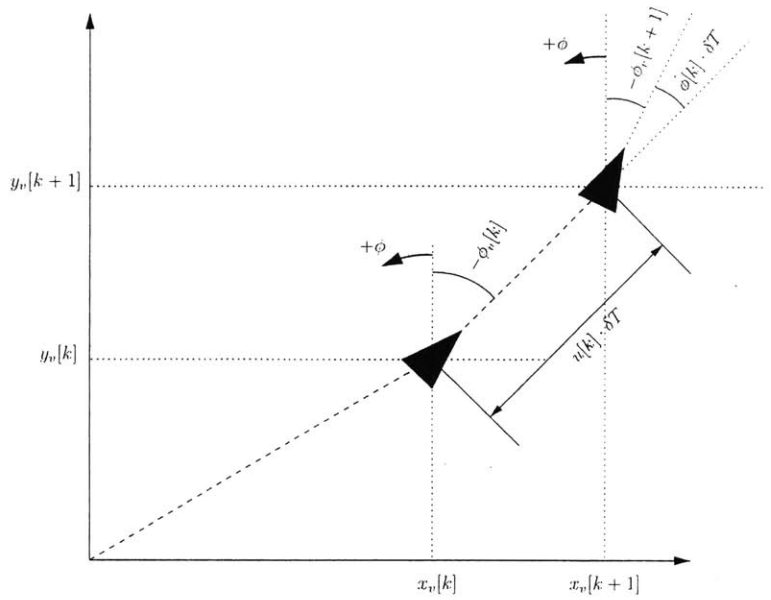

Figure 4-21: Dynamic model of the Johnson Center experiment. The positive heading angle of the robot is defined as being counter-clockwise from the y axis.

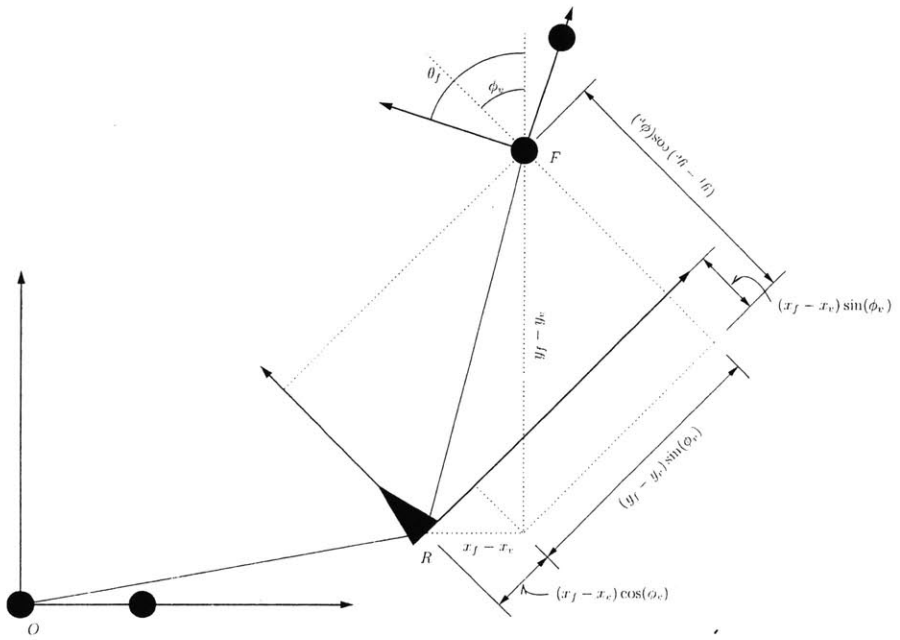

Figure 4-22: Measurement model of the Johnson Center experiment.

Similarly, the observation matrix $H[k+1]$ is constructed when the i-th feature is observed,

$$\mathbf{H}[k+1] = \begin{bmatrix} H_v & | & \cdots & H_I & \cdots \end{bmatrix} \tag{4.35}$$

where the robot block H_v and the observed feature block H_I are defined as

$$H_v = \begin{bmatrix} -\cos(\phi_v[k]) & -\sin(\phi_v[k]) & -(x_f[k]-x_v[k])\sin(\phi_v[k]) + (y_f[k]-y_v[k])\cos(\phi_v[k])) \\ \sin(\phi_v[k]) & -\cos(\phi_v[k]) & -(x_f[k]-x_v[k])\cos(\phi_v[k]) - (y_f[k]-y_v[k])\sin(\phi_v[k])) \\ 0 & 0 & -1 \end{bmatrix}$$

$$H_I = \begin{bmatrix} \cos(\phi_v[k]) & \sin(\phi_v[k]) & 0 \\ -\sin(\phi_v[k]) & \cos(\phi_v[k]) & 0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{4.36}$$

Finally, whenever the robot detects a new feature, the feature initialization is executed through

$$\mathbf{P}_{new} = \begin{bmatrix} \mathbf{P}_{old} & (G_x \mathbf{P}_{old})^T \\ (G_x \mathbf{P}_{old}) & G_x \mathbf{P}_{old} G_x^T + G_z \mathbf{R} G_z^T \end{bmatrix} \tag{4.37}$$

where G_x and G_z are

$$G_x = \begin{bmatrix} 1 & 0 & -x_1 \sin(\phi_v[k]) - y_1 \cos(\phi_v[k]) \\ 0 & 1 & x_1 \cos(\phi_v[k]) - y_1 \sin(\phi_v[k]) \\ 0 & 0 & 1 \end{bmatrix}$$

$$G_z = \begin{bmatrix} \cos(\phi_v[k]) & -\sin(\phi_v[k]) & 0 & 0 \\ \sin(\phi_v[k]) & \cos(\phi_v[k]) & 0 & 0 \\ \frac{\frac{y_2-y_1}{(x_2-x_1)^2}}{1+\left(\frac{y_2-y_1}{x_2-x_1}\right)^2} & \frac{-\frac{x_2-x_1}{(x_2-x_1)^2}}{1+\left(\frac{y_2-y_1}{x_2-x_1}\right)^2} & \frac{-\frac{y_2-y_1}{(x_2-x_1)^2}}{1+\left(\frac{y_2-y_1}{x_2-x_1}\right)^2} & \frac{\frac{x_2-x_1}{(x_2-x_1)^2}}{1+\left(\frac{y_2-y_1}{x_2-x_1}\right)^2} \end{bmatrix} . \tag{4.38}$$

The total time duration of the experiments is 39000 time steps for about one hour, and during which the robot closes the big loop 8 times.

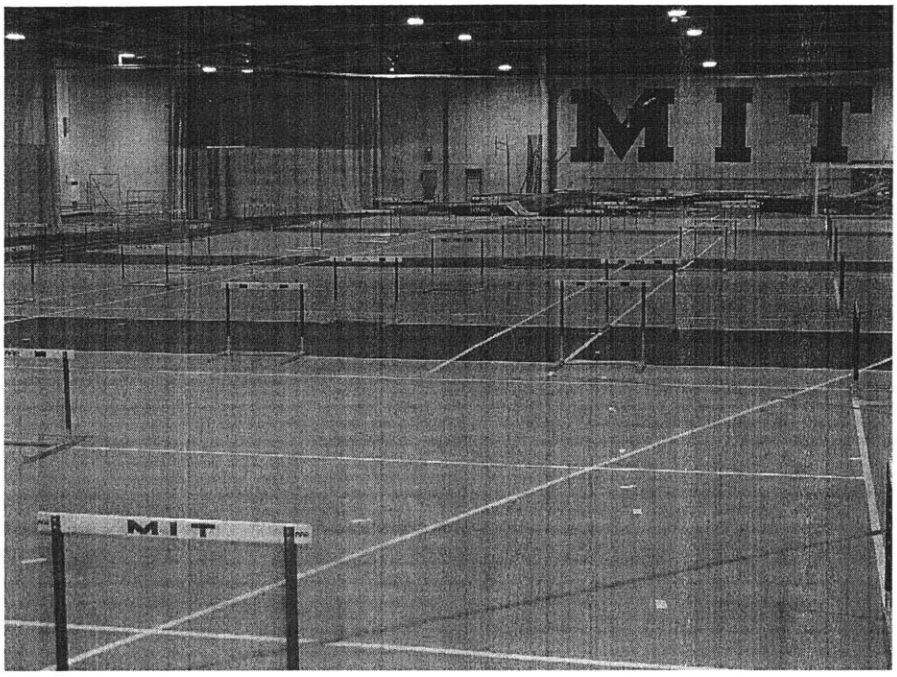

Figure 4-23: Johnson Center experiment.

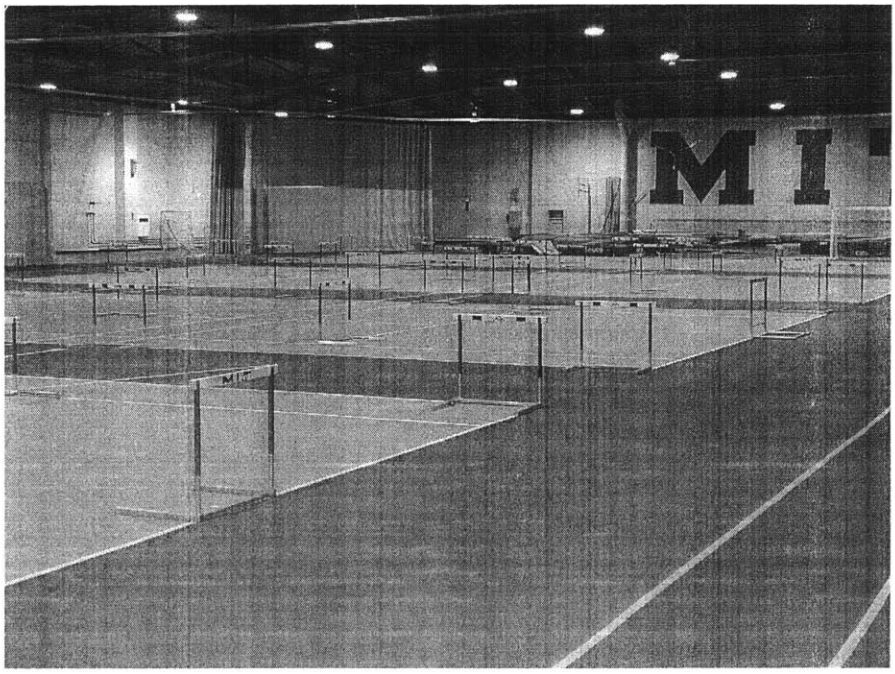

Figure 4-24: Johnson Center experiment.

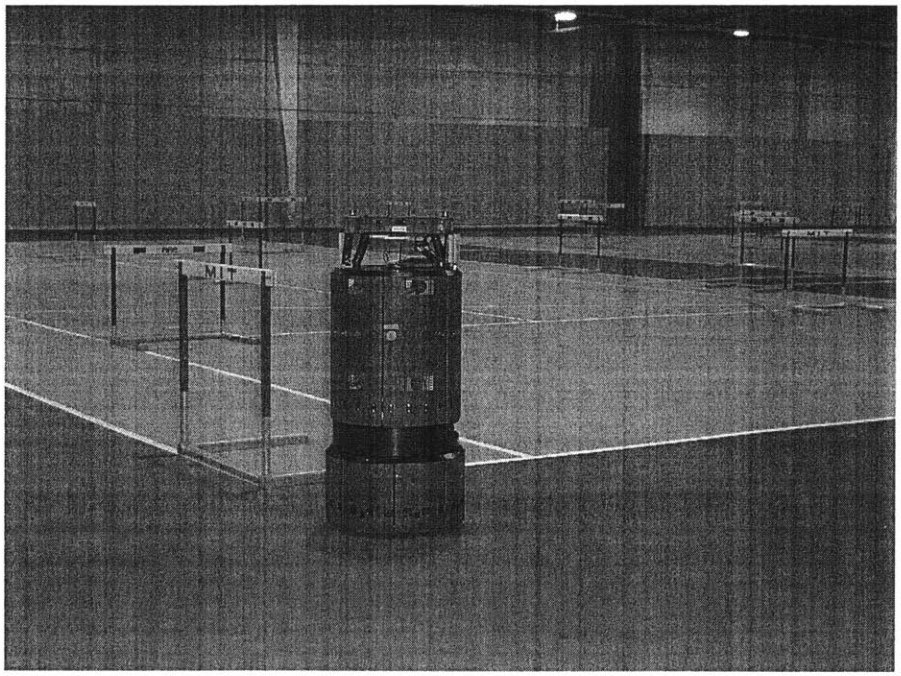

Figure 4-25: Johnson Center experiment.

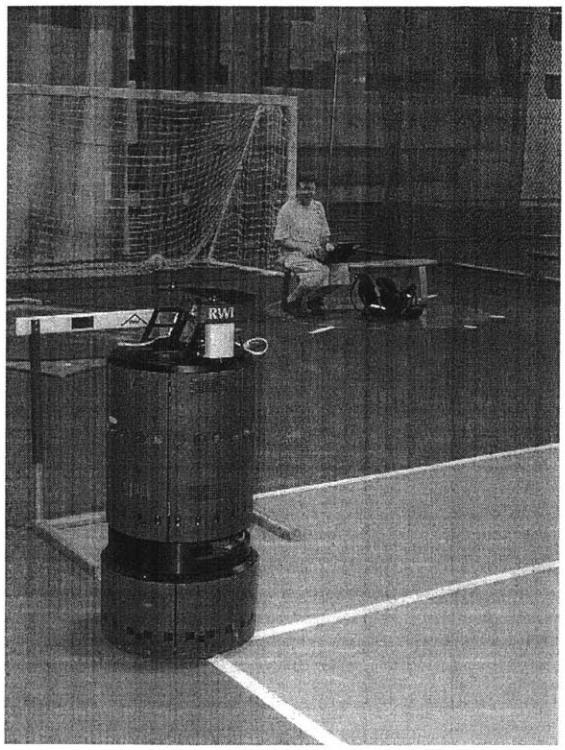

Figure 4-26: Johnson Center experiment.

4.5.2 Results and analysis

For the purpose of the performance comparisons, four algorithms are examined: (1) the basic CTS algorithm, (2) the CTS 2.0 algorithm, (3) the NOS algorithm, and (4) the full covariance SLAM algorithm.

a. The global uncertainties of a submap in time

First, the global uncertainties of each submap are compared in time. The result plots are shown in figures (4-27 through 4-32). The lower graph of each page represents the re-scaled version of the upper graph for clear comparisons. Each unceratinty is plotted at every 1000 time steps. In this examination, it suffices to compare the global determinants of the root feature through each algorithm, because the local uncertainties are identical. To compare to the full covariance SLAM result, the same features as the root feature through NOS algorithm are chosen.

First, the graph shows that the global uncertainties monotonically decrease with time. This monotonicity does not apply to the full covariance SLAM result because the chosen feature is determined by the NOS algorithm. For example, in Figure 4-32, the uncertainty at $t = 7000$ is bigger than that of $t = 6000$, because the NOS algorithm chooses a different root feature.

The stairwise decrements of the CTS algorithm result because the map location improvement occurs only once per the robot's visit. On the contrary, the decrements of the NOS algorithm imply that the map location improvement occurs frequently.

The overall performances are ranked in the following order: (1) the full covariance SLAM, (2) the NOS algorithm, (3) the CTS 2.0 algorithm, and (4) the basic CTS algorithm.

b. The global uncertainties of all submaps at a given timestep

In Figures 4-33 to 4-42, the performances of all the submaps are compared at a given timestep. For example, Figure 4-40 shows the global uncertainties of the root features of each submap through various algorithms at timestep 20000. It can be easily seen that the performaces of the algorithms are ranked: (1) the full covariance SLAM algorithm, (2) the NOS algorithm, (3) the CTS 2.0 algorithm, and (4) the basic CTS algorithm. The graphs also show that the uncertainties of a submap are generally larger when the submap

is located from the global origin.

Until the submap networks are fully developed, all four algorithms are similar in performance (See Figure 4-33 and Figure 4-34). In these early stages, only the full covariance results differ from those of the other three algorithms. After the robot creates several maps, the performances that each algorithm provides diverge. The differences result from the nonlinearity of the summation of the determinants (Figure 4-35 and Figure 4-36), and arise from the differences in the information propagation depths (Figure 4-37 through Figure 4-42). Until the robot closes a loop, the performance of the NOS algorithm is no better than the CTS 2.0 algorithm as seen in Figure 4-35 and Figure 4-36. Once a loop is closed, however, the NOS algorithm provides better performaces than the CTS 2.0 algorithm does as seen in the figures after 6000 time steps.

These comparative results at a given timestep imply that the NOS algorithm is the best choice for any finite time mission except the full covariance SLAM algorithm.

c. The submap tree structures

Figures 4-43 through 4-60 show the tree structures of two algorithms, the basic CTS algorithm and the NOS algorithm. As described in the previous sections, these tree structures are constructed by the parent and child relationships between submaps.

In the early stage of the mission, the tree structures are identical, which explains the reason why the NOS algorithm does not provide better performance than the CTS 2.0 algorithm until the tree structures diverge. The NOS algorithm looks for a better tree structure whenever a submap is improved and begins to build a different tree as shown in Figures 4-43 and 4-44.

c. The final map location estimates

Finally, Figures 4-61 through 4-71 show the final location estimates of each submap by two algorithms, the basic CTS algorithm (red squares) and the NOS (blue stars, '*') algorithm compared to the full covariance SLAM solution (yellow circles).

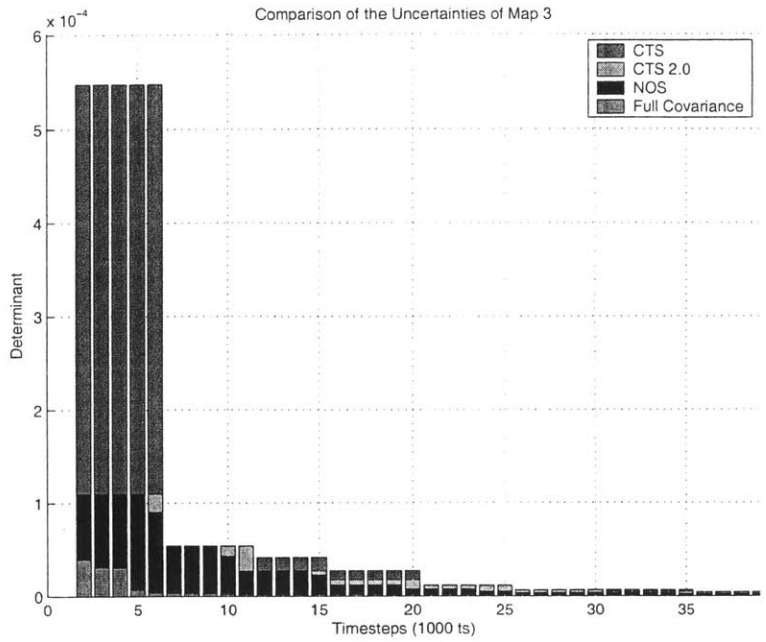

Figure 4-27: Global uncertainty profile of the root feature of the submap 3 with time.

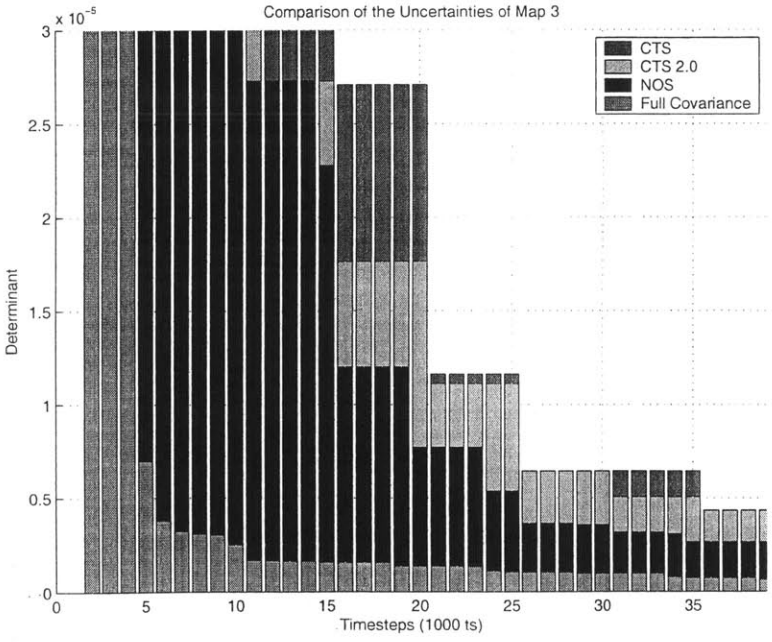

Figure 4-28: Global uncertainty profile of the root feature of the submap 3 with time.

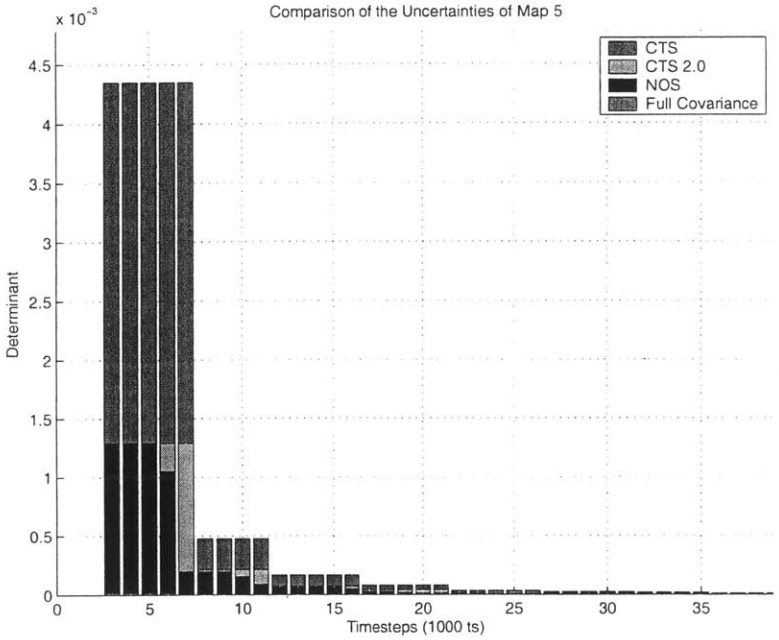

Figure 4-29: Global uncertainty profile of the root feature of the submap 5 with time.

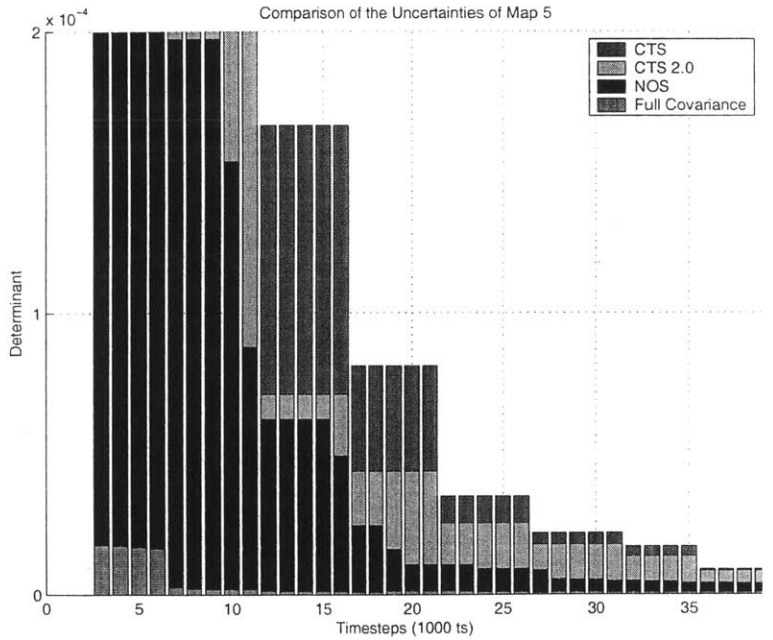

Figure 4-30: Global uncertainty profile of the root feature of the submap 5 with time.

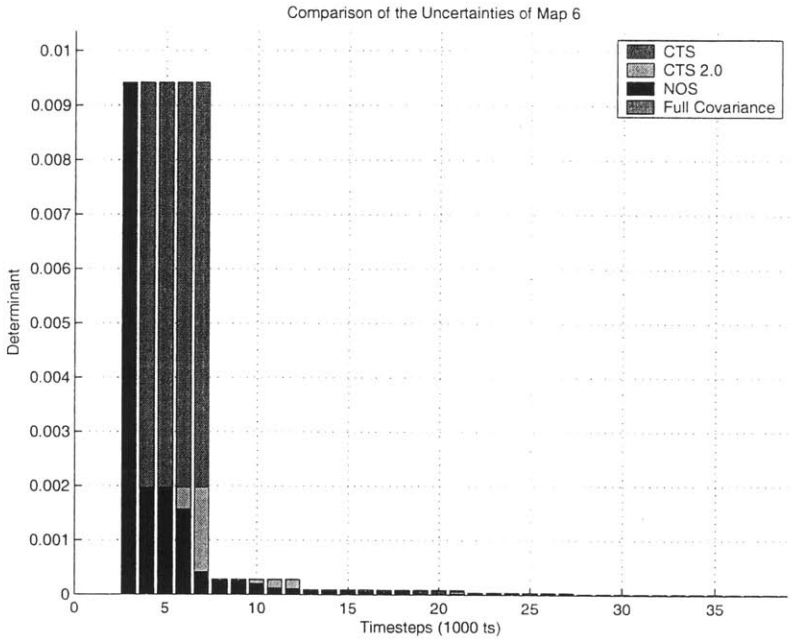

Figure 4-31: Global uncertainty profile of the root feature of the submap 6 with time.

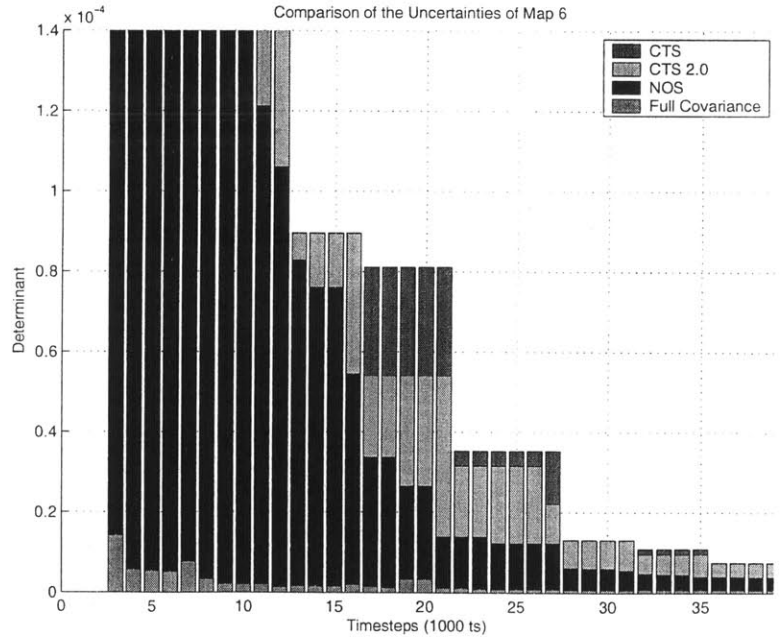

Figure 4-32: Global uncertainty profile of the root feature of the submap 6 with time.

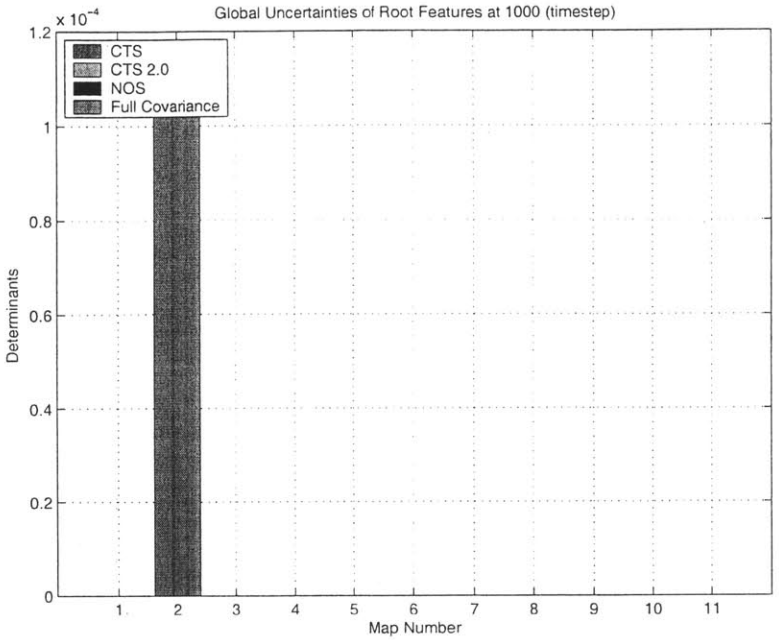

Figure 4-33: Global uncertainties of the root features at time step 1000.

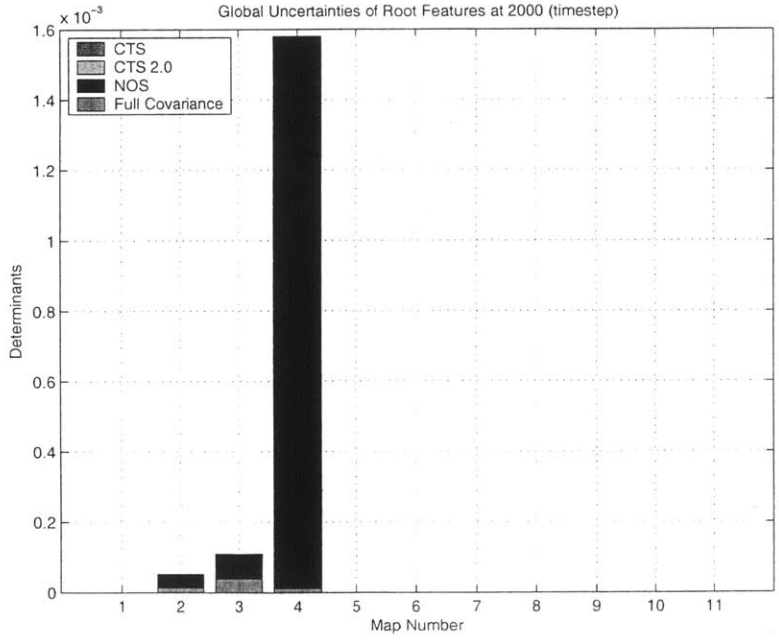

Figure 4-34: Global uncertainties of the root features at time step 2000.

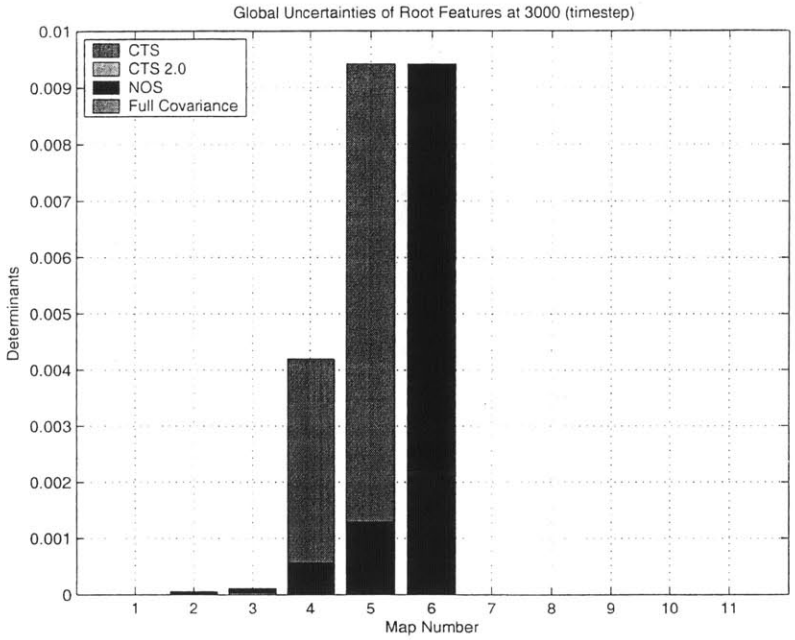

Figure 4-35: Global uncertainties of the root features at time step 3000.

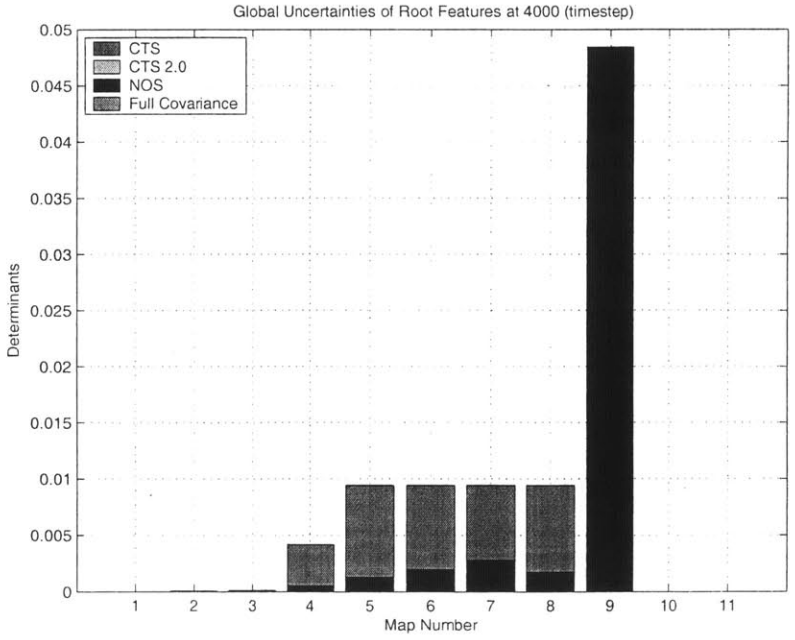

Figure 4-36: Global uncertainties of the root features at time step 4000.

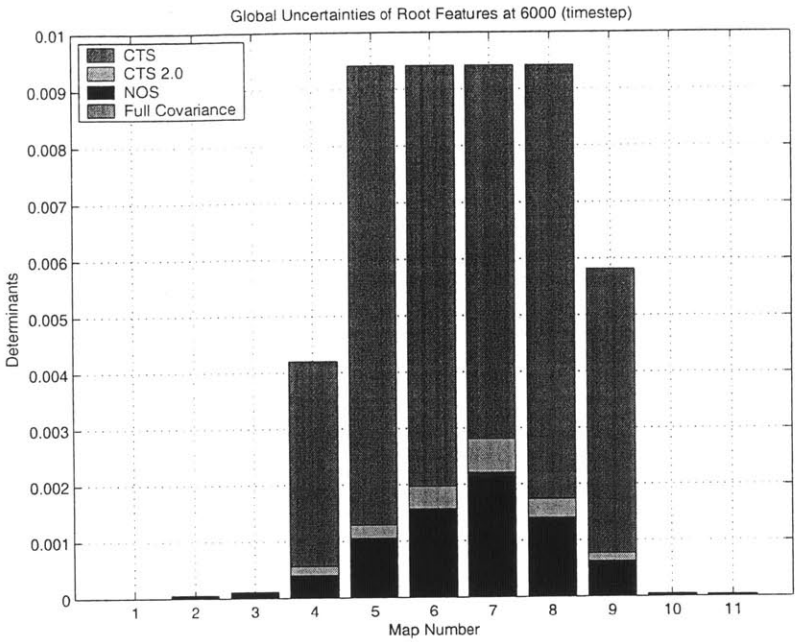

Figure 4-37: Global uncertainties of the root features at time step 6000.

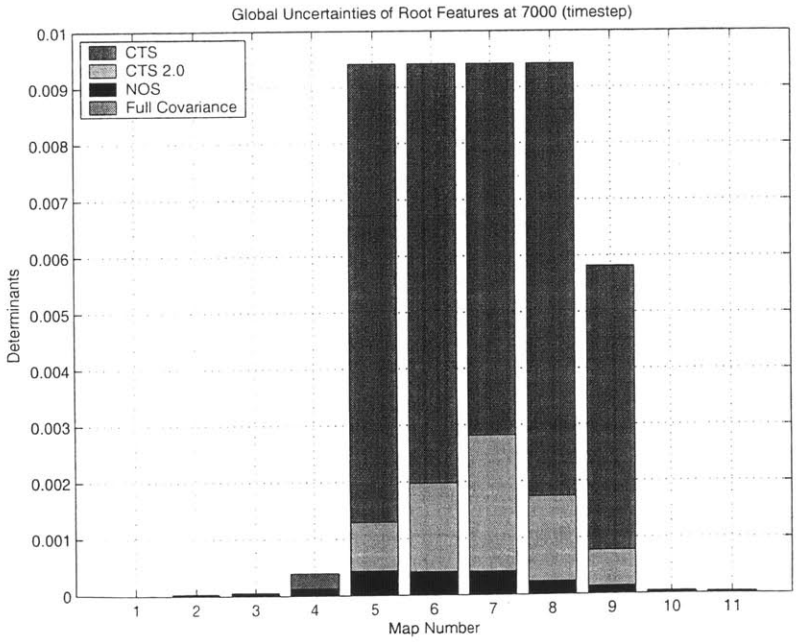

Figure 4-38: Global uncertainties of the root features at time step 7000.

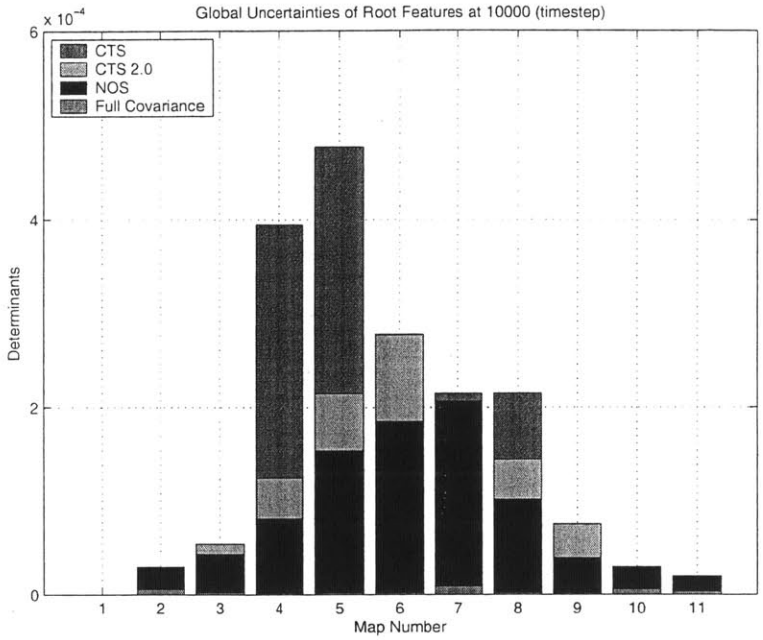

Figure 4-39: Global uncertainties of the root features at time step 10000.

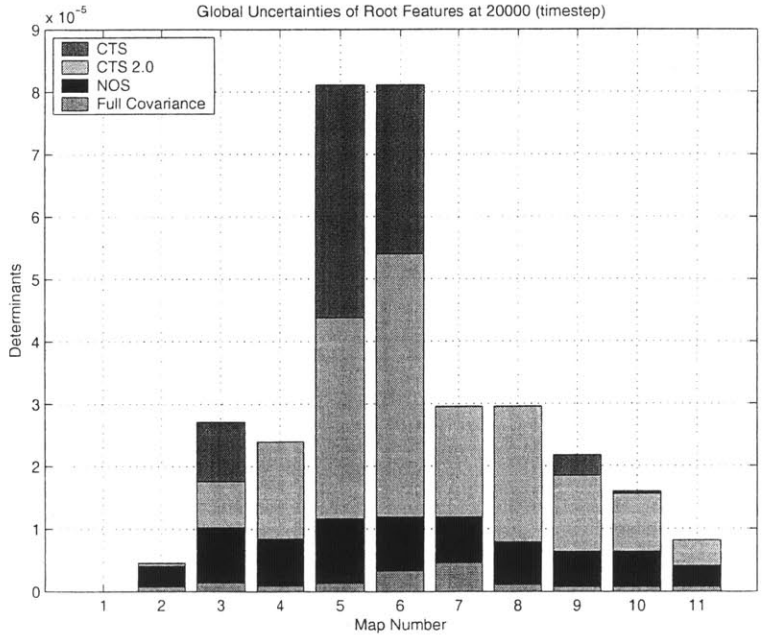

Figure 4-40: Global uncertainties of the root features at time step 20000.

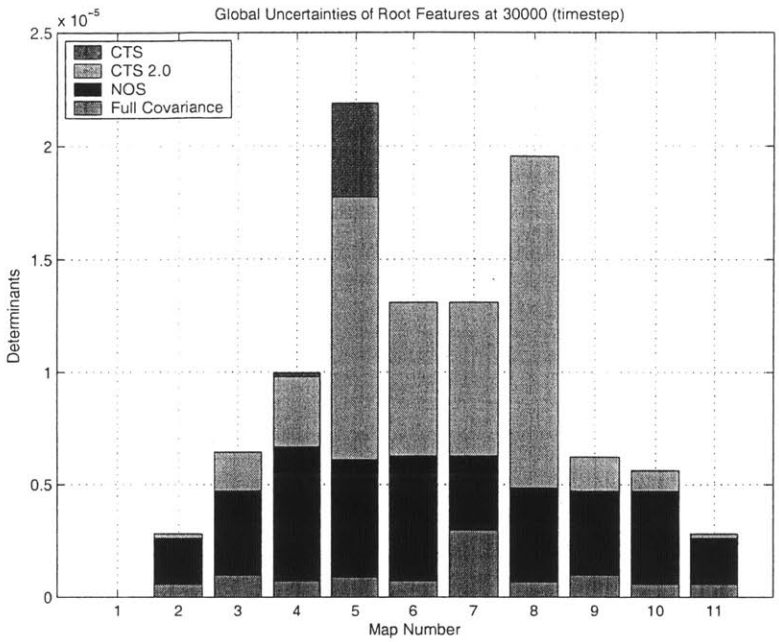

Figure 4-41: Global uncertainties of the root features at time step 30000.

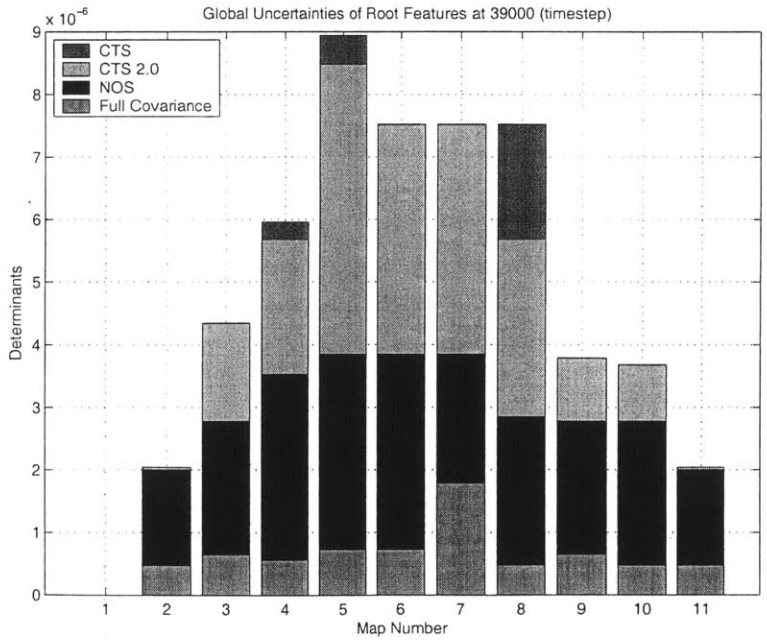

Figure 4-42: Global uncertainties of the root features at time step 39000.

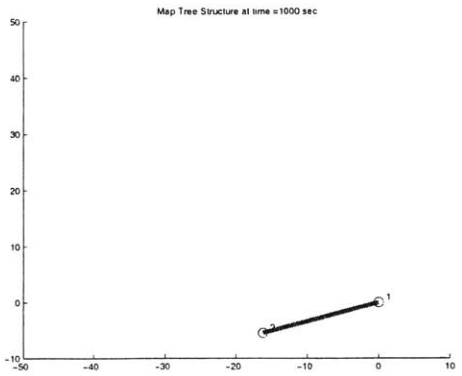

Figure 4-43: The structure of the map tree by CTS at timestep 1000.

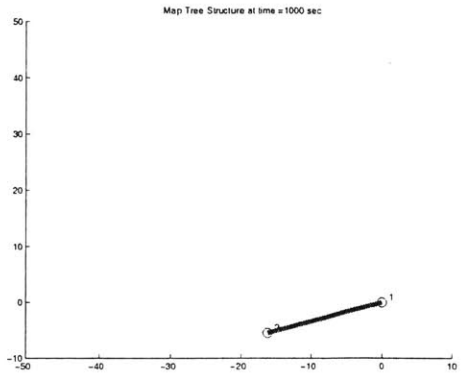

Figure 4-44: The structure of the map tree by NOS at timestep 1000.

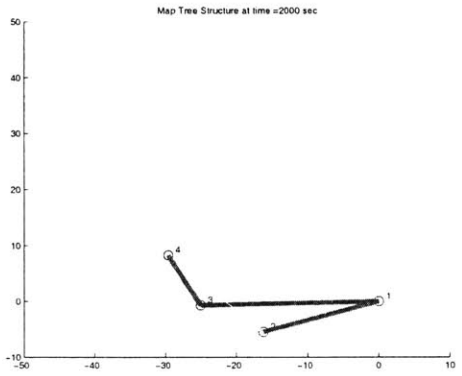

Figure 4-45: The structure of the map tree by CTS at timestep 2000.

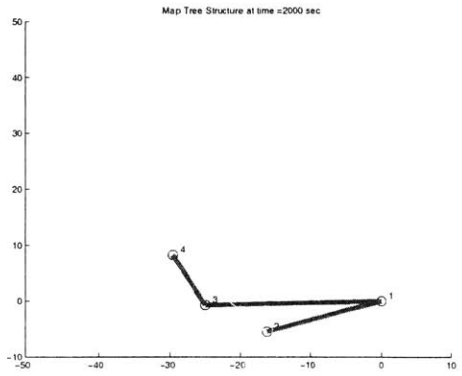

Figure 4-46: The structure of the map tree by NOS at timestep 2000.

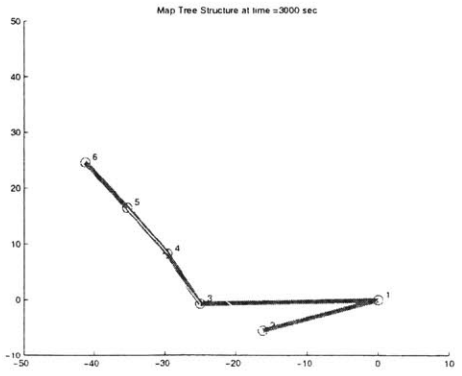

Figure 4-47: The structure of the map tree by CTS at timestep 3000.

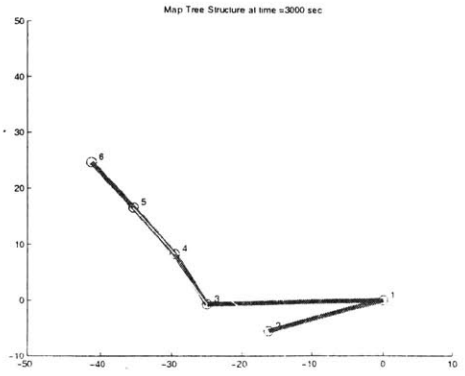

Figure 4-48: The structure of the map tree by NOS at timestep 3000.

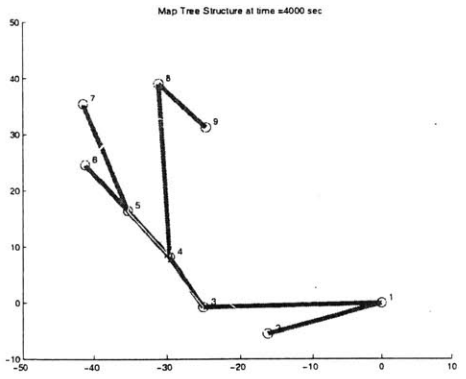

Figure 4-49: The structure of the map tree by CTS at timestep 4000.

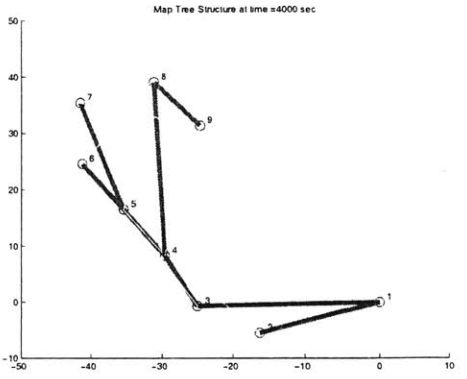

Figure 4-50: The structure of the map tree by NOS at timestep 4000.

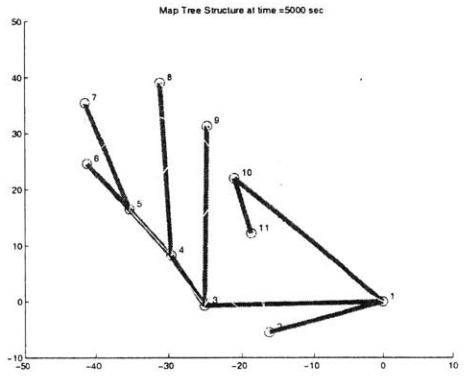

Figure 4-51: The structure of the map tree by CTS at timestep 5000.

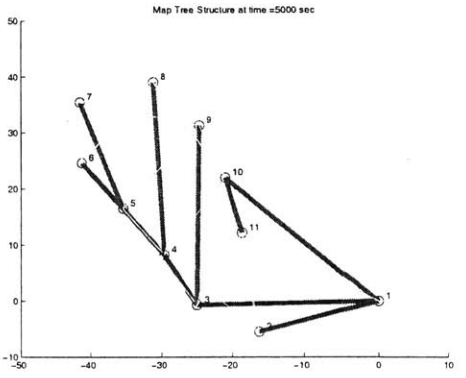

Figure 4-52: The structure of the map tree by NOS at timestep 5000.

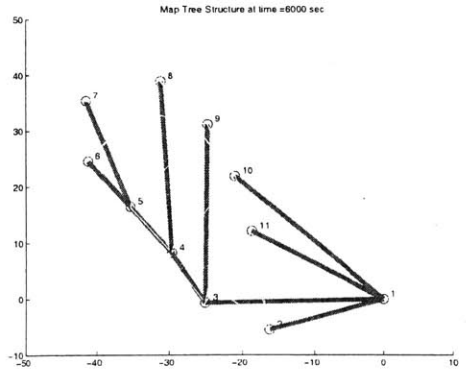

Figure 4-53: The structure of the map tree by CTS at timestep 6000.

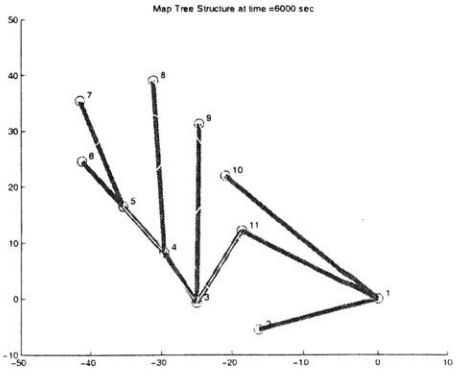

Figure 4-54: The structure of the map tree by NOS at timestep 6000.

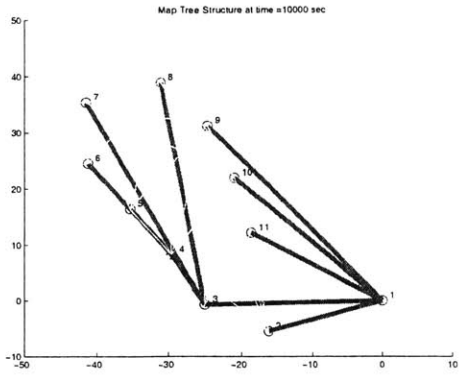

Figure 4-55: The structure of the map tree by CTS at timestep 10000.

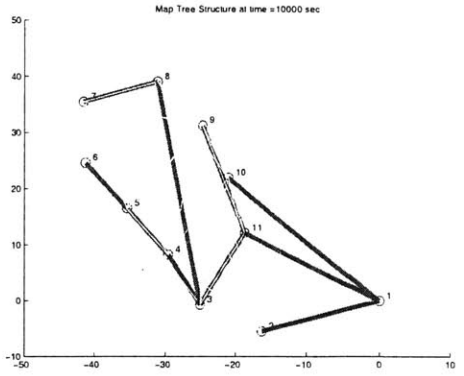

Figure 4-56: The structure of the map tree by NOS at timestep 10000.

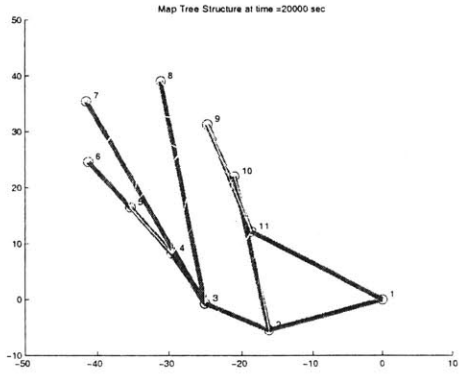

Figure 4-57: The structure of the map tree by CTS at timestep 20000.

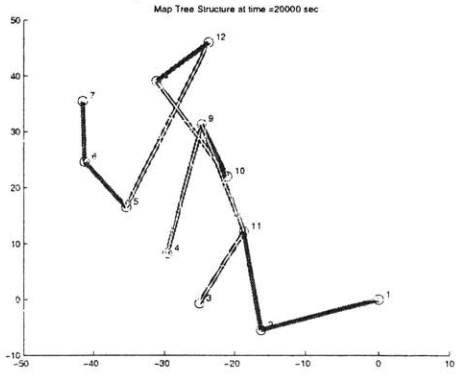

Figure 4-58: The structure of the map tree by NOS at timestep 20000.

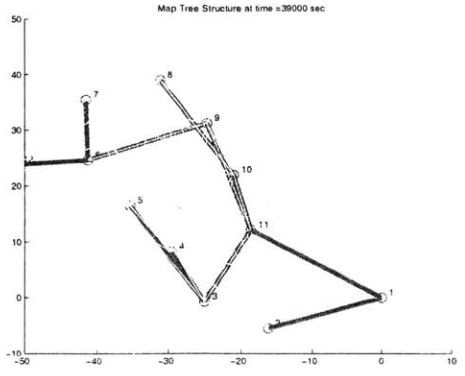

Figure 4-59: The structure of the map tree by CTS at timestep 39000.

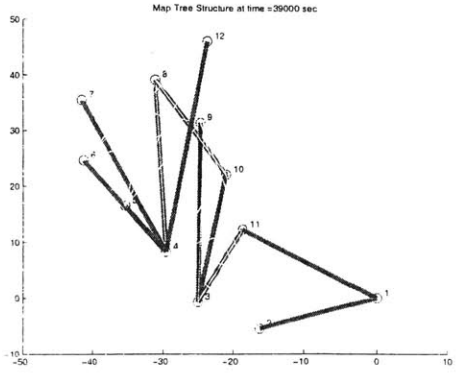

Figure 4-60: The structure of the map tree by NOS at timestep 39000.

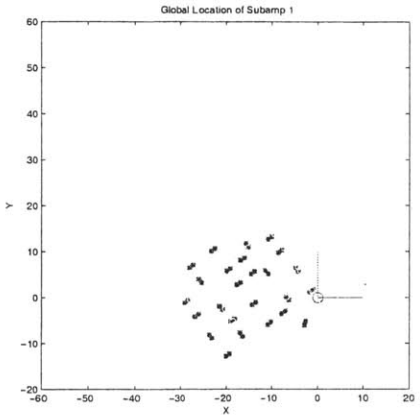

Figure 4-61: The global location estimate of submap 1.

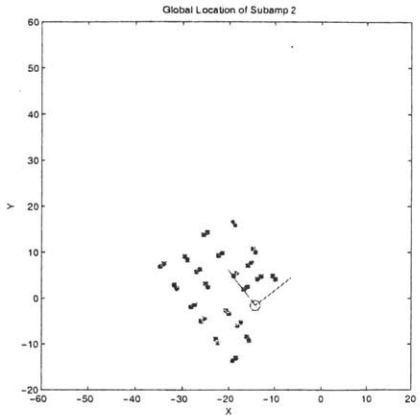

Figure 4-62: The global location estimate of submap 2.

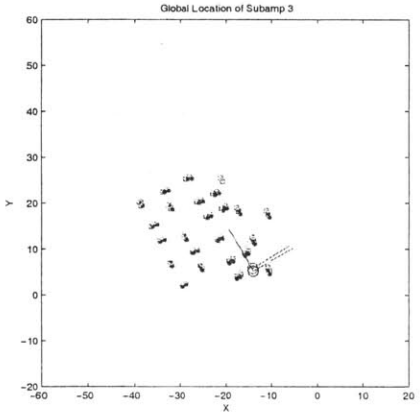

Figure 4-63: The global location estimate of submap 3.

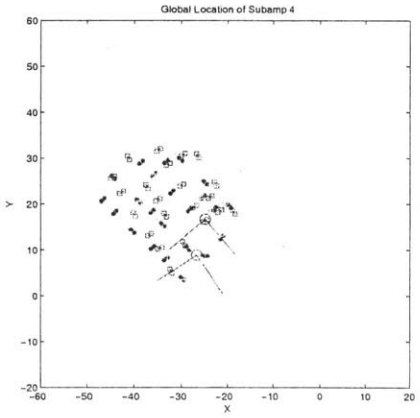

Figure 4-64: The global location estimate of submap 4.

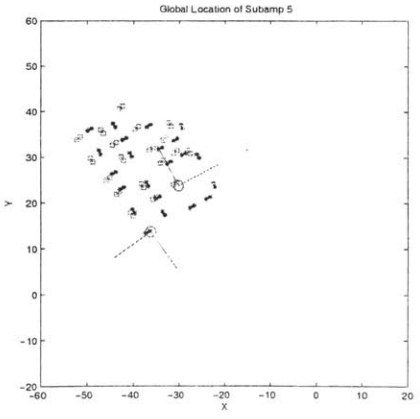

Figure 4-65: The global location estimate of submap 5.

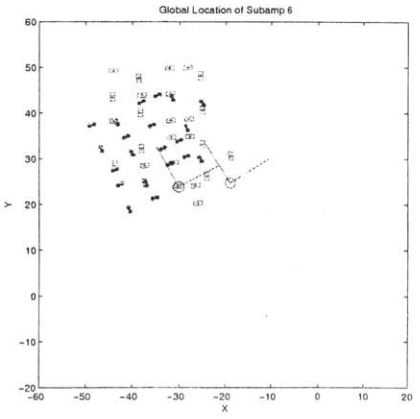

Figure 4-66: The global location estimate of submap 6.

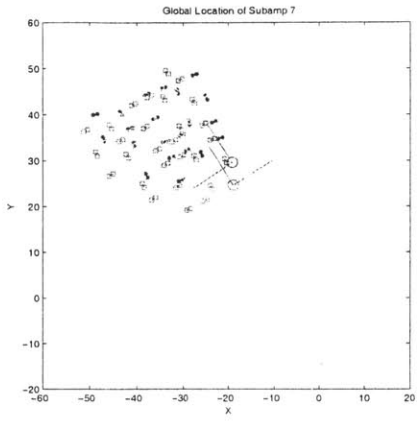

Figure 4-67: The global location estimate of submap 7.

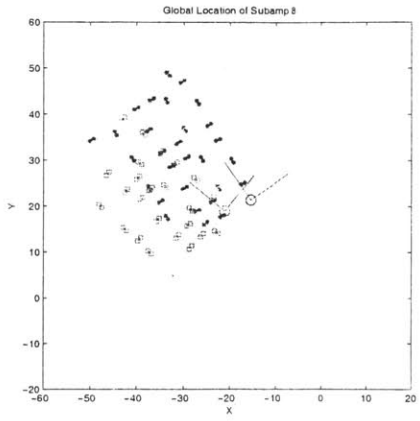

Figure 4-68: The global location estimate of submap 8.

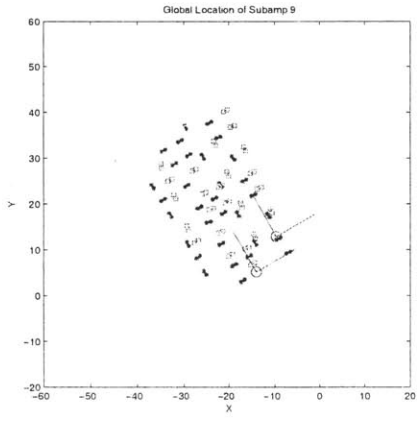

Figure 4-69: The global location estimate of submap 9.

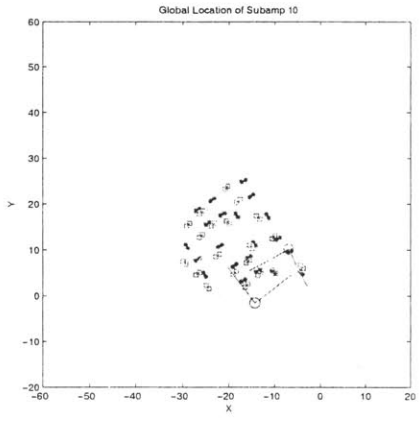

Figure 4-70: The global location estimate of submap 10.

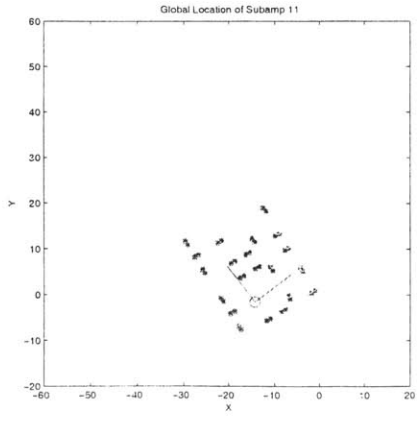

Figure 4-71: The global location estimate of submap 11.

4.6 Conclusion

In Chpater 4, we have provided analysis of the CTS algorithm in detail and improved the performances of the global uncertainties of local maps through two new algorithms using a submap network. The performances of the new algorithms have been exmained theoretically and empirically through an experiment.

The analysis of the CTS algorithm includes the tree structure, nonlinearity in the root feature selection, information dissemination depth, and the global uncertainties of all submaps. The submap network of the CTS algorithm always forms a tree structure, and the uncertainty of a child submap is always greater than or equal to that of its parent. The tree structure also supports the fact that global convergence can be achieved through local convergence in a series of submaps.

In order to improve the global location estimation of a particular submap, the standard CTS algorithm compares only the paths through the root features of neighboring maps to find the root feature. Also, this submap location estimation process for a submap occurs only once when the robot leaves the submap. These two operations for the global location estimation of a submap result in a slow converging rate in certain environments, such as a series of indoor corridors that form a large loop.

One new algorithm, termed CTS 2.0, considers the nonlinearity in summation of determinants and reduces the probability that root selection falls into a local minimum by maintaining the global pose estimates all root feature candidates. The running time of CTS 2.0 is still $\mathcal{O}(1)$ as long as the number of shared features between submaps and the number of neighboring maps are bounded by a constant (as assumed in standard CTS).

Another algorithm, termed NOS, utilizes the tree structure of the submaps and maximizes the information propagation depth in a very efficient manner. NOS algorithm constructs a more balanced tree structure than the standard CTS does. In order to propagate information to other submaps, NOS algorithm employes a modified depth first search that runs linearly in the number of submaps in worst case. However, it can be amortized and executed independently of the main process of the SLAM estimation in local maps. This is similar to the manner in which Bosse [10] performs the Dijkstra computation to propagate uncertainty through the submap network in the Atlas framework [9].

A real experiment has been implemented and a performance comparison has been pro-

vided for the four algorithms: (1) standard CTS algorithm, (2) CTS 2.0, (3) NOS, and (4) full covariance SLAM. The results shows that the global uncertainties of submaps are ranked from (4) to (1) with the full covariance SLAM solution being the best.

Also, our investigation of the global uncertainty of the whole region indicates that the determinant of the global map not depending on the root feature selections for two-dimensional SLAM. (The three-dimensional case will be investigated as future work). This result, however, does not degrade the importance of root feature selection, which still provides the best form of the covariance matrix whose trace values are minimum.

Chapter 5

Conclusion

In this chapter, we summarize the contributions of the thesis and make recommendations for future work.

5.1 Summary of Contributions

The first contribution of the thesis has been to provide a detailed analysis of the CTS algorithm, including the tree structure, nonlinearity in the root feature selection, information dissemination depth, and the global uncertainties of all submaps. Also, an experimental analysis has been performed for the nonlinear Gaussian case and the performance has been compared to other methods including the full covariance SLAM solution. In CTS, the submaps always form a tree structure and the uncertainty of a child submap is always greater than or equal to that of its parent. The tree structure also supports the fact that global convergence can be achieved through local convergence in a series of submaps.

In order to improve the global location estimate of a particular submap, the standard CTS algorithm compares only the paths through the root features of neighboring maps to find the root feature. Also, this submap location estimation process for a submap occurs only once when the robot leaves the submap. As a result of these two characteristics of CTS, the rate of convergence of CTS can be slow (in comparison to the full solution) in certain environments, such as a series of indoor corridors that form a large loop.

To speed the convergence of CTS, we have developed two new algorithms, CTS 2.0 and NOS. CTS 2.0 reduces the probability that root selection falls into a local minimum by

maintaining the global pose estimates of all root feature candidates. The performance of CTS 2.0 is always better than or equal to that of standard CTS because the root feature is one of the candidate features. CTS 2.0 runs in $\mathcal{O}(1)$ as long as the number of shared features is bounded by a constant (as assumed in standard CTS).

Network optimized SLAM utilizes the tree structure of the submaps and maximizes the information propagation depth in an efficient manner. Depth first search is employed to propagate information to other submaps. It produces a more balanced tree structure than the standard CTS algorithm does. The NOS algorithm runs linearly in the number of submaps in the worst case. However, it can be amortized and executed independently of the main process of SLAM estimation in local maps. This is similar to the manner in which Bosse [10] performs a Dijkstra computation to propagate uncertainty through the submap network in the Atlas framework [9].

An experiment with real data has been performed and a performance comparison has been provided for the four algorithms: (1) standard CTS algorithm, (2) CTS 2.0, (3) NOS, and (4) full covariance SLAM. The results show that the global uncertainties of submaps are ranked from (4) to (1) with the full covariance SLAM solution being the best.

Also, the investigation of the global uncertainty of the whole region results in the determinant of the global map does not depend on the root feature selections for two-dimensional SLAM. (The three-dimensional case will be investigated as a future work). This result, however, does not degrade the importance of root feature selection because it still provides the form of the covariance matrix whose trace values are minimized.

In addition to the study of large-scale SLAM algorithms, we investigated the properties of the covariance matrix in detail. These include: (1) non-zero terminal uncertainty, (2) the conditions for achieving a fully correlated covariance matrix, (3) exposition of the level-wise update structure in the Kalman filter based SLAM solution, and (4) the uncertainty projection during compounding and root shifting. The first two properties clarify the conditions that a SLAM solution should satisfy. The level-wise update provides clear dependencies between observed features and unobserved features during the Kalman update operation. The properties on the uncertainty projection are effectively used to analyze the CTS algorithm and to develop the CTS 2.0 and NOS algorithms in Chapter 4.

In order to examine the time behavior of the elements of the covariance matrix during a SLAM mission, a closed form solution for the one-dimensional (MonoRob) SLAM problem

has been provided. The solution applies for arbitrary initial conditions of the covariance matrix and partial observations (not all features observed at all time steps). These closed form solutions were then utilized to further investigate the properties of the covariance matrix identified above.

5.2 Future Work

5.2.1 The nonlinear SLAM problem

Currently, the submap network structure does not have any constraint related to the Gaussian assumption. This means that the current edge value, the determinant of a local covariance block, can be replaced with any metric as long as it measures the uncertainty between submaps. Also, because the global location estimation of a local submap is processed independently of the main SLAM solution, any estimation method can be employed without conflicting with the main part of CTS.

According to the resultant map location estimation, the accuracy of the pose estimation of a root feature is critical for a good global location estimation of its corresponding submap.

It is desirable to apply a nonlinear estimation technique such as the particle filter for the pose estimation of root feature.

5.2.2 Implementation with undersea sonar data

One clear important goal for future work is to implement the methods developed in this thesis on autonomous underwater vehicles (AUVs) performing large-scale mapping with undersea sonar sensors. In Chapter 1, we defined the scope of the thesis to focus on the scaling issue in SLAM. Accordingly, the problems of data association and environment representation have been ignored. The Johnson Center experiment utilized hurdles from the MIT Track team as "point" features. Measurements were associated to features using joint compatibility and random sample consensus, two state-of-the-art methods for data association in SLAM.

In moving our techniques to the ocean environment, we propose to employ the sonar data association and feature modeling techniques developed by Rikoski [72] and to combine them with the submap matching and relocation (loop closing) approach developed by Bosse

in the Atlas framework [10].

5.2.3 The multi-vehicle case

The problem of cooperative mapping by multiple AUVs is particularly useful and fascinating. For example, one project ongoing at MIT in collaboration with Bluefin Robotics seeks to develop the capability of rapid search and survey of large environments using heterogeneous network of AUVs. In this approach, some vehicles (called Comm/Nav Aids) are equipped with high-accuracy inertial navigation systems and other vehicles (called Search-Classify-Map vehicles) have higher resolution object mapping sonars but less accurate navigation capabilities. The vision for this work is that the multiple vehicles navigate together, using a technique known as "Moving Baseline Navigation" [69], which is analogous to conventional undersea Long Base Line (LBL) Navigation [60], but with mobile transponders.

The data fusion requirements of this type of system are very challenging because of the danger of the inconsistency that can occur when combining sensor information in a distributed sensor network. If shared robot pose information forms a cycle, then consistency breaks down. For example, if Robot A uses the pose information of robot B, that is based on previously communicated pose information for robot A, a cycle is formed and inconsistency will result. The methods developed in this thesis can prevent this type of situation, preventing inconsistency and enabling safe cooperative mapping and navigation of multiple AUVs.

5.2.4 Integration with autonomous exploration

In Chapter 1, we began the thesis by writing of the "dream" of a robot that could navigate fully autonomously. Subsequently, we have focused only on a narrow aspect of the overall navigation and mapping problem — large-scale SLAM. In future work, we hope to integrate the algorithms that we have developed and to apply the insights into the SLAM problem solution structure that we have gained, to build truly autonomous robots capable of making long distance, long-term duration excursions in unknown environments.

There has been extensive of work on the robot exploration problem, but there has been very little work, to our knowledge, that has coupled exploration and large-scale SLAM. For example, Newman, Bosse, and Leonard demonstrated feature-based real-time exploration of

unknown environments [66], but this approach did not explicitly reason about loop closing or even about the convergence of local maps.

Our investigation on the properties of the covariance matrix tells us that not only the frequency but also the manner and the order of observation matter to get good local convergence. For example, it is desired to observe multiple features simultaneously to strengthen their cross correlations. In terms of observing feature pairings, one large forest is preferred to multiple, weakly connected forests.

In term of the tree structure, it is desirable to balance the lengths of branches. One ideal structure is a tree whose branches are all one level; that is, all nodes except the root are leaves. During exploration, we can try to balance the tree structure by grafting branches.

In closing, we feel that our results demonstrate the utility of a network optimization view of the SLAM problem. Given that a network of submaps is a powerful representation for large-scale mapping, it is natural that network optimization algorithms [22, 1] can be effectively applied to SLAM to yield better performance.

We believe that one can formulate a unified Autonomous Mapping, Exploration, and Navigation (AMEN) problem that seeks to determine the optimal motion strategy for a robot to explore an unknown environment, applying network optimization algorithms to concurrent achieve (1) a desired coverage of accessible space, (2) the addition of new submap links to close loops and to connect different explored regions, (3) good convergence in local maps, and (4) doing this in the minimal amount of time. Given that we have a stochastic problem (due to robot motion and sensor uncertainty) and the network link costs are nonlinear, there are many open challenges to solve.

Appendix A

Solution Procedure for the One DOF SLAM Problem

This section provides details of the solution procedure for the one degree-of-freedom (MonoRob) SLAM problem presented in Chapter 3. We consider two cases: (1) one observed feature and two unobserved features, and (2) two observed features and one unobserved feature.

For simplicity, the number of features is set to be three (i, j, and n). The expansion to include more features is straightforward.

A.1 One Observed Feature and Two Unobserved Features

The MonoRob problem formulation for one observed feature (feature i) and two unobserved features (features j and n) is as follows:

- Dynamic Model

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\left(\mathbf{u}(t) + \mathbf{w}(t)\right) \tag{A.1}$$

  where, $\mathbf{F} = \mathbf{0}$, and $\mathbf{G} = [1\ 0\ \cdots\ 0]^T$

- Measurement Model

$$\hat{\mathbf{z}}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{v} \tag{A.2}$$

  where, $\mathbf{z}(t) = z_i(t)$, $\mathbf{v} = v_i$, $\mathbf{H} = [-1\ 0\ \cdots\ 1\ \cdots\ 0]$

- miscellaneous

$$\begin{aligned} \mathbf{x}(t) &= [x_v(t)\ x_1(t)\ \cdots\ x_n(t)]^T \\ \mathbf{Q} &= E[\mathbf{w}\mathbf{w}^T] = q \\ \mathbf{R} &= E[\mathbf{v}\mathbf{v}^T] = r_i \end{aligned} \tag{A.3}$$

- Solution by the differential Riccati equation

$$\begin{aligned}\dot{\mathbf{P}}(t) &= \mathbf{F}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T - \mathbf{P}(t)^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}(t) \\ \mathbf{P}(t) &= \mathbf{U}(t)\mathbf{V}^{-1}\end{aligned} \tag{A.4}$$

where $\mathbf{U}$ and $\mathbf{V}$ satisfy the following:

$$\begin{bmatrix}\dot{\mathbf{U}}(t) \\ \dot{\mathbf{V}}(t)\end{bmatrix} = \begin{bmatrix}\mathbf{F} & \mathbf{G}\mathbf{Q}\mathbf{G}^T \\ \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} & -\mathbf{F}^T\end{bmatrix}\begin{bmatrix}\mathbf{U}(t) \\ \mathbf{V}(t)\end{bmatrix} \tag{A.5}$$

with,

$$\begin{bmatrix}\mathbf{U}(0) \\ \mathbf{V}(0)\end{bmatrix} = \begin{bmatrix}\mathbf{P}(0) \\ \mathbf{I}\end{bmatrix}. \tag{A.6}$$

A.1.1 $\dot{\mathbf{U}}$ and $\dot{\mathbf{V}}$

$$\begin{aligned}\dot{\mathbf{U}}(t) = GQG^T\mathbf{V}(t) &= \begin{bmatrix}q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}V_{vv}(t) & V_{vi}(t) & V_{vj}(t) & V_{vn} \\ V_{iv}(t) & V_{ii}(t) & V_{ij}(t) & V_{in} \\ V_{jv}(t) & V_{ji}(t) & V_{jj}(t) & V_{jn} \\ V_{nv}(t) & V_{ni}(t) & V_{nj}(t) & V_{nn}\end{bmatrix} \\ &= \begin{bmatrix}qV_{vv}(t) & qV_{vi}(t) & qV_{vj}(t) & qV_{vn} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0\end{bmatrix}\end{aligned} \tag{A.7}$$

$$\begin{aligned}\dot{\mathbf{V}}(t) = H^TR^{-1}H\mathbf{U}(t) &= \begin{bmatrix}\alpha & -\alpha & 0 & 0 \\ -\alpha & \alpha & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}U_{vv}(t) & U_{vi}(t) & U_{vj}(t) & U_{vn}(t) \\ U_{iv}(t) & U_{ii}(t) & U_{ij}(t) & U_{in}(t) \\ U_{jv}(t) & U_{ji}(t) & U_{jj}(t) & U_{jn}(t) \\ U_{nv}(t) & U_{ni}(t) & U_{nj}(t) & U_{nn}(t)\end{bmatrix} \\ &= \begin{bmatrix}\alpha(U_{vv}(t)-U_{iv}(t)) & \alpha(U_{vi}(t)-U_{ii}(t)) & \alpha(U_{vj}(t)-U_{ij}(t)) & \alpha(U_{vn}(t)-U_{in}(t)) \\ -\alpha(U_{vv}(t)-U_{iv}(t)) & -\alpha(U_{vi}(t)-U_{ii}(t)) & -\alpha(U_{vj}(t)-U_{ij}(t)) & -\alpha(U_{vn}(t)-U_{in}(t)) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0\end{bmatrix}\end{aligned} \tag{A.8}$$

A.1.2 Solutions of U and V

1. $U_{vv}(t)$

$$
\begin{aligned}
\dot{U}_{vv}(t) &= qV_{vv}(t) \\
\ddot{U}_{vv}(t) &= q\dot{V}_{vv}(t) \\
&= q\left\{\alpha\left[U_{vv}(t) - P_{iv}\right]\right\} \\
\ddot{U}_{vv}(t) - q\alpha U_{vv}(t) + q\alpha P_{iv} &= 0 \\
U_{vv}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{iv}
\end{aligned}
\tag{A.9}
$$

By the boundary condition,

$$
\begin{aligned}
U_{vv}(0) = P_{vv} &= A + B + P_{iv} \\
A + B &= P_{vv} - P_{iv}
\end{aligned}
\tag{A.10}
$$

its 1st order derivative, and its boundary condition

$$
\begin{aligned}
\dot{U}_{vv}(t) &= A\sqrt{q\alpha}e^{\sqrt{q\alpha}t} - B\sqrt{q\alpha}e^{-\sqrt{q\alpha}t} \\
\dot{U}_{vv}(0) = qV_{vv}(0) = q &= (A - B)\sqrt{q\alpha} \\
A - B &= \sqrt{\frac{q}{\alpha}}
\end{aligned}
\tag{A.11}
$$

it follows that

$$
\begin{aligned}
A &= \frac{1}{2}\left[P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right] \\
B &= \frac{1}{2}\left[P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right]
\end{aligned}
\tag{A.12}
$$

$$
\begin{aligned}
U_{vv}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{iv} \\
&= \frac{1}{2}\left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} + \frac{1}{2}\left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} + P_{iv} \\
&= \frac{1}{2}\left(P_{vv} - P_{iv}\right)\left(e^{\sqrt{q\alpha}t} + e^{-\sqrt{q\alpha}t}\right) + \frac{1}{2}\sqrt{\frac{q}{\alpha}}\left(e^{\sqrt{q\alpha}t} - e^{-\sqrt{q\alpha}t}\right) + P_{iv} \\
&= \left(P_{vv} - P_{iv}\right)\cosh(\sqrt{q\alpha}t) + \sqrt{\frac{q}{\alpha}}\sinh(\sqrt{q\alpha}t) + P_{iv}
\end{aligned}
\tag{A.13}
$$

2. $U_{vi}(t)$

$$
\begin{aligned}
\dot{U}_{vi}(t) &= qV_{vi}(t) \\
\ddot{U}_{vi}(t) &= q\dot{V}_{vi}(t) \\
&= q\left\{\alpha\left[U_{vi}(t) - P_{ii}\right]\right\} \\
\ddot{U}_{vi}(t) - q\alpha U_{vi}(t) + q\alpha P_{ii} &= 0 \\
U_{vi}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{ii}
\end{aligned}
\tag{A.14}
$$

By the boundary conditions,

$$
\begin{aligned}
U_{vi}(0) = P_{vi} &= A + B + P_{ii} \\
A + B &= P_{vi} - P_{ii}
\end{aligned}
\tag{A.15}
$$

its first order derivative, and its boundary condition

$$
\begin{aligned}
\dot{U}_{vi}(t) &= A\sqrt{q\alpha}e^{\sqrt{q\alpha}t} - B\sqrt{q\alpha}e^{-\sqrt{q\alpha}t} \\
\dot{U}_{vi}(0) = qV_{vi}(0) = 0 &= (A - B)\sqrt{q\alpha} \\
A - B &= 0
\end{aligned}
\tag{A.16}
$$

it follows that

$$
\begin{aligned}
A &= \frac{1}{2}\left[P_{vi} - P_{ii}\right] \\
B &= \frac{1}{2}\left[P_{vi} - P_{ii}\right]
\end{aligned}
\tag{A.17}
$$

$$
\begin{aligned}
U_{vi}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{ii} \\
&= \frac{1}{2}\left(P_{vi} - P_{ii}\right)e^{\sqrt{q\alpha}t} + \frac{1}{2}\left(P_{vi} - P_{ii}\right)e^{-\sqrt{q\alpha}t} + P_{ii} \\
&= \frac{1}{2}\left(P_{vi} - P_{ii}\right)\left(e^{\sqrt{q\alpha}t} + e^{-\sqrt{q\alpha}t}\right) + P_{ii} \\
&= \left(P_{vi} - P_{ii}\right)\cosh(\sqrt{q\alpha}t) + P_{ii}
\end{aligned}
\tag{A.18}
$$

3. $U_{vj}(t)$

$$
\begin{aligned}
\dot{U}_{vj}(t) &= qV_{vj}(t) \\
\ddot{U}_{vj}(t) &= q\dot{V}_{vj}(t) \\
&= q\left\{\alpha\left[U_{vj}(t) - P_{ij}\right]\right\} \\
\ddot{U}_{vj}(t) - q\alpha U_{vj}(t) + q\alpha P_{ij} &= 0 \\
U_{vj}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{ij}
\end{aligned}
\tag{A.19}
$$

By the boundary condition,

$$\begin{aligned} U_{vj}(0) = P_{vj} &= A + B + P_{ij} \\ A + B &= P_{vj} - P_{ij} \end{aligned} \tag{A.20}$$

its first order derivative, and its boundary condition

$$\begin{aligned} \dot{U}_{vj}(t) &= A\sqrt{q\alpha}e^{\sqrt{q\alpha}t} - B\sqrt{q\alpha}e^{-\sqrt{q\alpha}t} \\ \dot{U}_{vj}(0) = qV_{vj}(0) = 0 &= (A - B)\sqrt{q\alpha} \\ A - B &= 0 \end{aligned} \tag{A.21}$$

it follows that

$$\begin{aligned} A &= \frac{1}{2}\left[P_{vj} - P_{ij}\right] \\ B &= \frac{1}{2}\left[P_{vj} - P_{ij}\right] \end{aligned} \tag{A.22}$$

$$\begin{aligned} U_{vj}(t) &= Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{ij} \\ &= \frac{1}{2}\left(P_{vj} - P_{ij}\right)e^{\sqrt{q\alpha}t} + \frac{1}{2}\left(P_{vj} - P_{ij}\right)e^{-\sqrt{q\alpha}t} + P_{ij} \\ &= \frac{1}{2}\left(P_{vj} - P_{ij}\right)\left(e^{\sqrt{q\alpha}t} + e^{-\sqrt{q\alpha}t}\right) + P_{ij} \\ &= \left(P_{vj} - P_{ij}\right)\cosh(\sqrt{q\alpha}t) + P_{ij} \end{aligned} \tag{A.23}$$

4. $U_{vn}(t)$

$$\begin{aligned} U_{vn}(t) &= \left(P_{vn} - P_{in}\right)\cosh(\sqrt{q\alpha}t) + P_{in} \\ &= \frac{1}{2}\left(P_{vn} - P_{in}\right)e^{\sqrt{q\alpha}t} + \frac{1}{2}\left(P_{vn} - P_{in}\right)e^{-\sqrt{q\alpha}t} + P_{in} \end{aligned} \tag{A.24}$$

5. $V_{vv}(t)$

$$\begin{aligned} \dot{V}_{vv}(t) &= \alpha\left\{U_{vv}(t) - P_{iv}\right\} \\ &= \alpha\left\{Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t} + P_{iv} - P_{iv}\right\} \\ &= \alpha\left\{Ae^{\sqrt{q\alpha}t} + Be^{-\sqrt{q\alpha}t}\right\} \\ V_{vv}(t) &= \alpha\frac{1}{\sqrt{q\alpha}}\left\{Ae^{\sqrt{q\alpha}t} - Be^{-\sqrt{q\alpha}t}\right\} + C \end{aligned} \tag{A.25}$$

$$
\begin{aligned}
V_{vv}(0) = 1 &= \alpha \frac{1}{\sqrt{q\alpha}} \{A - B\} + C \\
&= \alpha \frac{1}{\sqrt{q\alpha}} \frac{q}{\sqrt{q\alpha}} + C \\
&= 1 + C \\
C &= 0
\end{aligned}
$$

$$
\begin{aligned}
V_{vv}(t) &= \alpha \frac{1}{\sqrt{q\alpha}} \left\{ A e^{\sqrt{q\alpha}t} - B e^{-\sqrt{q\alpha}t} \right\} \\
&= \frac{1}{2}\sqrt{\frac{\alpha}{q}} \left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}} \right) e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}} \left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}} \right) e^{-\sqrt{q\alpha}t} \\
&= \sqrt{\frac{\alpha}{q}} (P_{vv} - P_{iv}) \sinh(\sqrt{q\alpha}t) + \cosh(\sqrt{q\alpha}t)
\end{aligned}
$$

6. $V_{vi}(t)$, $V_{vj}(t)$ and $V_{vn}(t)$

$$
\begin{aligned}
\dot{V}_{vi}(t) &= \alpha \{U_{vi}(t) - P_{ii}\} \\
&= \alpha \{(P_{vi} - P_{ii}) \cosh(\sqrt{q\alpha}t)\} \\
V_{vi}(t) &= \alpha \frac{(P_{vi} - P_{ii})}{\sqrt{q\alpha}} \sinh(\sqrt{q\alpha}t) + C \\
V_{vi}(0) = 0 &= C \\
V_{vi}(t) &= \sqrt{\frac{\alpha}{q}} (P_{vi} - P_{ii}) \sinh(\sqrt{q\alpha}t) \\
&= \frac{1}{2}\sqrt{\frac{\alpha}{q}} (P_{vi} - P_{ii}) e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}} (P_{vi} - P_{ii}) e^{-\sqrt{q\alpha}t}
\end{aligned}
\tag{A.26}
$$

$$
\begin{aligned}
\dot{V}_{vj}(t) &= \frac{1}{q}\dot{U}_{vj} \\
&= \frac{1}{q} \left\{ \frac{1}{2}\sqrt{q\alpha} (P_{vj} - P_{ij}) e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{q\alpha} (P_{vj} - P_{ij}) e^{-\sqrt{q\alpha}t} + P_{ij} \right\} \\
&= \frac{1}{2}\sqrt{\frac{\alpha}{q}} (P_{vj} - P_{ij}) e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}} (P_{vj} - P_{ij}) e^{-\sqrt{q\alpha}t} \\
&= \sqrt{\frac{\alpha}{q}} (P_{vj} - P_{ij}) \sinh(\sqrt{q\alpha}t)
\end{aligned}
\tag{A.27}
$$

$$
\begin{aligned}
\dot{V}_{vn}(t) &= \alpha \{U_{vn}(t) - P_{in}\} \\
V_{vn}(t) &= \sqrt{\frac{\alpha}{q}} (P_{vn} - P_{in}) \sinh(\sqrt{q\alpha}t) \\
&= \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn} - P_{in})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn} - P_{in})e^{-\sqrt{q\alpha}t}
\end{aligned}
\tag{A.28}
$$

7. $U_{iv}(t)$, $U_{ii}(t)$, $U_{ij}(t)$, and $U_{in}(t)$

$$
\begin{aligned}
\dot{U}_{iv}(t) = 0 &\rightarrow U_{iv}(t) = P_{iv} \\
\dot{U}_{ii}(t) = 0 &\rightarrow U_{ii}(t) = P_{ii} \\
\dot{U}_{ij}(t) = 0 &\rightarrow U_{ij}(t) = P_{ij} \\
\dot{U}_{in}(t) = 0 &\rightarrow U_{in}(t) = P_{in}
\end{aligned}
\tag{A.29}
$$

8. $V_{iv}(t)$

$$
\begin{aligned}
\dot{V}_{iv}(t) &= -\alpha(U_{vv}(t) - U_{iv}(t)) \\
&= -\left\{\sqrt{\frac{\alpha}{q}}(P_{vv} - P_{iv}) \sinh(\sqrt{q\alpha}t) + \cosh(\sqrt{q\alpha}t)\right\} + C \\
V_{iv}(0) = 0 &= -1 + C \\
C &= 1 \\
V_{iv}(t) &= -\left\{\sqrt{\frac{\alpha}{q}}(P_{vv} - P_{iv}) \sinh(\sqrt{q\alpha}t) + \cosh(\sqrt{q\alpha}t)\right\} + 1 \\
&= -\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right) e^{\sqrt{q\alpha}t} \\
&\quad + \frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right) e^{-\sqrt{q\alpha}t} + 1
\end{aligned}
\tag{A.30}
$$

9. $V_{ii}(t)$

$$\begin{aligned}
\dot{V}_{ii}(t) &= -\dot{V}_{vi}(t) \\
V_{ii}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vi} - P_{ii})\sinh(\sqrt{q\alpha}t) + C \\
V_{ii}(0) = 1 &= 0 + C \\
C &= 1 \\
V_{ii}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vi} - P_{ii})\sinh(\sqrt{q\alpha}t) + 1 \\
&= -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t} + 1
\end{aligned} \tag{A.31}$$

10. $V_{ij}(t)$

$$\begin{aligned}
\dot{V}_{ij}(t) &= -\dot{V}_{vj}(t) \\
V_{ij}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vj} - P_{ij})\sinh(\sqrt{q\alpha}t) + C \\
V_{ij}(0) = 0 &= C \\
V_{ij}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vj} - P_{ij})\sinh(\sqrt{q\alpha}t) \\
&= -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj} - P_{ij})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj} - P_{ij})e^{-\sqrt{q\alpha}t}
\end{aligned} \tag{A.32}$$

11. $V_{in}(t)$

$$\begin{aligned}
\dot{V}_{in}(t) &= -\dot{V}_{vn}(t) \\
V_{in}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vn} - P_{in})\sinh(\sqrt{q\alpha}t) + C \\
V_{in}(0) = 0 &= C \\
V_{in}(t) &= -\sqrt{\frac{\alpha}{q}}\,(P_{vn} - P_{in})\sinh(\sqrt{q\alpha}t) \\
&= -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn} - P_{in})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn} - P_{in})e^{-\sqrt{q\alpha}t}
\end{aligned} \tag{A.33}$$

12. The other elements

$$
\begin{aligned}
\dot{U}_{jv}(t) = 0 &\rightarrow U_{jv}(t) = P_{jv} \\
\dot{U}_{ji}(t) = 0 &\rightarrow U_{ji}(t) = P_{ji} \\
\dot{U}_{jj}(t) = 0 &\rightarrow U_{ji}(t) = P_{jj} \\
\dot{U}_{jn}(t) = 0 &\rightarrow U_{jn}(t) = P_{jn} \\
\dot{U}_{nv}(t) = 0 &\rightarrow U_{nv}(t) = P_{nv} \\
\dot{U}_{ni}(t) = 0 &\rightarrow U_{ni}(t) = P_{ni} \\
\dot{U}_{nj}(t) = 0 &\rightarrow U_{ni}(t) = P_{nj} \\
\dot{U}_{nn}(t) = 0 &\rightarrow U_{nn}(t) = P_{nn}
\end{aligned}
\tag{A.34}
$$

$$
\begin{aligned}
\dot{V}_{jv}(t) = 0 &\rightarrow V_{jv}(t) = 0 \\
\dot{V}_{ji}(t) = 0 &\rightarrow V_{ji}(t) = 0 \\
\dot{V}_{jj}(t) = 0 &\rightarrow V_{jj}(t) = 1 \\
\dot{V}_{jn}(t) = 0 &\rightarrow V_{jn}(t) = 0 \\
\dot{V}_{nv}(t) = 0 &\rightarrow V_{nv}(t) = 0 \\
\dot{V}_{ni}(t) = 0 &\rightarrow V_{ni}(t) = 0 \\
\dot{V}_{nj}(t) = 0 &\rightarrow V_{nj}(t) = 0 \\
\dot{V}_{nn}(t) = 0 &\rightarrow V_{nn}(t) = 1
\end{aligned}
\tag{A.35}
$$

A.1.3 Summary in matrix form

- $\mathbf{U}(t)$

$$
\begin{aligned}
\mathbf{U}(t) &= \begin{bmatrix}
(P_{vv}-P_{iv})\cosh(\sqrt{q\alpha}t)+\left(\frac{q}{\sqrt{q\alpha}}\right)\sinh(\sqrt{q\alpha}t)+P_{iv} & P_{iv} & P_{jv} & P_{nv} \\
(P_{vi}-P_{ii})\cosh(\sqrt{q\alpha}t)+P_{ii} & P_{ii} & P_{ji} & P_{ni} \\
(P_{vj}-P_{ij})\cosh(\sqrt{q\alpha}t)+P_{ij} & P_{ij} & P_{jj} & P_{nj} \\
(P_{vn}-P_{in})\cosh(\sqrt{q\alpha}t)+P_{in} & P_{in} & P_{jn} & P_{nn}
\end{bmatrix}^T \\
&= \begin{bmatrix}
\frac{1}{2}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t}+\frac{1}{2}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}+P_{iv} & P_{iv} & P_{jv} & P_{nv} \\
\frac{1}{2}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t}+\frac{1}{2}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t}+P_{ii} & P_{ii} & P_{ji} & P_{ni} \\
\frac{1}{2}(P_{vj}-P_{ij})e^{\sqrt{q\alpha}t}+\frac{1}{2}(P_{vj}-P_{ij})e^{-\sqrt{q\alpha}t}+P_{ij} & P_{ij} & P_{jj} & P_{jn} \\
\frac{1}{2}(P_{vn}-P_{in})e^{\sqrt{q\alpha}t}+\frac{1}{2}(P_{vn}-P_{in})e^{-\sqrt{q\alpha}t}+P_{in} & P_{in} & P_{jn} & P_{nn}
\end{bmatrix}^T
\end{aligned}
\tag{A.36}
$$

- $\mathbf{V}(t)$

$$
\begin{aligned}
\mathbf{V}(t) &= \begin{bmatrix}
\frac{\alpha}{\sqrt{q\alpha}}(P_{vv}-P_{iv})\sinh(\sqrt{q\alpha}t)+\cosh(\sqrt{q\alpha}t) & \alpha\frac{(P_{vi}-P_{ii})}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t) & \alpha\frac{\left(P_{vj}-P_{ij}\right)}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t) & \alpha\frac{(P_{vn}-P_{in})}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t) \\
-\left\{\frac{\alpha}{\sqrt{q\alpha}}(P_{vv}-P_{iv})\sinh(\sqrt{q\alpha}t)+\cosh(\sqrt{q\alpha}t)\right\}+1 & -\alpha\frac{(P_{vi}-P_{ii})}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t)+1 & -\alpha\frac{\left(P_{vj}-P_{ij}\right)}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t) & -\alpha\frac{(P_{vn}-P_{in})}{\sqrt{q\alpha}}\sinh(\sqrt{q\alpha}t) \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \\
&= \begin{bmatrix}
\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t}-\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} & -\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t}+\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}+1 & 0 & 0 \\
\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t}-\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t} & -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t}+\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t}+1 & 0 & 0 \\
\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{\sqrt{q\alpha}t}-\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{-\sqrt{q\alpha}t} & -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{\sqrt{q\alpha}t}+\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{-\sqrt{q\alpha}t} & 1 & 0 \\
\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{\sqrt{q\alpha}t}-\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{-\sqrt{q\alpha}t} & -\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{\sqrt{q\alpha}t}+\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{-\sqrt{q\alpha}t} & 0 & 1
\end{bmatrix}^T
\end{aligned}
\tag{A.37}
$$

A.1.4 Inverse of $\mathbf{V}(t)$

Let $\mathbf{V}(t)$ be $\begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ V_{iv} & V_{ii} & V_{ij} & V_{in} \\ V_{jv} & V_{ji} & V_{jj} & V_{jn} \\ V_{nv} & V_{ni} & V_{nj} & V_{nn} \end{bmatrix}$. Then,

$$\mathbf{V}(t) = \begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ V_{iv} & V_{ii} & V_{ij} & V_{in} \\ V_{jv} & V_{ji} & V_{jj} & V_{jn} \\ V_{nv} & V_{ni} & V_{nj} & V_{nn} \end{bmatrix} = \begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ 1 - V_{vv} & 1 - V_{vi} & -V_{vj} & -V_{vn} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.38}$$

$$\begin{aligned} \mathbf{V}^{adj}(t) &= \begin{bmatrix} V_{ii} & -V_{vi} & V_{vi}V_{ij} - V_{vj}V_{ii} & V_{vi}V_{in} - V_{vn}V_{ii} \\ -V_{iv} & V_{vv} & V_{vj}V_{iv} - V_{vv}V_{ij} & V_{vn}V_{iv} - V_{vv}V_{in}) \\ 0 & 0 & V_{vv}V_{ii} - V_{vi}V_{iv} & 0 \\ 0 & 0 & 0 & V_{vv}V_{ii} - V_{vi}V_{iv} \end{bmatrix} \\ &= \begin{bmatrix} V_{ii} & -V_{vi} & -V_{vj} & -V_{vn} \\ -V_{iv} & V_{vv} & V_{vj} & V_{vn} \\ 0 & 0 & V_{vv} - V_{vi} & 0 \\ 0 & 0 & 0 & V_{vv} - V_{vi} \end{bmatrix} \end{aligned} \tag{A.39}$$

and

$$\begin{aligned} det(\mathbf{V}(t)) &= V_{vv}\{V_{ii} \cdot 1 \cdot 1\} + V_{vi}\{V_{iv} \cdot 1 \cdot 1\} + V_{vj} \cdot 0 + V_{vn} \cdot 0 \\ &= V_{vv}V_{ii} - V_{vi}V_{iv} \\ &= V_{vv}(1 - V_{vi}) - V_{vi}(1 - V_{vv}) \\ &= V_{vv} - V_{vi} \\ &= \sqrt{\frac{\alpha}{q}}(P_{vv} - P_{iv})\sinh(\sqrt{q\alpha}t) + \cosh(\sqrt{q\alpha}t) - \sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})\sinh(\sqrt{q\alpha}t) \\ &= \sqrt{\frac{\alpha}{q}}(P_{vv} - P_{iv} - P_{vi} + P_{ii})\sinh(\sqrt{q\alpha}t) + \cosh(\sqrt{q\alpha}t) \\ &= \frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{\left[P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right]e^{\sqrt{q\alpha}t} - \left[P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right]e^{-\sqrt{q\alpha}t}\right\} \\ &\quad - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t} \\ &= \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left(P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right) \\ &\quad - \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left(P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right) \end{aligned} \tag{A.40}$$

Thus,

$$\mathbf{V}^{-1}(t) = \frac{1}{V_{vv}V_{ii} - V_{iv}V_{vi}} \begin{bmatrix} V_{ii} & -V_{vi} & V_{vi}V_{ij} - V_{vj}V_{ii} & V_{vi}V_{in} - V_{vn}V_{ii} \\ -V_{iv} & V_{vv} & V_{vj}V_{iv} - V_{vv}V_{ij} & V_{vn}V_{iv} - V_{vv}V_{in}) \\ 0 & 0 & V_{vv}V_{ii} - V_{vi}V_{iv} & 0 \\ 0 & 0 & 0 & V_{vv}V_{ii} - V_{vi}V_{iv} \end{bmatrix}$$
$$= \frac{1}{V_{vv}V_{ii} - V_{iv}V_{vi}} \begin{bmatrix} V_{ii} & -V_{vi} & -V_{vj} & -V_{vn} \\ -V_{iv} & V_{vv} & V_{vj} & V_{vn} \\ 0 & 0 & V_{vv} - V_{vi} & 0 \\ 0 & 0 & 0 & V_{vv} - V_{vi} \end{bmatrix} \tag{A.41}$$

A.1.5 Elements of the covariance matrix

1. $\mathbf{P}_{vv}$

$$\begin{aligned}
\mathbf{P}_{vv}(t) &= (U_{vv}V_{vv}^{-1} + U_{vi}V_{iv}^{-1} + U_{vj}V_{jv}^{-1} + U_{vn}V_{nv}^{-1}) \\
&= \frac{1}{D}\left[U_{vv}V_{ii} + U_{vi}(-V_{iv})\right] \\
&= \tfrac{1}{D}\left\{\tfrac{1}{2}\left(P_{vv} - P_{iv} + \tfrac{q}{\sqrt{q\alpha}}\right)e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\left(P_{vv} - P_{iv} - \tfrac{q}{\sqrt{q\alpha}}\right)e^{-\sqrt{q\alpha}t} + P_{iv}\right\} \\
&\quad \times\left\{-\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t} + 1\right\} \\
&\quad -\tfrac{1}{D}\left\{\tfrac{1}{2}(P_{vi} - P_{ii})\,e^{\sqrt{q\alpha}t} + \tfrac{1}{2}(P_{vi} - P_{ii})\,e^{-\sqrt{q\alpha}t} + P_{ii}\right\} \\
&\quad \times\left\{-\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} + 1\right\} \\
&= \frac{e^{2\sqrt{q\alpha}t}}{D}\left\{-\tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)(P_{vi} - P_{ii}) + \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{e^{-2\sqrt{q\alpha}t}}{D}\left\{\tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)(P_{vi} - P_{ii}) - \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{e^{\sqrt{q\alpha}t}}{D}\begin{Bmatrix} \tfrac{1}{2}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right) - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}P_{iv}(P_{vi} - P_{ii}) \\ +\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}P_{ii}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right) - \tfrac{1}{2}(P_{vi} - P_{ii}) \end{Bmatrix} \\
&\quad + \frac{e^{-\sqrt{q\alpha}t}}{D}\begin{Bmatrix} \tfrac{1}{2}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right) + \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}P_{iv}(P_{vi} - P_{ii}) \\ -\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}P_{ii}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right) - \tfrac{1}{2}(P_{vi} - P_{ii}) \end{Bmatrix} \\
&\quad + \frac{1}{D}\begin{Bmatrix} \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)(P_{vi} - P_{ii}) - \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)(P_{vi} - P_{ii}) + P_{iv} \\ -\tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right) + \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right) - P_{ii} \end{Bmatrix}
\end{aligned} \tag{A.42}$$

$$= \frac{e^{\sqrt{q\alpha}t}}{D}\left\{\tfrac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\tfrac{q}{\alpha}}\right)+\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left[-P_{iv}P_{vi}+P_{iv}P_{ii}+P_{ii}P_{vv}-P_{ii}P_{iv}+P_{ii}\sqrt{\tfrac{q}{\alpha}}\right]\right\}$$
$$+ \frac{e^{-\sqrt{q\alpha}t}}{D}\left\{\tfrac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}-\sqrt{\tfrac{q}{\alpha}}\right)+\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left[P_{iv}P_{vi}-P_{iv}P_{ii}-P_{ii}P_{vv}+P_{ii}P_{iv}+P_{ii}\sqrt{\tfrac{q}{\alpha}}\right]\right\}$$
$$+ \frac{1}{D}\left\{\begin{array}{c}\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left[(P_{vv}-P_{iv})(P_{vi}-P_{ii})+(P_{vi}-P_{ii})\sqrt{\tfrac{q}{\alpha}}\right] \\ -\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left[(P_{vv}-P_{iv})(P_{vi}-P_{ii})-(P_{vi}-P_{ii})\sqrt{\tfrac{q}{\alpha}}\right] \\ +P_{iv}-P_{ii}\end{array}\right\}$$

$$= \frac{e^{\sqrt{q\alpha}t}}{D}\left\{\frac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right)+\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{ii}P_{vv}-P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}$$
$$+ \frac{e^{-\sqrt{q\alpha}t}}{D}\left\{\frac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right)+\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[-P_{ii}P_{vv}+P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}$$
$$+ \frac{1}{D}\left\{\frac{2}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})\sqrt{\frac{q}{\alpha}}+P_{iv}-P_{ii}\right\}$$

$$= \frac{e^{\sqrt{q\alpha}t}}{D}\left\{\frac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right)+\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{ii}P_{vv}-P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}$$
$$+ \frac{e^{-\sqrt{q\alpha}t}}{D}\left\{\frac{1}{2}\left(P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right)+\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[-P_{ii}P_{vv}+P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}$$
$$+ \frac{2}{D}\{P_{vi}-P_{ii}\}$$

$$\Rightarrow \frac{\left\{\sqrt{\frac{q}{\alpha}}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right)+\left[P_{ii}P_{vv}-P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}$$

$$= \sqrt{\frac{q}{\alpha}} \quad + \quad \frac{\left[P_{ii}P_{vv}-P_{iv}^2+P_{ii}\sqrt{\frac{q}{\alpha}}\right]}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}$$

2. $\mathbf{P}_{vi}$

$$
\begin{aligned}
\mathbf{P}_{vi}(t) &= (U_{vv}V_{vi}^{-1} + U_{vi}V_{ii}^{-1} + U_{vj}V_{ji}^{-1} + U_{vn}V_{ni}^{-1}) \\
&= \frac{1}{D}\left[U_{vv}(-V_{vi}) + U_{vi}V_{vv}\right] \\
&= -\tfrac{1}{D}\left\{\tfrac{1}{2}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} + P_{iv}\right\} \\
&\quad \times\left\{\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t}\right\} \\
&\quad + \tfrac{1}{D}\left\{\tfrac{1}{2}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} + \tfrac{1}{2}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t} + P_{ii}\right\} \\
&\quad \times\left\{\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}\right\} \\
&= \frac{1}{D}e^{2\sqrt{q\alpha}t}\left\{-\tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)(P_{vi}-P_{ii}) + \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi}-P_{ii})\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}e^{-2\sqrt{q\alpha}t}\left\{+\tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)(P_{vi}-P_{ii}) - \tfrac{1}{4}\sqrt{\tfrac{\alpha}{q}}(P_{vi}-P_{ii})\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}e^{\sqrt{q\alpha}t}\left\{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}P_{iv}(P_{vi} - P_{ii}) + \frac{1}{2}\sqrt{\frac{\alpha}{q}}P_{ii}\left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}e^{-\sqrt{q\alpha}t}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}P_{iv}(P_{vi} - P_{ii}) - \frac{1}{2}\sqrt{\frac{\alpha}{q}}P_{ii}\left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}\frac{1}{4}\sqrt{\frac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right)(P_{vi} - P_{ii}) \\
&\quad - \frac{1}{D}\frac{1}{4}\sqrt{\frac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right)(P_{vi} - P_{ii}) \\
&\quad - \frac{1}{D}\frac{1}{4}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}}\right) \\
&\quad + \frac{1}{D}\frac{1}{4}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})\left(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}}\right) \\
&= \frac{1}{D}e^{\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{-P_{iv}P_{vi} + P_{iv}P_{ii} + P_{ii}P_{vv} - P_{ii}P_{iv} + P_{ii}\sqrt{\frac{q}{\alpha}}\right\} \\
&\quad + \frac{1}{D}e^{-\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{P_{iv}P_{vi} - P_{iv}P_{ii} - P_{ii}P_{vv} + P_{ii}P_{iv} + P_{ii}\sqrt{\frac{q}{\alpha}}\right\} \\
&\quad + \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)(P_{vi}-P_{ii}) - \left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)(P_{vi}-P_{ii})\right\}
\end{aligned}
\tag{A.43}
$$

$$= \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)$$
$$+ \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left(-P_{ii}P_{vv} + P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)$$
$$+ \frac{1}{D}(P_{vi} - P_{ii})$$

$$= \frac{e^{\sqrt{q\alpha}t}\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right) + e^{-\sqrt{q\alpha}t}\left(-P_{ii}P_{vv} + P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right) + 2\sqrt{\frac{q}{\alpha}}(P_{vi} - P_{ii})}{e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]}$$

$$\Rightarrow \frac{\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]}$$

3. $\mathbf{P}_{vj}$

$$\mathbf{P}_{vj}(t) = (U_{vv}V_{vj}^{-1} + U_{vi}V_{ij}^{-1} + U_{vj}V_{jj}^{-1} + U_{vn}V_{nj}^{-1})$$
$$= \frac{1}{D}\left[U_{vv}(-V_{vj}) + U_{vi}V_{vj} + U_{vj}(V_{vv} - V_{vi})\right]$$
$$= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}P_{vi} - P_{jv}P_{ii} - P_{ji}P_{vv} + P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{jv} - P_{ji}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]} \qquad \text{(A.44)}$$

$$\Rightarrow \frac{\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]}$$

4. $\mathbf{P}_{vn}$

$$\begin{aligned}
\mathbf{P}_{vn}(t) &= (U_{vv}V_{vn}^{-1} + U_{vi}V_{in}^{-1} + U_{vj}V_{jn}^{-1} + U_{vn}V_{nn}^{-1}) \\
&= \frac{1}{D}\left[U_{vv}(-V_{vn}) + U_{vi}V_{vn} + U_{vn}(V_{vv} - V_{vi})\right] \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}P_{vi} - P_{nv}P_{ii} - P_{ni}P_{vv} + P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{nv} - P_{ni}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]}
\end{aligned} \tag{A.45}$$

$$\Rightarrow \frac{\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]}$$

5. $\mathbf{P}_{iv}$

$$\begin{aligned}
\mathbf{P}_{iv}(t) &= (U_{iv}V_{vv}^{-1} + U_{ii}V_{iv}^{-1} + U_{ij}V_{jv}^{-1} + U_{in}V_{nv}^{-1}) \\
&= \frac{1}{D}\left[P_{iv}V_{ii} + P_{ii}(-V_{iv})\right] \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right) \\ +e^{-\sqrt{q\alpha}t}\left(-P_{ii}P_{vv} + P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right) \\ +2\sqrt{\frac{q}{\alpha}}(P_{vi} - P_{ii}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]}
\end{aligned}$$

$$\Rightarrow \frac{\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.46}$$

6. $\mathbf{P}_{ii}$

$$
\begin{aligned}
\mathbf{P}_{ii}(t) &= (U_{iv}V_{vi}^{-1} + U_{ii}V_{ii}^{-1} + U_{ij}V_{ji}^{-1} + U_{in}V_{ni}^{-1}) \\
&= \frac{1}{D}\left[P_{iv}(-V_{vi}) + P_{ii}V_{vv}\right] \\
&= -\frac{1}{D}P_{iv}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi} - P_{ii})e^{-\sqrt{q\alpha}t}\right\} \\
&\quad + \frac{1}{D}P_{ii}\left\{\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} + \sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv} - P_{iv} - \sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}\right\} \\
&= \frac{1}{D}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left[-P_{iv}(P_{vi} - P_{ii}) + P_{ii}(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}})\right]\right\} \\
&\quad + \frac{1}{D}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left[P_{iv}(P_{vi} - P_{ii}) - P_{ii}(P_{vv} - P_{iv} - \sqrt{\frac{q}{\alpha}})\right]\right\} \\
&= \frac{\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{e^{\sqrt{q\alpha}t}\left[P_{vv}P_{ii} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}})\right] + e^{-\sqrt{q\alpha}t}\left[P_{iv}^2 - P_{vv}P_{ii} + P_{ii}\sqrt{\frac{q}{\alpha}})\right]\right\}}{D} \\
&= \frac{\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left\{e^{\sqrt{q\alpha}t}\left[P_{vv}P_{ii} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}})\right] + e^{-\sqrt{q\alpha}t}\left[P_{iv}^2 - P_{vv}P_{ii} + P_{ii}\sqrt{\frac{q}{\alpha}})\right]\right\}}{\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]} \\
&= \frac{e^{\sqrt{q\alpha}t}\left[P_{vv}P_{ii} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}})\right] + e^{-\sqrt{q\alpha}t}\left[P_{iv}^2 - P_{vv}P_{ii} + P_{ii}\sqrt{\frac{q}{\alpha}})\right]}{e^{\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv} - 2P_{iv} + P_{ii} - \sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left[P_{vv}P_{ii} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}})\right]}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]}
\end{aligned}
\tag{A.47}
$$

7. $\mathbf{P}_{ij}\ \ (=\mathbf{P}_{ji})$

$$\begin{aligned}
\mathbf{P}_{ij} &= U_{iv}V_{vj}^{-1} + U_{ii}V_{ij}^{-1} + U_{ij}V_{jj}^{-1} + U_{in}V_{nj}^{-1} \\
&= \frac{1}{D}\left[P_{iv}(-V_{vj}) + P_{ii}V_{vj} + P_{ij}(V_{vv} - V_{vi})\right] \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{jv}(P_{vi}-P_{ii}) + P_{ji}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}(P_{vi}-P_{ii}) - P_{ji}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{jv}(P_{vi}-P_{ii}) + P_{ji}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned} \tag{A.48}$$

8. $\mathbf{P}_{in}\ \ (=\mathbf{P}_{ni})$

$$\begin{aligned}
\mathbf{P}_{in} &= U_{iv}V_{vn}^{-1} + U_{ii}V_{in}^{-1} + U_{ij}V_{jn}^{-1} + U_{in}V_{nn}^{-1} \\
&= \frac{1}{D}\left[P_{iv}(-V_{vn}) + P_{ii}V_{vn} + P_{in}(V_{vv} - V_{vi})\right] \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{nv}(P_{vi}-P_{ii}) + P_{ni}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}(P_{vi}-P_{ii}) - P_{ni}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{nv}(P_{vi}-P_{ii}) + P_{ni}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned} \tag{A.49}$$

9. $\mathbf{P}_{jv}$

$$\begin{aligned}
\mathbf{P}_{jv} &= U_{jv}V_{vv}^{-1} + U_{ji}V_{iv}^{-1} + U_{jj}V_{jv}^{-1} + U_{jn}V_{nv}^{-1} \\
&= \frac{1}{D}\left[P_{jv}V_{ii} + P_{ji}(-V_{iv})\right] \\
&= \frac{1}{D}P_{jv}\left[-\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t} + 1\right] \\
&\quad - \frac{1}{D}P_{ji}\left[-\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} + 1\right] \\
&= \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left\{-P_{jv}\left(P_{vi}-P_{ii}\right) + P_{ji}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left\{P_{jv}\left(P_{vi}-P_{ii}\right) - P_{ji}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}2\sqrt{\frac{q}{\alpha}}(P_{jv}-P_{ji}) \qquad \text{(A.50)} \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{jv}\left(P_{vi}-P_{ii}\right) + P_{ji}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}\left(P_{vi}-P_{ii}\right) - P_{ji}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{jv}-P_{ji}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}P_{vi} - P_{jv}P_{ii} - P_{ji}P_{vv} + P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{jv}-P_{ji}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned}$$

10. $\mathbf{P}_{ji}\ \ (=\mathbf{P}_{ij})$

$$
\begin{aligned}
\mathbf{P}_{ji} &= U_{jv}V_{vi}^{-1} + U_{ji}V_{ii}^{-1} + U_{jj}V_{ji}^{-1} + U_{jn}V_{ni}^{-1} \\
&= \frac{1}{D}\left[P_{jv}(-V_{vi}) + P_{ji}V_{vv}\right] \\
&= -\frac{1}{D}P_{jv}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t}\right\} \\
&\quad + \frac{1}{D}P_{ji}\left\{\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}\right\} \\
&= \frac{\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[\begin{matrix} e^{\sqrt{q\alpha}t}\left\{-P_{jv}(P_{vi}-P_{ii}) + P_{ji}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}(P_{vi}-P_{ii}) - P_{ji}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{matrix}\right]}{D} \\
&= \frac{\left\{\begin{matrix} e^{\sqrt{q\alpha}t}\left\{-P_{jv}(P_{vi}-P_{ii}) + P_{ji}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{jv}(P_{vi}-P_{ii}) - P_{ji}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{matrix}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{jv}(P_{vi}-P_{ii}) + P_{ji}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned}
\tag{A.51}
$$

11. $\mathbf{P}_{jj}$

$$
\begin{aligned}
\mathbf{P}_{jj} &= U_{jv}V_{vj}^{-1} + U_{ji}V_{ij}^{-1} + U_{jj}V_{jj}^{-1} + U_{jn}V_{nj}^{-1} \\
&= \frac{1}{D}\left[P_{jv}(-V_{vj}) + P_{ji}V_{vj} + P_{jj}(V_{vv}-V_{vi})\right] \\
&= \frac{1}{D}\left[P_{jv}(-V_{vj}) + P_{ji}V_{vj}\right] + P_{jj} \\
&= \frac{V_{vj}(P_{ji}-P_{jv})}{D} + P_{jj} \\
&= \frac{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}(P_{ji}-P_{jv})^2 + \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}(P_{ji}-P_{jv})^2}{D} + P_{jj} \\
&= \frac{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}(P_{ji}-P_{jv})^2 + \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}(P_{ji}-P_{jv})^2}{e^{\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} + P_{jj} \\
&\Rightarrow \frac{-(P_{ji}-P_{jv})^2}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]} + P_{jj}
\end{aligned}
\tag{A.52}
$$

12. $\mathbf{P}_{jn}$

$$
\begin{aligned}
\mathbf{P}_{jn} &= U_{jv}V_{vn}^{-1} + U_{ji}V_{in}^{-1} + U_{jj}V_{jn}^{-1} + U_{jn}V_{nn}^{-1} \\
&= \frac{1}{D}\left\{P_{jv}(-V_{vn}) + P_{ji}(V_{vn}) + P_{jj}\cdot 0 + P_{jn}(V_{vv} - V_{vi})\right\} \\
&= \frac{(P_{ji} - P_{jv})V_{vn}}{D} + P_{jn} \\
&= \frac{(P_{ji}-P_{jv})\left[\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vn}-P_{in})e^{-\sqrt{q\alpha}t}\right]}{\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right) - \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left(P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right)} + P_{jn} \\
&\Rightarrow \frac{(P_{ji} - P_{jv})(P_{vn} - P_{in})}{\left(P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right)} + P_{jn}
\end{aligned}
\tag{A.53}
$$

13. $\mathbf{P}_{nv}$

$$\begin{aligned}
\mathbf{P}_{nv} &= U_{nv}V_{vv}^{-1} + U_{ni}V_{iv}^{-1} + U_{nj}V_{jv}^{-1} + U_{nn}V_{nv}^{-1} \\
&= \frac{1}{D}\left[P_{nv}V_{ii} + P_{ni}(-V_{iv})\right] \\
&= \frac{1}{D}P_{nv}\left[-\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t} + \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t} + 1\right] \\
&\quad - \frac{1}{D}P_{ni}\left[-\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} + \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t} + 1\right] \\
&= \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left\{-P_{nv}\left(P_{vi}-P_{ii}\right) + P_{ni}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left\{P_{nv}\left(P_{vi}-P_{ii}\right) - P_{ni}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)\right\} \\
&\quad + \frac{1}{D}\frac{1}{2}\sqrt{\frac{\alpha}{q}}2\sqrt{\frac{q}{\alpha}}(P_{nv}-P_{ni}) \qquad \text{(A.54)} \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{nv}\left(P_{vi}-P_{ii}\right) + P_{ni}\left(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}}\right)\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}\left(P_{vi}-P_{ii}\right) - P_{ni}\left(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}}\right)\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{nv}-P_{ni}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&= \frac{\left\{\begin{array}{c} e^{\sqrt{q\alpha}t}\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}P_{vi} - P_{nv}P_{ii} - P_{ni}P_{vv} + P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\} \\ +2\sqrt{\frac{q}{\alpha}}(P_{nv}-P_{ni}) \end{array}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned}$$

14. $\mathbf{P}_{ni}$ $(= \mathbf{P}_{in})$

$$\begin{aligned}
\mathbf{P}_{ni} &= U_{nv}V_{vi}^{-1} + U_{ni}V_{ii}^{-1} + U_{nj}V_{ji}^{-1} + U_{nn}V_{ni}^{-1} \\
&= \frac{1}{D}\left[P_{nv}(-V_{vi}) + P_{ni}V_{vv}\right] \\
&= -\frac{1}{D}P_{nv}\left\{\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vi}-P_{ii})e^{-\sqrt{q\alpha}t}\right\} \\
&\quad + \frac{1}{D}P_{ni}\left\{\tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}+\sqrt{\tfrac{q}{\alpha}}\right)e^{\sqrt{q\alpha}t} - \tfrac{1}{2}\sqrt{\tfrac{\alpha}{q}}\left(P_{vv}-P_{iv}-\sqrt{\tfrac{q}{\alpha}}\right)e^{-\sqrt{q\alpha}t}\right\} \\
&= \frac{\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[\begin{matrix} e^{\sqrt{q\alpha}t}\left\{-P_{nv}(P_{vi}-P_{ii}) + P_{ni}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}(P_{vi}-P_{ii}) - P_{ni}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{matrix}\right]}{D} \\
&= \frac{\left\{\begin{matrix} e^{\sqrt{q\alpha}t}\left\{-P_{nv}(P_{vi}-P_{ii}) + P_{ni}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\} \\ +e^{-\sqrt{q\alpha}t}\left\{P_{nv}(P_{vi}-P_{ii}) - P_{ni}(P_{vv}-P_{iv}-\sqrt{\frac{q}{\alpha}})\right\}\end{matrix}\right\}}{e^{\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} \\
&\Rightarrow \frac{\left\{-P_{nv}(P_{vi}-P_{ii}) + P_{ni}(P_{vv}-P_{iv}+\sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right]}
\end{aligned} \tag{A.55}$$

15. $\mathbf{P}_{nj}$

$$\begin{aligned}
\mathbf{P}_{nj} &= U_{nv}V_{vj}^{-1} + U_{ni}V_{ij}^{-1} + U_{nj}V_{jj}^{-1} + U_{nn}V_{nj}^{-1} \\
&= \frac{1}{D}\left\{P_{nv}(-V_{vj}) + P_{ni}(V_{vj}) + P_{nj}(V_{vv}-V_{vi}) + P_{nn}\cdot 0\right\} \\
&= \frac{(P_{ni}-P_{nv})V_{vj}}{D} + P_{nj} \\
&= \frac{(P_{ni}-P_{nv})\left[\frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{\sqrt{q\alpha}t} - \frac{1}{2}\sqrt{\frac{\alpha}{q}}(P_{vj}-P_{ij})e^{-\sqrt{q\alpha}t}\right]}{\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right) - \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}\left(P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right)} + P_{nj} \\
&\Rightarrow \frac{(P_{ni}-P_{nv})(P_{vj}-P_{ij})}{\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right)} + P_{nj} \\
&= \frac{(P_{ji}-P_{jv})(P_{vn}-P_{in})}{\left(P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right)} + P_{jn}
\end{aligned} \tag{A.56}$$

16. $\mathbf{P}_{nn}$

$$
\begin{aligned}
\mathbf{P}_{nn} &= U_{nv}V_{vn}^{-1} + U_{ni}V_{in}^{-1} + U_{nj}V_{jn}^{-1} + U_{nn}V_{nn}^{-1} \\
&= \frac{1}{D}\left[P_{nv}(-V_{vn}) + P_{ni}V_{vn} + P_{nn}(V_{vv} - V_{vi})\right] \\
&= \frac{1}{D}\left[P_{nv}(-V_{vn}) + P_{ni}V_{vn}\right] + P_{nn} \\
&= \frac{V_{vn}(P_{ni} - P_{nv})}{D} + P_{nn} \\
&= \frac{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}(P_{ni} - P_{nv})^2 + \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}(P_{ni} - P_{nv})^2}{D} + P_{nn} \\
&= \frac{-\frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{\sqrt{q\alpha}t}(P_{ni}-P_{nv})^2 + \frac{1}{2}\sqrt{\frac{\alpha}{q}}e^{-\sqrt{q\alpha}t}(P_{ni}-P_{nv})^2}{e^{\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{vv}-2P_{iv}+P_{ii}+\sqrt{\frac{q}{\alpha}}\right] - e^{-\sqrt{q\alpha}t}\frac{1}{2}\sqrt{\frac{\alpha}{q}}\left[P_{vv}-2P_{iv}+P_{ii}-\sqrt{\frac{q}{\alpha}}\right]} + P_{nn}
\end{aligned}
\tag{A.57}
$$

$$
\Rightarrow \frac{-(P_{ni} - P_{nv})^2}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} + P_{nn}
$$

Limiting value of each element

1. P_{vv}

$$\lim_{t\to\infty} P_{vv} = \sqrt{\frac{q}{\alpha}} + \frac{\left[P_{ii}P_{vv} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}}\right]}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.58}$$

2. P_{vi}

$$\lim_{t\to\infty} P_{vi} = \frac{\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.59}$$

3. P_{vj}

$$\lim_{t\to\infty} P_{vj} = \frac{\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.60}$$

4. P_{vn}

$$\lim_{t\to\infty} P_{vn} = \frac{\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.61}$$

5. P_{iv}

$$\lim_{t\to\infty} P_{iv} = \frac{\left(P_{ii}P_{vv} - P_{iv}P_{vi} + P_{ii}\sqrt{\frac{q}{\alpha}}\right)}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.62}$$

6. P_{ii}

$$\lim_{t\to\infty} P_{ii} = \frac{\left[P_{vv}P_{ii} - P_{iv}^2 + P_{ii}\sqrt{\frac{q}{\alpha}})\right]}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.63}$$

7. P_{ij}

$$\lim_{t\to\infty} P_{ij} = \frac{\left\{-P_{jv}(P_{vi} - P_{ii}) + P_{ji}(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.64}$$

8. P_{in}

$$\lim_{t\to\infty} P_{in} = \frac{\left\{-P_{nv}(P_{vi} - P_{ii}) + P_{ni}(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.65}$$

9. P_{jv}

$$\lim_{t\to\infty} P_{jv} = \frac{\left\{-P_{jv}P_{vi} + P_{jv}P_{ii} + P_{ji}P_{vv} - P_{ji}P_{iv} + P_{ji}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.66}$$

10. P_{ji}

$$\lim_{t\to\infty} P_{ji} = \frac{\left\{-P_{jv}(P_{vi} - P_{ii}) + P_{ji}(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.67}$$

11. P_{jj}

$$\lim_{t\to\infty} P_{jj} = \frac{-(P_{ji} - P_{jv})^2}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} + P_{jj} \tag{A.68}$$

12. P_{jn}

$$\lim_{t\to\infty} P_{jn} = \frac{(P_{ji} - P_{jv})(P_{vn} - P_{in})}{\left(P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right)} + P_{jn} \tag{A.69}$$

13. P_{nv}

$$\lim_{t\to\infty} P_{nv} = \frac{\left\{-P_{nv}P_{vi} + P_{nv}P_{ii} + P_{ni}P_{vv} - P_{ni}P_{iv} + P_{ni}\sqrt{\frac{q}{\alpha}}\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.70}$$

14. P_{ni}

$$\lim_{t\to\infty} P_{ni} = \frac{\left\{-P_{nv}(P_{vi} - P_{ii}) + P_{ni}(P_{vv} - P_{iv} + \sqrt{\frac{q}{\alpha}})\right\}}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} \tag{A.71}$$

15. P_{jn}

$$\lim_{t\to\infty} P_{jn} = \frac{(P_{ni} - P_{nv})(P_{vj} - P_{ij})}{\left(P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right)} + P_{nj} \tag{A.72}$$

16. P_{nn}

$$\lim_{t\to\infty} P_{nn} = \frac{-(P_{ni} - P_{nv})^2}{\left[P_{vv} - 2P_{iv} + P_{ii} + \sqrt{\frac{q}{\alpha}}\right]} + P_{nn} \tag{A.73}$$

A.2 Two Observed Features and One Unobserved Feature

We next present details of the derivation of the MonoRob solution for two observed feature (features i and j) and one unobserved feature (feature n).

- Dynamic Model

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}\left(\mathbf{u}(t) + \mathbf{w}(t)\right) \tag{A.74}$$

  where, $\mathbf{F} = \mathbf{0}$, and $\mathbf{G} = [1\ 0\ \cdots\ 0]^T$

- Measurement Model

$$\hat{\mathbf{z}}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{v} \tag{A.75}$$

  where, $\mathbf{z}(t) = [z_i z_j]^T(t)$, $\mathbf{v} = [v_i v_j]^T$, $\mathbf{H} = \left[\begin{smallmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \end{smallmatrix}\right]$

- miscellaneous

$$\begin{aligned} \mathbf{x}(t) &= [x_v(t)\ x_1(t)\ \cdots\ x_n(t)]^T \\ \mathbf{Q} &= E[\mathbf{w}\mathbf{w}^T] = q \\ \mathbf{R} &= E[\mathbf{v}\mathbf{v}^T] = \left[\begin{smallmatrix} r_i & 0 \\ 0 & r_j \end{smallmatrix}\right] \end{aligned} \tag{A.76}$$

- Solution by the differential Riccati equation

$$\begin{aligned} \dot{\mathbf{P}}(t) &= \mathbf{F}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T - \mathbf{P}(t)^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}(t) \\ \mathbf{P}(t) &= \mathbf{U}(t)\mathbf{V}^{-1}(t) \end{aligned} \tag{A.77}$$

  where $\mathbf{U}$ and $\mathbf{V}$ satisfy the following:

$$\begin{bmatrix} \dot{\mathbf{U}}(t) \\ \dot{\mathbf{V}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{F} & \mathbf{G}\mathbf{Q}\mathbf{G}^T \\ \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} & -\mathbf{F}^T \end{bmatrix} \begin{bmatrix} \mathbf{U}(t) \\ \mathbf{V}(t) \end{bmatrix} \tag{A.78}$$

  with,

$$\begin{bmatrix} \mathbf{U}(0) \\ \mathbf{V}(0) \end{bmatrix} = \begin{bmatrix} \mathbf{P}(0) \\ \mathbf{I} \end{bmatrix} \tag{A.79}$$

A.2.1 $\dot{\mathbf{U}}$ and $\dot{\mathbf{V}}$

$$\begin{aligned}\dot{\mathbf{U}}(t) = GQG^T\mathbf{V}(t) &= \begin{bmatrix} q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_{vv}(t) & V_{vi}(t) & V_{vj}(t) & V_{vn} \\ V_{iv}(t) & V_{ii}(t) & V_{ij}(t) & V_{in} \\ V_{jv}(t) & V_{ji}(t) & V_{jj}(t) & V_{jn} \\ V_{nv}(t) & V_{ni}(t) & V_{nj}(t) & V_{nn} \end{bmatrix} \\ &= \begin{bmatrix} qV_{vv}(t) & qV_{vi}(t) & qV_{vj}(t) & qV_{vn} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\end{aligned} \tag{A.80}$$

$$\begin{aligned}\dot{\mathbf{V}}(t) &= H^T R^{-1} H\mathbf{U}(t) \\ &= \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{r_i} & 0 \\ 0 & \frac{1}{r_j} \end{bmatrix} \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} U_{vv}(t) & U_{vi}(t) & U_{vj}(t) & U_{vn}(t) \\ U_{iv}(t) & U_{ii}(t) & U_{ij}(t) & U_{in}(t) \\ U_{jv}(t) & U_{ji}(t) & U_{jj}(t) & U_{jn}(t) \\ U_{nv}(t) & U_{ni}(t) & U_{nj}(t) & U_{nn}(t) \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{r_i}+\frac{1}{r_j} & -\frac{1}{r_i} & -\frac{1}{r_j} & 0 \\ -\frac{1}{r_i} & \frac{1}{r_i} & 0 & 0 \\ -\frac{1}{r_j} & 0 & \frac{1}{r_j} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} U_{vv}(t) & U_{vi}(t) & U_{vj}(t) & U_{vn}(t) \\ U_{iv}(t) & U_{ii}(t) & U_{ij}(t) & U_{in}(t) \\ U_{jv}(t) & U_{ji}(t) & U_{jj}(t) & U_{jn}(t) \\ U_{nv}(t) & U_{ni}(t) & U_{nj}(t) & U_{nn}(t) \end{bmatrix} \\ &= \begin{bmatrix} \left(\frac{1}{r_i}+\frac{1}{r_j}\right)U_{vv} - \frac{1}{r_i}U_{iv} - \frac{1}{r_j}U_{jv} & \left(\frac{1}{r_i}+\frac{1}{r_j}\right)U_{vi} - \frac{1}{r_i}U_{ii} - \frac{1}{r_j}U_{ji} & \left(\frac{1}{r_i}+\frac{1}{r_j}\right)U_{vj} - \frac{1}{r_i}U_{ij} - \frac{1}{r_j}U_{jj} & \left(\frac{1}{r_i}+\frac{1}{r_j}\right)U_{vn} - \frac{1}{r_i}U_{in} - \frac{1}{r_j}U_{jn} \\ \frac{1}{r_i}(U_{iv}-U_{vv}) & \frac{1}{r_i}(U_{ii}-U_{vi}) & \frac{1}{r_i}(U_{ij}-U_{vj}) & \frac{1}{r_i}(U_{in}-U_{vn}) \\ \frac{1}{r_j}(U_{jv}-U_{vv}) & \frac{1}{r_j}(U_{ji}-U_{vi}) & \frac{1}{r_j}(U_{jj}-U_{vj}) & \frac{1}{r_j}(U_{jn}-U_{vn}) \\ 0 & 0 & 0 & 0 \end{bmatrix}\end{aligned} \tag{A.81}$$

A.2.2 Solution of U and V

1. $U_{vv}(t)$

$$\begin{aligned}\dot{U}_{vv}(t) &= qV_{vv}(t) \\ \ddot{U}_{vv}(t) &= q\dot{V}_{vv}(t) \\ &= q\left\{\left(\frac{1}{r_i}+\frac{1}{r_j}\right)U_{vv} - \frac{1}{r_i}U_{iv} - \frac{1}{r_j}U_{jv}\right\} \\ \ddot{U}_{vv}(t) - qIU_{vv}(t) + q\left(\frac{P_{iv}}{r_i}+\frac{P_{jv}}{r_j}\right) &= 0 \\ U_{vv}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{iv}}{r_i}+\frac{P_{jv}}{r_j}\right)\end{aligned} \tag{A.82}$$

where,

$$\begin{aligned} \alpha &= \sqrt{qI} \\ I &= \frac{1}{r_i} + \frac{1}{r_j} = \frac{r_i + r_j}{r_i r_j} \end{aligned} \tag{A.83}$$

By the boundary condition,

$$\begin{aligned} U_{vv}(0) = P_{vv} &= A + B + \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) \\ A + B &= P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) \end{aligned} \tag{A.84}$$

its first order derivative, and its boundary condition

$$\begin{aligned} \dot{U}_{vv}(t) &= A\alpha e^{\alpha t} - B\alpha e^{-\alpha t} \\ \dot{U}_{vv}(0) = qV_{vv}(0) = q &= (A - B)\alpha \\ A - B &= \frac{q}{\alpha} \end{aligned} \tag{A.85}$$

it follows that

$$\begin{aligned} A &= \frac{1}{2}\left\{P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) + \frac{q}{\alpha}\right\} \\ B &= \frac{1}{2}\left\{P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) - \frac{q}{\alpha}\right\} \end{aligned} \tag{A.86}$$

$$\begin{aligned} U_{vv}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) \\ &= \frac{1}{2}\left\{P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) + \frac{q}{\alpha}\right\} e^{\alpha t} \\ &\quad + \frac{1}{2}\left\{P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) - \frac{q}{\alpha}\right\} e^{-\alpha t} \\ &\quad + \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) \\ &= \left(P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right)\right)\cosh(\alpha t) + \frac{q}{\alpha}\sinh(\alpha t) + \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j}\right) \end{aligned} \tag{A.87}$$

2. $U_{vi}(t)$

$$\begin{aligned}
\dot{U}_{vi}(t) &= qV_{vi}(t) \\
\ddot{U}_{vi}(t) &= q\dot{V}_{vi}(t) \\
&= q\left\{IU_{vi} - \frac{1}{r_i}U_{ii} - \frac{1}{r_j}U_{ji}\right\} \\
\ddot{U}_{vi}(t) - qIU_{vi}(t) + q\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right) &= 0 \\
U_{vi}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)
\end{aligned} \tag{A.88}$$

By Boundary Condition,

$$\begin{aligned}
U_{vi}(0) = P_{vi} &= A + B + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right) \\
A + B &= P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)
\end{aligned} \tag{A.89}$$

Its first order derivative, and its boundary condition

$$\begin{aligned}
\dot{U}_{vi}(t) &= A\alpha e^{\alpha t} - B\alpha e^{-\alpha t} \\
\dot{U}_{vi}(0) = qV_{vi}(0) = 0 &= (A - B)\alpha \\
A - B &= 0
\end{aligned} \tag{A.90}$$

Thus,

$$\begin{aligned}
A &= \frac{1}{2}\left\{P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right\} \\
B &= \frac{1}{2}\left\{P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right\}
\end{aligned} \tag{A.91}$$

$$\begin{aligned}
U_{vi}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right) \\
&= \frac{1}{2}\left\{P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right\}e^{\alpha t} + \frac{1}{2}\left\{P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right) \\
&= \left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)\cosh(\alpha t) + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)
\end{aligned} \tag{A.92}$$

3. $U_{vj}(t)$

$$\begin{aligned}
\dot{U}_{vj}(t) &= qV_{vj}(t) \\
\ddot{U}_{vj}(t) &= q\dot{V}_{vj}(t) \\
&= q\left\{IU_{vj} - \frac{1}{r_i}U_{ij} - \frac{1}{r_j}U_{jj}\right\} \\
\ddot{U}_{vj}(t) - qIU_{vj}(t) + q\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right) &= 0 \\
U_{vj}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)
\end{aligned} \tag{A.93}$$

By Boundary Condition,

$$\begin{aligned}
U_{vj}(0) = P_{vj} &= A + B + \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right) \\
A + B &= P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)
\end{aligned} \tag{A.94}$$

Its first order derivative, and its boundary condition

$$\begin{aligned}
\dot{U}_{vj}(t) &= A\alpha e^{\alpha t} - B\alpha e^{-\alpha t} \\
\dot{U}_{vj}(0) = qV_{vj}(0) = 0 &= (A - B)\alpha \\
A - B &= 0
\end{aligned} \tag{A.95}$$

Thus,

$$\begin{aligned}
A &= \frac{1}{2}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\} \\
B &= \frac{1}{2}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}
\end{aligned} \tag{A.96}$$

$$\begin{aligned}
U_{vj}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right) \\
&= \frac{1}{2}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} + \frac{1}{2}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right) \\
&= \left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)\cosh(\alpha t) + \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)
\end{aligned} \tag{A.97}$$

4. $U_{vn}(t)$

$$\begin{aligned}
\dot{U}_{vn}(t) &= qV_{vn}(t) \\
\ddot{U}_{vn}(t) &= q\dot{V}_{vn}(t) \\
&= q\left\{IU_{vn} - \frac{1}{r_i}U_{in} - \frac{1}{r_j}U_{jn}\right\} \\
\ddot{U}_{vn}(t) - qIU_{vn}(t) + q\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right) &= 0 \\
U_{vn}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)
\end{aligned} \tag{A.98}$$

By Boundary Condition,

$$\begin{aligned}
U_{vn}(0) = P_{vn} &= A + B + \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right) \\
A + B &= P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)
\end{aligned} \tag{A.99}$$

Its first order derivative, and its boundary condition

$$\begin{aligned}
\dot{U}_{vn}(t) &= A\alpha e^{\alpha t} - B\alpha e^{-\alpha t} \\
\dot{U}_{vn}(0) = qV_{vn}(0) = 0 &= (A - B)\alpha \\
A - B &= 0
\end{aligned} \tag{A.100}$$

Thus,

$$\begin{aligned}
A &= \frac{1}{2}\left\{P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right\} \\
B &= \frac{1}{2}\left\{P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right\}
\end{aligned} \tag{A.101}$$

$$\begin{aligned}
U_{vn}(t) &= Ae^{\alpha t} + Be^{-\alpha t} + \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right) \\
&= \tfrac{1}{2}\left\{P_{vn} - \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j}\right)\right\}e^{\alpha t} + \tfrac{1}{2}\left\{P_{vn} - \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j}\right)\right\}e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j}\right) \\
&= \left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)\cosh(\alpha t) + \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)
\end{aligned} \tag{A.102}$$

5. $U_{iv}(t)$, $U_{ii}(t)$, $U_{ij}(t)$, and $U_{in}(t)$

$$\begin{aligned}
\dot{U}_{iv}(t) = 0 &\rightarrow U_{iv}(t) = P_{iv} \\
\dot{U}_{ii}(t) = 0 &\rightarrow U_{ii}(t) = P_{ii} \\
\dot{U}_{ij}(t) = 0 &\rightarrow U_{ij}(t) = P_{ij} \\
\dot{U}_{in}(t) = 0 &\rightarrow U_{in}(t) = P_{in}
\end{aligned} \tag{A.103}$$

6. $U_{jv}(t)$, $U_{ji}(t)$, $U_{jj}(t)$, and $U_{jn}(t)$

$$\begin{aligned}
\dot{U}_{jv}(t) = 0 &\rightarrow U_{jv}(t) = P_{jv} \\
\dot{U}_{ji}(t) = 0 &\rightarrow U_{ji}(t) = P_{ji} \\
\dot{U}_{jj}(t) = 0 &\rightarrow U_{ji}(t) = P_{jj} \\
\dot{U}_{jn}(t) = 0 &\rightarrow U_{jn}(t) = P_{jn}
\end{aligned} \tag{A.104}$$

7. $U_{nv}(t)$, $U_{ni}(t)$, $U_{nj}(t)$, and $U_{nn}(t)$

$$\begin{aligned}
\dot{U}_{nv}(t) = 0 &\rightarrow U_{nv}(t) = P_{nv} \\
\dot{U}_{ni}(t) = 0 &\rightarrow U_{ni}(t) = P_{ni} \\
\dot{U}_{nj}(t) = 0 &\rightarrow U_{ni}(t) = P_{nj} \\
\dot{U}_{nn}(t) = 0 &\rightarrow U_{nn}(t) = P_{nn}
\end{aligned} \tag{A.105}$$

8. $V_{vv}(t)$

$$\begin{aligned}
V_{vv}(t) &= \frac{1}{q}\dot{U}_{vv}(t) \\
&= \tfrac{1}{q}\left\{\tfrac{1}{2}\alpha\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)+\tfrac{q}{\alpha}\right)e^{\alpha t}-\tfrac{1}{2}\alpha\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)-\tfrac{q}{\alpha}\right)e^{-\alpha t}\right\} \\
&= \tfrac{1}{2}\tfrac{\alpha}{q}\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)+\tfrac{q}{\alpha}\right)e^{\alpha t}-\tfrac{1}{2}\tfrac{\alpha}{q}\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)-\tfrac{q}{\alpha}\right)e^{-\alpha t} \\
&= \frac{\alpha}{q}\left(P_{vv}-\frac{1}{I}\left(\frac{P_{iv}}{r_i}+\frac{P_{jv}}{r_j}\right)\right)\sinh(\alpha t)+\cosh(\alpha t)
\end{aligned} \tag{A.106}$$

9. $V_{vi}(t)$

$$\begin{aligned}
V_{vi}(t) &= \frac{1}{q}\dot{U}_{vi}(t) \\
&= \frac{1}{q}\left\{\frac{1}{2}\alpha\left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\alpha\left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)e^{-\alpha t}\right\} \\
&= \frac{1}{2}\frac{\alpha}{q}\left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\frac{\alpha}{q}\left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)e^{-\alpha t} \\
&= \frac{\alpha}{q}\left(P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j}\right)\right)\sinh(\alpha t)
\end{aligned} \tag{A.107}$$

10. $V_{vj}(t)$

$$\begin{aligned}
V_{vj}(t) &= \frac{1}{q}\dot{U}_{vj}(t) \\
&= \frac{1}{q}\left\{\frac{1}{2}\alpha\left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\alpha\left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)e^{-\alpha t}\right\} \\
&= \frac{1}{2}\frac{\alpha}{q}\left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\frac{\alpha}{q}\left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)e^{-\alpha t} \\
&= \frac{\alpha}{q}\left(P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right)\sinh(\alpha t)
\end{aligned} \tag{A.108}$$

11. $V_{vn}(t)$

$$\begin{aligned}
V_{vn}(t) &= \frac{1}{q}\dot{U}_{vn}(t) \\
&= \frac{1}{q}\left\{\frac{1}{2}\alpha\left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\alpha\left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)e^{-\alpha t}\right\} \\
&= \frac{1}{2}\frac{\alpha}{q}\left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)e^{\alpha t} - \frac{1}{2}\frac{\alpha}{q}\left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)e^{-\alpha t} \\
&= \frac{\alpha}{q}\left(P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j}\right)\right)\sinh(\alpha t)
\end{aligned} \tag{A.109}$$

12. $V_{iv}(t)$

$$
\begin{aligned}
\dot{V}_{iv}(t) &= \frac{1}{r_i}(U_{iv}(t) - U_{vv}(t)) \\
&= \frac{1}{r_i}\left\{ P_{iv} - \left[\tfrac{1}{2}\left\{ P_{vv} - \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) + \tfrac{q}{\alpha} \right\} e^{\alpha t} + \tfrac{1}{2}\left\{ P_{vv} - \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) - \tfrac{q}{\alpha} \right\} e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv}}{r_i} - \frac{1}{r_i}\frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) \\
&= -\frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv}}{r_i}\left(1 - \frac{1}{r_i}\frac{1}{I} \right) - \frac{1}{r_i}\frac{1}{I}\frac{P_{jv}}{r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv}}{r_i}\left(1 - \frac{r_j}{r_i + r_j} \right) - \frac{P_{jv}}{r_i + r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_i}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv} - P_{jv}}{r_i + r_j} \\
V_{iv} &= -\frac{1}{2r_i\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv} - P_{jv}}{r_i + r_j} t + C \\
V_{iv}(0) = 0 &= -\tfrac{1}{2r_i\alpha}\left\{ P_{vv} - \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) + \tfrac{q}{\alpha} \right\} + \tfrac{1}{2r_i\alpha}\left\{ P_{vv} - \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) - \tfrac{q}{\alpha} \right\} + C \\
&= -\frac{1}{2r_i\alpha}\frac{q}{\alpha} - \frac{1}{2r_i\alpha}\frac{q}{\alpha} + C \\
C &= \frac{q}{r_i\alpha^2} \\
V_{iv}(t) &= -\frac{1}{2r_i\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{iv} - P_{jv}}{r_i + r_j} t + \frac{q}{r_i\alpha^2} \\
&= -\tfrac{1}{r_i\alpha}\left\{ P_{vv} - \tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i} + \tfrac{P_{jv}}{r_j} \right) \right\} \sinh(\alpha t) - \tfrac{q}{r_i\alpha^2}\cosh(\alpha t) + \tfrac{P_{iv} - P_{jv}}{r_i + r_j} t + \tfrac{q}{r_i\alpha^2}
\end{aligned}
\tag{A.110}
$$

13. $V_{ii}(t)$

$$
\begin{aligned}
\dot{V}_{ii}(t) &= \frac{1}{r_i}(U_{ii}(t) - U_{vi}(t)) \\
&= \frac{1}{r_i}\left\{ P_{ii} - \left[\frac{1}{2}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} + \frac{1}{2}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ii}}{r_i} - \frac{1}{r_i}\frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \\
&= -\frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ii}}{r_i}\left(1 - \frac{1}{r_i}\frac{1}{I} \right) - \frac{1}{r_i}\frac{1}{I}\frac{P_{ji}}{r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ii}}{r_i}\left(1 - \frac{r_j}{r_i + r_j} \right) - \frac{P_{ji}}{r_i + r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ii} - P_{ji}}{r_i + r_j} \\
V_{ii} &= -\frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{ii} - P_{ji}}{r_i + r_j}t + C \\
V_{ii}(0) = 1 &= -\frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} + \frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} + C \\
C &= 1 \\
V_{ii}(t) &= -\frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{ii} - P_{ji}}{r_i + r_j}t + 1 \\
&= -\frac{1}{r_i\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} \sinh(\alpha t) + \frac{P_{ii} - P_{ji}}{r_i + r_j}t + 1
\end{aligned}
\tag{A.111}
$$

14. $V_{ij}(t)$

$$
\begin{aligned}
\dot{V}_{ij}(t) &= \frac{1}{r_i}(U_{ij}(t) - U_{vj}(t)) \\
&= \frac{1}{r_i}\left\{ P_{ij} - \left[\tfrac{1}{2}\left\{P_{vj} - \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} + \tfrac{1}{2}\left\{P_{vj} - \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j}\right)\right] \right\} \\
&= -\frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} - \frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{P_{ij}}{r_i} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right) \\
&= -\frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} - \frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{P_{ij}}{r_i}\left(1 - \frac{1}{r_j}\frac{1}{I}\right) - \frac{1}{r_i}\frac{1}{I}\frac{P_{jj}}{r_j} \\
&= -\frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} - \frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{P_{ij}}{r_i}\left(1 - \frac{r_j}{r_i + r_j}\right) - \frac{P_{jj}}{r_i + r_j} \\
&= -\frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} - \frac{1}{2r_i}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} \\
&\quad + \frac{P_{ij} - P_{jj}}{r_i + r_j} \\
V_{ij} &= -\frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} + \frac{P_{ij} - P_{jj}}{r_i + r_j}t + C \\
V_{ij}(0) = 0 &= -\frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\} + \frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\} + C \\
C &= 0 \\
V_{ij}(t) &= -\frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}e^{-\alpha t} + \frac{P_{ij} - P_{jj}}{r_i + r_j}t \\
&= -\frac{1}{r_i\alpha}\left\{P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j}\right)\right\}\sinh(\alpha t) + \frac{P_{ij} - P_{jj}}{r_i + r_j}t
\end{aligned}
\tag{A.112}
$$

15. $V_{in}(t)$

$$
\begin{aligned}
\dot{V}_{in}(t) &= \frac{1}{r_i}(U_{in}(t) - U_{vn}(t)) \\
&= \frac{1}{r_i}\left\{ P_{in} - \left[\frac{1}{2}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} + \frac{1}{2}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \frac{1}{I}\left(\frac{P_{jn}}{r_i} + \frac{P_{jn}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{in}}{r_i} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \\
&= -\frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{in}}{r_i}\left(1 - \frac{1}{r_j}\frac{1}{I} \right) - \frac{1}{r_i}\frac{1}{I}\frac{P_{jn}}{r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{in}}{r_i}\left(1 - \frac{r_n}{r_i + r_j} \right) - \frac{P_{jn}}{r_i + r_j} \\
&= -\frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_i}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{in} - P_{jn}}{r_i + r_j} \\
V_{in} &= -\frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{in} - P_{jn}}{r_i + r_j}t + C \\
V_{in}(0) = 0 &= -\frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} + \frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} + C \\
C &= 0 \\
V_{in}(t) &= -\frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{in} - P_{jn}}{r_i + r_j}t \\
&= -\frac{1}{r_i\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} \sinh(\alpha t) + \frac{P_{in} - P_{jn}}{r_i + r_j}t
\end{aligned}
\tag{A.113}
$$

16. $V_{jv}(t)$

$$
\begin{aligned}
\dot{V}_{jv}(t) &= \frac{1}{r_j}(U_{jv}(t) - U_{vv}(t)) \\
&= \frac{1}{r_j}\left\{ P_{jv} - \left[\frac{1}{2}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} + \frac{1}{2}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv}}{r_j} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) \\
&= -\frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv}}{r_j}\left(1 - \frac{1}{r_j}\frac{1}{I} \right) - \frac{1}{r_j}\frac{1}{I}\frac{P_{iv}}{r_i} \\
&= -\frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv}}{r_j}\left(1 - \frac{r_i}{r_i + r_j} \right) - \frac{P_{iv}}{r_i + r_j} \\
&= -\frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad - \frac{1}{2r_j}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv} - P_{iv}}{r_i + r_j} \\
V_{jv} &= -\frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv} - P_{iv}}{r_i + r_j}t + C \\
V_{jv}(0) = 0 &= -\frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} + \frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} + C \\
&= -\frac{1}{2r_j\alpha}\frac{q}{\alpha} - \frac{1}{2r_j\alpha}\frac{q}{\alpha} + C \\
C &= \frac{q}{r_j\alpha^2} \\
V_{jv}(t) &= -\frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) + \frac{q}{\alpha} \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) - \frac{q}{\alpha} \right\} e^{-\alpha t} + \frac{P_{jv} - P_{iv}}{r_i + r_j}t + \frac{q}{r_j\alpha^2} \\
&= -\frac{1}{r_j\alpha}\left\{ P_{vv} - \frac{1}{I}\left(\frac{P_{iv}}{r_i} + \frac{P_{jv}}{r_j} \right) \right\} \sinh(\alpha t) - \frac{q}{r_j\alpha^2}\cosh(\alpha t) + \frac{P_{jv} - P_{iv}}{r_i + r_j}t + \frac{q}{r_j\alpha^2}
\end{aligned}
\tag{A.114}
$$

17. $V_{ji}(t)$

$$
\begin{aligned}
\dot{V}_{ji}(t) &= \frac{1}{r_j}(U_{ji}(t) - U_{vi}(t)) \\
&= \frac{1}{r_j}\left\{ P_{ji} - \left[\tfrac{1}{2}\left\{ P_{vi} - \tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i} + \tfrac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} + \tfrac{1}{2}\left\{ P_{vi} - \tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i} + \tfrac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i} + \tfrac{P_{ji}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ji}}{r_j} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \\
&= -\frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ji}}{r_j}\left(1 - \frac{1}{r_j}\frac{1}{I} \right) - \frac{1}{r_j}\frac{1}{I}\frac{P_{ii}}{r_i} \\
&= -\frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ji}}{r_j}\left(1 - \frac{r_i}{r_i + r_j} \right) - \frac{P_{ii}}{r_i + r_j} \\
&= -\frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{ji} - P_{ii}}{r_i + r_j} \\
V_{ji} &= -\frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{ji} - P_{ii}}{r_i + r_j} t + C \\
V_{ji}(0) = 0 &= -\frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} + \frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} + C \\
C &= 0 \\
V_{ji}(t) &= -\frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{ji} - P_{ii}}{r_i + r_j} t \\
&= -\frac{1}{r_j\alpha}\left\{ P_{vi} - \frac{1}{I}\left(\frac{P_{ii}}{r_i} + \frac{P_{ji}}{r_j} \right) \right\} \sinh(\alpha t) + \frac{P_{ji} - P_{ii}}{r_i + r_j} t
\end{aligned}
\tag{A.115}
$$

18. $V_{jj}(t)$

$$
\begin{aligned}
\dot{V}_{jj}(t) &= \frac{1}{r_j}(U_{jj}(t) - U_{vj}(t)) \\
&= \frac{1}{r_j}\left\{ P_{jj} - \left[\tfrac{1}{2}\left\{ P_{vj} - \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} + \tfrac{1}{2}\left\{ P_{vj} - \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i} + \tfrac{P_{jj}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jj}}{r_j} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \\
&= -\frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jj}}{r_j}\left(1 - \frac{1}{r_j}\frac{1}{I} \right) - \frac{1}{r_j}\frac{1}{I}\frac{P_{ij}}{r_i} \\
&= -\frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jj}}{r_j}\left(1 - \frac{r_i}{r_i + r_j} \right) - \frac{P_{ij}}{r_i + r_j} \\
&= -\frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jj} - P_{ij}}{r_i + r_j} \\
V_{jj} &= -\frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{jj} - P_{ij}}{r_i + r_j}t + C \\
V_{jj}(0) = 1 &= -\frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} + \frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} + C \\
C &= 1 \\
V_{jj}(t) &= -\frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{jj} - P_{ij}}{r_i + r_j}t + 1 \\
&= -\frac{1}{r_j\alpha}\left\{ P_{vj} - \frac{1}{I}\left(\frac{P_{ij}}{r_i} + \frac{P_{jj}}{r_j} \right) \right\} \sinh(\alpha t) + \frac{P_{jj} - P_{ij}}{r_i + r_j}t + 1
\end{aligned}
\tag{A.116}
$$

19. $V_{jn}(t)$

$$
\begin{aligned}
\dot{V}_{jn}(t) &= \frac{1}{r_j}(U_{jn}(t) - U_{vn}(t)) \\
&= \frac{1}{r_j}\left\{ P_{jn} - \left[\tfrac{1}{2}\left\{ P_{vn} - \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} + \tfrac{1}{2}\left\{ P_{vn} - \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \tfrac{1}{I}\left(\tfrac{P_{in}}{r_i} + \tfrac{P_{jn}}{r_j} \right) \right] \right\} \\
&= -\frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jn}}{r_j} - \frac{1}{r_j}\frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \\
&= -\frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jn}}{r_j}\left(1 - \frac{1}{r_j}\frac{1}{I} \right) - \frac{1}{r_j}\frac{1}{I}\frac{P_{in}}{r_i} \\
&= -\frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jn}}{r_j}\left(1 - \frac{r_i}{r_i + r_j} \right) - \frac{P_{in}}{r_i + r_j} \\
&= -\frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} - \frac{1}{2r_j}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} \\
&\quad + \frac{P_{jn} - P_{in}}{r_i + r_j} \\
V_{jn} &= -\frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{jn} - P_{in}}{r_i + r_j} t + C \\
V_{jn}(0) = 0 &= -\frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} + \frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} + C \\
C &= 0 \\
V_{jn}(t) &= -\frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{\alpha t} \\
&\quad + \frac{1}{2r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} e^{-\alpha t} + \frac{P_{jn} - P_{in}}{r_i + r_j} t \\
&= -\frac{1}{r_j\alpha}\left\{ P_{vn} - \frac{1}{I}\left(\frac{P_{in}}{r_i} + \frac{P_{jn}}{r_j} \right) \right\} \sinh(\alpha t) + \frac{P_{jn} - P_{in}}{r_i + r_j} t
\end{aligned}
\tag{A.117}
$$

20. V_{nv}, V_{ni}, V_{nj}, and V_{nn}

$$\begin{aligned}
\dot{V}_{nv}(t) = 0 &\rightarrow V_{nv}(t) = 0 \\
\dot{V}_{ni}(t) = 0 &\rightarrow V_{ni}(t) = 0 \\
\dot{V}_{nj}(t) = 0 &\rightarrow V_{nj}(t) = 0 \\
\dot{V}_{nn}(t) = 0 &\rightarrow V_{nn}(t) = 1
\end{aligned} \tag{A.118}$$

A.2.3 Inverse of $\mathbf{V}(t)$

a. Structure of $\mathbf{V}(t)$

Let $\mathbf{V}(t)$ be $\begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ V_{iv} & V_{ii} & V_{ij} & V_{in} \\ V_{jv} & V_{ji} & V_{jj} & V_{jn} \\ V_{nv} & V_{ni} & V_{nj} & V_{nn} \end{bmatrix}$. Then,

$$\begin{aligned}
\mathbf{V}(t) &= \begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ V_{iv} & V_{ii} & V_{ij} & V_{in} \\ V_{jv} & V_{ji} & V_{jj} & V_{jn} \\ V_{nv} & V_{ni} & V_{nj} & V_{nn} \end{bmatrix} \\
&= \begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ \frac{P_{iv}-P_{jv}}{r_i+r_j}t+\frac{q}{r_i\alpha^2}(1-V_{vv}) & \frac{P_{ii}-P_{ji}}{r_i+r_j}t+\frac{q}{r_i\alpha^2}(-V_{vi})+1 & \frac{P_{ij}-P_{jj}}{r_i+r_j}t+\frac{q}{r_i\alpha^2}(-V_{vj}) & \frac{P_{in}-P_{jn}}{r_i+r_j}t+\frac{q}{r_i\alpha^2}(-V_{vn}) \\ -\frac{P_{iv}-P_{jv}}{r_i+r_j}t+\frac{q}{r_j\alpha^2}(1-V_{vv}) & -\frac{P_{ii}-P_{ji}}{r_i+r_j}t+\frac{q}{r_j\alpha^2}(-V_{vi}) & -\frac{P_{ij}-P_{jj}}{r_i+r_j}t+\frac{q}{r_j\alpha^2}(-V_{vj})+1 & -\frac{P_{in}-P_{jn}}{r_i+r_j}t+\frac{q}{r_j\alpha^2}(-V_{vn}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} V_{vv} & V_{vi} & V_{vj} & V_{vn} \\ \phi(iv,jv)+\frac{q}{r_i\alpha^2}(1-V_{vv}) & \phi(ii,ji)+\frac{q}{r_i\alpha^2}(-V_{vi})+1 & \phi(ij,jj)+\frac{q}{r_i\alpha^2}(-V_{vj}) & \phi(in,jn)+\frac{q}{r_i\alpha^2}(-V_{vn}) \\ -\phi(iv,jv)+\frac{q}{r_j\alpha^2}(1-V_{vv}) & -\phi(ii,ji)+\frac{q}{r_j\alpha^2}(-V_{vi}) & -\phi(ij,jj)+\frac{q}{r_j\alpha^2}(-V_{vj})+1 & -\phi(in,jn)+\frac{q}{r_j\alpha^2}(-V_{vn}) \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{A.119}$$

A.2.4 Determinant of $\mathbf{V}(t)$

$$-V_{iv}V_{jj}V_{vi} + V_{ij}V_{jv}V_{vi} + V_{iv}V_{ji}V_{vj} - V_{ii}V_{jv}V_{vj} - V_{ij}V_{ji}V_{vv} + V_{ii}V_{jj}V_{vv}$$

$$
\begin{aligned}
Det(V) = {} & V_{vv}\left\{\phi(ii,ji)-\phi(ij,jj)+1\right\} \\
& + V_{vi}\left\{\phi(ij,jj)-\phi(iv,jv)-\frac{q}{r_i\alpha^2}\right\} \\
& + V_{vj}\left\{\phi(iv,jv)-\phi(ii,ji)-\frac{q}{r_j\alpha^2}\right\}
\end{aligned}
$$

$$
\begin{aligned}
& \left\{\tfrac{\alpha}{q}\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)\right)\sinh(\alpha t)+\cosh(\alpha t)\right\}\left\{\tfrac{P_{ii}-P_{ij}}{r_i+r_j}t-\tfrac{P_{ij}-P_{jj}}{r_i+r_j}t+1\right\} \\
= {} & \quad + \left\{\tfrac{\alpha}{q}\left(P_{vi}-\tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i}+\tfrac{P_{ji}}{r_j}\right)\right)\sinh(\alpha t)\right\}\left\{\tfrac{P_{ij}-P_{jj}}{r_i+r_j}t-\tfrac{P_{vi}-P_{vj}}{r_i+r_j}t-\tfrac{q}{r_i\alpha^2}\right\} \\
& \quad + \left\{\tfrac{\alpha}{q}\left(P_{vj}-\tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i}+\tfrac{P_{jj}}{r_j}\right)\right)\sinh(\alpha t)\right\}\left\{\tfrac{P_{vi}-P_{vj}}{r_i+r_j}t-\tfrac{P_{ii}-P_{ij}}{r_i+r_j}t-\tfrac{q}{r_j\alpha^2}\right\} \\
& \left\{\tfrac{\alpha}{q}\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)\right)\sinh(\alpha t)+\cosh(\alpha t)\right\}\left\{\tfrac{P_{ii}-P_{ij}}{r_i+r_j}t-\tfrac{P_{ij}-P_{jj}}{r_i+r_j}t+1\right\} \\
= {} & \quad + \left\{\tfrac{\alpha}{q}\left(P_{vi}-\tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i}+\tfrac{P_{ji}}{r_j}\right)\right)\sinh(\alpha t)\right\}\left\{\tfrac{P_{ij}-P_{jj}}{r_i+r_j}t-\tfrac{P_{vi}-P_{vj}}{r_i+r_j}t-\tfrac{q}{r_i\alpha^2}\right\} \\
& \quad + \left\{\tfrac{\alpha}{q}\left(P_{vj}-\tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i}+\tfrac{P_{jj}}{r_j}\right)\right)\sinh(\alpha t)\right\}\left\{\tfrac{P_{vi}-P_{vj}}{r_i+r_j}t-\tfrac{P_{ii}-P_{ij}}{r_i+r_j}t-\tfrac{q}{r_j\alpha^2}\right\}
\end{aligned}
$$

$$
\begin{aligned}
= {} & \frac{\alpha}{q}\sinh(\alpha t)\left\{\begin{array}{l}
\left(P_{vv}-\tfrac{1}{I}\left(\tfrac{P_{iv}}{r_i}+\tfrac{P_{jv}}{r_j}\right)\right)\left\{\tfrac{P_{ii}-2P_{ij}+P_{jj}}{r_i+r_j}t+1\right\} \\
+\left(P_{vi}-\tfrac{1}{I}\left(\tfrac{P_{ii}}{r_i}+\tfrac{P_{ji}}{r_j}\right)\right)\left\{\tfrac{P_{ij}-P_{jj}-P_{vi}+P_{vj}}{r_i+r_j}t-\tfrac{q}{r_i\alpha^2}\right\} \\
+\left(P_{vj}-\tfrac{1}{I}\left(\tfrac{P_{ij}}{r_i}+\tfrac{P_{jj}}{r_j}\right)\right)\left\{\tfrac{P_{vi}-P_{vj}-P_{ii}+P_{ij}}{r_i+r_j}t-\tfrac{q}{r_j\alpha^2}\right\}
\end{array}\right\} \\
& + \cosh(\alpha t)\left\{\frac{P_{ii}-2P_{ij}+P_{jj}}{r_i+r_j}t+1\right\}
\end{aligned}
$$

$$
\begin{aligned}
= {} & \frac{\alpha}{q}\sinh(\alpha t)\left\{\begin{array}{l}
\left(P_{vv}-\tfrac{r_jP_{iv}+r_iP_{jv}}{r_i+r_j}\right)\left\{\tfrac{P_{ii}-2P_{ij}+P_{jj}}{r_i+r_j}t+1\right\} \\
+\left(P_{vi}-\tfrac{r_jP_{ii}+r_iP_{ji}}{r_i+r_j}\right)\left\{\tfrac{P_{ij}-P_{jj}-P_{vi}+P_{vj}}{r_i+r_j}t-\tfrac{q}{r_i\alpha^2}\right\} \\
+\left(P_{vj}-\tfrac{r_jP_{ij}+r_iP_{jj}}{r_i+r_j}\right)\left\{\tfrac{P_{vi}-P_{vj}-P_{ii}+P_{ij}}{r_i+r_j}t-\tfrac{q}{r_j\alpha^2}\right\}
\end{array}\right\} \\
& + \cosh(\alpha t)\left\{\frac{P_{ii}-2P_{ij}+P_{jj}}{r_i+r_j}t+1\right\}
\end{aligned}
$$

$$
\begin{aligned}
= {} & \frac{\alpha}{q}\frac{\sinh(\alpha t)}{(r_i+r_j)^2}\left\{\begin{array}{l}
\{(r_i+r_j)P_{vv}-r_jP_{iv}-r_iP_{jv}\}\{(P_{ii}-2P_{ij}+P_{jj})t+(r_i+r_j)\} \\
+\{(r_i+r_j)P_{vi}-r_jP_{ii}-r_iP_{ji}\}\{(P_{ij}-P_{jj}-P_{vi}+P_{vj})t-r_j\} \\
+\{(r_i+r_j)P_{vj}-r_jP_{ij}-r_iP_{jj}\}\{(P_{vi}-P_{vj}-P_{ii}+P_{ij})t-r_i\}
\end{array}\right\} \\
& + \cosh(\alpha t)\left\{\frac{P_{ii}-2P_{ij}+P_{jj}}{r_i+r_j}t+1\right\}
\end{aligned}
$$

(A.120)

A.2.5 Structure of $\mathbf{V}(t)$ inverse

$$\mathbf{V}(t) = \frac{1}{|\mathbf{V}|} \begin{bmatrix} -V_{ij}V_{ji} + V_{ii}V_{jj} & -V_{jj}V_{vi} + V_{ji}V_{vj} & V_{ij}V_{vi} - V_{ii}V_{vj} & \begin{matrix} V_{in}V_{jj}V_{vi} - V_{ij}V_{jn}V_{vi} - V_{in}V_{ji}V_{vj} \\ +V_{ii}V_{jn}V_{vj} + V_{ij}V_{ji}V_{vn} - V_{ii}V_{jj}V_{vn} \end{matrix} \\ -V_{iv}V_{jj} + V_{ij}V_{jv} & -V_{jv}V_{vj} + V_{jj}V_{vv} & V_{iv}V_{vj} - V_{ij}V_{vv} & \begin{matrix} -V_{iv}V_{jn}V_{vj} + V_{in}V_{jv}V_{vj} + V_{iv}V_{jj}V_{vn} \\ -V_{ij}V_{jv}V_{vn} - V_{in}V_{jj}V_{vv} + V_{ij}V_{jn}V_{vv} \end{matrix} \\ V_{iv}V_{ji} - V_{ii}V_{jv} & V_{jv}V_{vi} - V_{ji}V_{vv} & -V_{iv}V_{vi} + V_{ii}V_{vv} & \begin{matrix} +V_{iv}V_{jn}V_{vi} - V_{in}V_{jv}V_{vi} - V_{iv}V_{ji}V_{vn} \\ +V_{ii}V_{jv}V_{vn} + V_{in}V_{ji}V_{vv} - V_{ii}V_{jn}V_{vv} \end{matrix} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.121}$$

A.2.6 Summary of the elements of the $\mathbf{V}(t)$ inverse

In the following, $\frac{1}{|\mathbf{V}|}$ is omitted.

1. $V_{vv}^{-1} = V_{vi}\left\{\phi(ij,jj) - \frac{q}{r_i\alpha^2}\right\} + V_{vj}\left\{-\phi(ii,ji) - \frac{q}{r_j\alpha^2}\right\} + \{\phi(ii,ji) - \phi(ij,jj) + 1\}$
2. $V_{iv}^{-1} = V_{vv}\left\{\frac{q}{r_i\alpha^2} - \phi(ij,jj)\right\} + V_{vj}\{\phi(iv,jv)\} + \left\{\phi(ij,jj) - \phi(iv,jv) - \frac{q}{r_i\alpha^2}\right\}$
3. $V_{jv}^{-1} = V_{vv}\left\{\phi(ii,ji) + \frac{q}{r_j\alpha^2}\right\} + V_{vi}\{-\phi(iv,jv)\} + \left\{\phi(iv,jv) - \phi(ii,ji) - \frac{q}{r_j\alpha^2}\right\}$
4. $V_{nv}^{-1} = 0$
5. $V_{vi}^{-1} = V_{vi}\{\phi(ij,jj) - 1\} + V_{vj}\{-\phi(ii,ji)\}$
6. $V_{ii}^{-1} = V_{vv}\{1 - \phi(ij,jj)\} + V_{vj}\left\{-\frac{q}{r_j\alpha^2} + \phi(iv,jv)\right\}$
7. $V_{ji}^{-1} = V_{vv}\{\phi(ii,ji)\} + V_{vi}\left\{\frac{q}{r_j\alpha^2} - \phi(iv,jv)\right\}$
8. $V_{ni}^{-1} = 0$
9. $V_{vj}^{-1} = V_{vi}\{\phi(ij,jj)\} + V_{vj}\{-\phi(ii,ji) - 1\}$
10. $V_{ij}^{-1} = V_{vv}\{-\phi(ij,jj)\} + V_{vj}\left\{\frac{q}{r_i\alpha^2} + \phi(iv,jv)\right\}$
11. $V_{jj}^{-1} = V_{vv}\{1 + \phi(ii,ji)\} + V_{vi}\left\{-\frac{q}{r_i\alpha^2} - \phi(iv,jv)\right\}$
12. $V_{nj}^{-1} = 0$
13. $V_{vn}^{-1} = V_{vi}\{\phi(in,jn)\} + V_{vj}\{-\phi(in,jn)\} + V_{vn}\{\phi(ij,jj) - \phi(ii,ji) - 1\}$
14. $V_{in}^{-1} = V_{vv}\{-\phi(in,jn)\} + V_{vj}\{\phi(in,jn)\} + V_{vn}\left\{\phi(iv.jv) - \phi(ij,jj) + \frac{q}{r_i\alpha^2}\right\}$
15. $V_{jn}^{-1} = V_{vv}\{+\phi(in,jn)\} + V_{vi}\{-\phi(in,jn)\} + V_{vn}\left\{\phi(ii,ji) - \phi(iv,jv) + \frac{q}{r_j\alpha^2}\right\}$
16. $V_{nn}^{-1} = 1$
17.

$$|\mathbf{V}(t)| = V_{vv}\{\phi(ii,ji) - \phi(ij,jj) + 1\} + V_{vi}\left\{\phi(ij,jj) - \phi(iv,jv) - \frac{q}{r_i\alpha^2}\right\} + V_{vj}\left\{\phi(iv,jv) - \phi(ii,ji) - \frac{q}{r_j\alpha^2}\right\} \tag{A.122}$$

A.2.7 Elements of the covariance matrix

1. $P_{vv}(t)$

$$
\begin{aligned}
P_{vv}(t) = & \sinh(\alpha t)\frac{q}{\alpha} + \frac{t\sinh(\alpha t)}{(r_i+r_j)}\frac{q}{\alpha}\left(P_{ii} - 2P_{ij} + P_{jj}\right) \\
& + \cosh(\alpha t)P_{vv} \\
& + \frac{t\cosh(\alpha t)}{(r_i+r_j)}\left\{P_{ii}P_{vv} + P_{vv}P_{jj} + 2P_{vj}P_{vi} - 2P_{ij}P_{vv} - P_{vi}^2 - P_{vj}^2\right\} \\
& - \frac{(1-e^{\alpha t})(1-e^{-\alpha t})}{(r_i+r_j)^2}\left\{\begin{array}{c}(r_i+r_j)t\left[-P_{ij}^2 + P_{ij}P_{vi} - P_{vj}P_{ii} + P_{vj}P_{ij} + P_{jj}P_{ii} - P_{jj}P_{vi}\right] \\ +r_i^2\left[P_{jj} - P_{vj}\right] + r_j^2\left[P_{ii} - P_{vi}\right] + r_i r_j(2P_{ij} - P_{vi} - P_{vj})\end{array}\right\} \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q}\left\{\begin{array}{c}t(r_i+r_j)\left\{P_{ij}^2P_{vv} + P_{vj}^2P_{ii} + P_{vi}^2P_{jj} - 2P_{vi}P_{vj}P_{ij} - P_{jj}P_{vv}P_{ii}\right\} \\ +2r_ir_j\left\{P_{vi}P_{vj} - P_{ij}P_{vv}\right\} + r_i^2\left(P_{vj}^2 - P_{jj}P_{vv}\right) + r_j^2\left(P_{vi}^2 - P_{ii}P_{vv}\right)\end{array}\right\}
\end{aligned}
\tag{A.123}
$$

2. $P_{ii}(t)$

$$
\begin{aligned}
P_{ii}(t) = & \cosh(\alpha t)P_{ii} \\
& + \frac{\cosh(\alpha t)}{(r_i+r_j)}t\left(P_{ii}P_{jj} - P_{ij}^2\right) \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q}\left\{\begin{array}{c}t(r_i+r_j)\left\{P_{vi}^2P_{jj} - 2P_{ij}P_{vi}P_{vj} + P_{ij}^2P_{vv} + P_{ii}P_{vj}^2 - P_{ii}P_{jj}P_{vv}\right\} \\ +(r_i+r_j)^2\left(P_{vi}^2 - P_{ii}P_{vv}\right) + 2r_i(r_i+r_j)\left(P_{ii}P_{vj} - P_{ij}P_{vi}\right) \\ +r_i^2\left(P_{ij}^2 - P_{ii}P_{jj}\right)\end{array}\right\}
\end{aligned}
\tag{A.124}
$$

3. $P_{jj}(t)$

$$
\begin{aligned}
P_{jj}(t) = & \cosh(\alpha t)P_{jj} \\
& + \frac{\cosh(\alpha t)}{(r_i+r_j)}t\left(P_{ii}P_{jj} - P_{ij}^2\right) \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q}\left\{\begin{array}{c}t(r_i+r_j)\left\{P_{vj}^2P_{ii} - 2P_{ij}P_{vj}P_{vi} + P_{ij}^2P_{vv} + P_{jj}P_{vi}^2 - P_{vv}P_{ii}P_{jj}\right\} \\ (r_i+r_j)^2\left(P_{vj}^2 - P_{jj}P_{vv}\right) + 2r_j(r_i+r_j)\left(P_{jj}P_{vi} - P_{ij}P_{vj}\right) \\ +r_j^2\left(P_{ij}^2 - P_{jj}P_{ii}\right)\end{array}\right\}
\end{aligned}
\tag{A.125}
$$

4. $P_{nn}(t)$

$$
\begin{aligned}
P_{nn}(t) &= \cosh(\alpha t) P_{nn} \\
&\quad + \frac{\cosh(\alpha t)}{(r_i + r_j)} t \left\{ P_{nn}P_{ii} + P_{nn}P_{jj} + 2P_{in}P_{jn} - 2P_{ij}P_{nn} - P_{in}^2 - P_{jn}^2 \right\} \\
&\quad - \frac{\sinh(\alpha t)}{(r_i + r_j)^2} \frac{\alpha}{q} \left(\begin{array}{c}
\begin{array}{c} r_i^2(P_{vn}^2 + P_{jn}^2 + 2P_{nn}P_{vj} - P_{nn}P_{vv} - P_{jj}P_{nn} - 2P_{jn}P_{vn}) \\ + r_j^2(P_{vn}^2 + P_{in}^2 + 2P_{nn}P_{vi} - P_{nn}P_{vv} - P_{ii}P_{nn} - 2P_{in}P_{vn}) \end{array} \\
+2r_i r_j(P_{vn}^2 + P_{nn}P_{vi} + P_{nn}P_{vj} + P_{in}P_{jn} - P_{ij}P_{nn} - P_{nn}P_{vv} - P_{in}P_{vn} - P_{jn}P_{vn}) \\
+t(r_i + r_j) \left\{ \begin{array}{c}
P_{ij}^2 P_{nn} - 2P_{ij}P_{nn}P_{vi} + P_{nn}P_{vi}^2 - 2P_{in}^2 P_{vj} \\
+2P_{ii}P_{nn}P_{vj} - 2P_{ij}P_{nn}P_{vj} - 2P_{nn}P_{vi}P_{vj} + P_{nn}P_{vj}^2 \\
+2P_{ij}P_{in}P_{vn} - 2P_{in}P_{vi}P_{vn} + 2P_{in}P_{vj}P_{vn} + P_{ii}P_{vn}^2 \\
-2P_{ij}P_{vn}^2 + P_{in}^2 P_{vv} - P_{ii}P_{nn}P_{vv} + 2P_{ij}P_{nn}P_{vv} \\
+P_{jn}^2 P_{ii} - 2P_{jn}^2 P_{vi} + P_{jn}^2 P_{vv} + P_{jj}P_{in}^2 \\
-2P_{jj}P_{in}P_{vn} + P_{jj}P_{vn}^2 - P_{jj}P_{nn}P_{ii} + 2P_{jj}P_{nn}P_{vi} - P_{jj}P_{nn}P_{vv} \\
-2P_{jn}P_{vn}P_{ii} + 2P_{jn}P_{vn}P_{ij} + 2P_{jn}P_{vn}P_{vi} - 2P_{jn}P_{vn}P_{vj} \\
-2P_{jn}P_{in}P_{ij} + 2P_{jn}P_{in}P_{vi} + 2P_{jn}P_{in}P_{vj} - 2P_{jn}P_{in}P_{vv}
\end{array} \right\}
\end{array} \right)
\end{aligned}
$$

(A.126)

5. $P_{vi}(t)$

$$
\begin{aligned}
P_{vi}(t) &= P_{vi} \\
&\quad + \frac{e^{-\alpha t}}{2(r_i + r_j)} \left(\begin{array}{l} (1 - e^{\alpha t})^2 \left\{ P_{ii}r_j + P_{ij}r_i + (P_{ii}P_{jj} - P_{ij}^2)t \right\} \\ +2te^{\alpha t}(+P_{vi}P_{jj} + P_{ii}P_{vj} - P_{ij}P_{vi} - P_{ij}P_{vj}) \end{array} \right) \\
&\quad - \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i + r_j)^2} \frac{\alpha}{q} \left(\begin{array}{c}
r_i^2(P_{vi}P_{vj} + P_{ij}P_{vj} - P_{jj}P_{vi} - P_{ij}P_{vv}) \\
+r_j^2\left(P_{vi}^2 - P_{ii}P_{vv}\right) \\
+r_i r_j\left(P_{vi}^2 + P_{ii}P_{vj} + P_{vi}P_{vj} - P_{ii}P_{vv} - P_{ij}P_{vi} - P_{ij}P_{vv}\right) \\
+t(r_i + r_j)\left(P_{ii}P_{vj}^2 + P_{ij}^2 P_{vv} + P_{jj}P_{vi}^2 - P_{jj}P_{ii}P_{vv} - 2P_{ij}P_{vj}P_{vi}\right)
\end{array} \right)
\end{aligned}
$$

(A.127)

6. $P_{vj}(t)$

$$
\begin{aligned}
P_{vj}(t) &= P_{vj} \\
&\quad + \frac{e^{-\alpha t}}{2(r_i + r_j)} \left(\begin{array}{l} (1 - e^{\alpha t})^2 \left\{ P_{ij}r_j + P_{jj}r_i + (P_{ii}P_{jj} - P_{ij}^2)t \right\} \\ +2te^{\alpha t}\left\{ P_{vi}P_{jj} + P_{vj}P_{ii} - P_{vj}P_{ij} - P_{ij}P_{vi} \right\} \end{array} \right) \\
&\quad - \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i + r_j)^2} \frac{\alpha}{q} \left(\begin{array}{c}
+r_i^2\left\{ P_{vj}^2 - P_{jj}P_{vv} \right\} \\
+r_j^2\{P_{ij}P_{vi} + P_{vj}P_{vi} - P_{vj}P_{ii} - P_{ij}P_{vv}\} \\
+r_i r_j\left\{ P_{vj}^2 + P_{jj}P_{vi} + P_{vj}P_{vi} - P_{ij}P_{vj} - P_{jj}P_{vv} - P_{ij}P_{vv} \right\} \\
+t(r_i + r_j)\left\{ P_{ii}P_{vj}^2 + P_{ij}^2 P_{vv} + P_{jj}P_{vi}^2 - 2P_{ij}P_{vj}P_{vi} - P_{ii}P_{jj}P_{vv} \right\}
\end{array} \right)
\end{aligned}
$$

(A.128)

7. $P_{vn}(t)$

$$
\begin{aligned}
P_{vn}(t) = {} & P_{vn} \\
& + \frac{e^{-\alpha t}}{2(r_i+r_j)}(1-e^{\alpha t})^2 \left\{P_{in}r_j + P_{jn}r_i + t\left(P_{in}P_{jj} + P_{ii}P_{jn} - P_{ij}P_{jn} - P_{in}P_{ij}\right)\right\} \\
& + \frac{t}{(r_i+r_j)} \left\{P_{vi}P_{jn} + P_{vj}P_{in} + P_{vn}P_{ii} + P_{vn}P_{jj} - 2P_{vn}P_{ij} - P_{vj}P_{jn} - P_{vi}P_{in}\right\} \\
& - \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i+r_j)^2}\frac{\alpha}{q} \left(\begin{array}{c}
+r_i^2\{P_{jn}P_{vj}+P_{vj}P_{vn}-P_{jj}P_{vn}-P_{jn}P_{vv}\} \\
+r_j^2\{P_{in}P_{vi}+P_{vi}P_{vn}-P_{ii}P_{vn}-P_{in}P_{vv}\} \\
+r_ir_j\{P_{in}P_{vj}+P_{jn}P_{vi}+P_{vi}P_{vn}+P_{vj}P_{vn}-P_{in}P_{vv}-P_{jn}P_{vv}-2P_{ij}P_{vn}\} \\
+t(r_i+r_j)\left\{ \begin{array}{c}
+P_{jn}P_{vi}^2+P_{jn}P_{vj}P_{ii}+P_{jn}P_{vv}P_{ij} \\
+P_{in}P_{jj}P_{vi}+P_{in}P_{vj}^2+P_{ij}^2P_{vn}+P_{jj}P_{vi}P_{vn} \\
+P_{ii}P_{vj}P_{vn}+P_{ij}P_{in}P_{vv}-P_{ij}P_{vj}P_{vn}-P_{in}P_{jj}P_{vv} \\
-P_{ij}P_{in}P_{vj}-P_{ii}P_{jj}P_{vn}-P_{ij}P_{vi}P_{vn}-P_{in}P_{vi}P_{vj} \\
-P_{jn}P_{ij}P_{vi}-P_{jn}P_{vj}P_{vi}-P_{jn}P_{vv}P_{ii}
\end{array} \right\}
\end{array} \right)
\end{aligned}
\tag{A.129}
$$

8. $P_{ij}(t)$

$$
\begin{aligned}
P_{ij}(t) = {} & \cosh(\alpha t)P_{ij} \\
& + \frac{\cosh(\alpha t)}{(r_i+r_j)} t\left(P_{ii}P_{jj} - P_{ij}^2\right) \\
& - \frac{\sinh(\alpha t)}{(r_i+r_j)^2}\frac{\alpha}{q} \left\{ \begin{array}{c}
t(r_i+r_j)\left[P_{vv}P_{ij}^2 + P_{ii}P_{vj}^2 + P_{jj}P_{vi}^2 - 2P_{ij}P_{vi}P_{vj} - P_{vv}P_{ii}P_{jj}\right] \\
+(r_i+r_j)^2\left[P_{vi}P_{vj} - P_{vv}P_{ij}\right] + r_ir_j(P_{jj}P_{ii} - P_{ij}^2) \\
+(r_i+r_j)\left[r_i(P_{ij}P_{vj} - P_{jj}P_{vi}) + r_j(P_{ij}P_{vi} - P_{vj}P_{ii})\right]
\end{array} \right\}
\end{aligned}
\tag{A.130}
$$

9. $P_{in}(t)$

$$
\begin{aligned}
P_{in}(t) = {} & \frac{(e^{\alpha t} + e^{-\alpha t})}{2}P_{in} \\
& + \frac{(e^{\alpha t} + e^{-\alpha t})}{2(r_i+r_j)} t\left(P_{ii}P_{jn} + P_{in}P_{jj} - P_{ij}P_{jn} - P_{in}P_{ij}\right) \\
& + \frac{(e^{\alpha t} - e^{-\alpha t})}{2(r_i+r_j)^2}\frac{\alpha}{q} \left\{ \begin{array}{c}
r_i^2\left\{ \begin{array}{c} P_{jn}P_{vi} + P_{in}P_{vv} + P_{in}P_{jj} + P_{ij}P_{vn} \\ -P_{vi}P_{vn} - 2P_{in}P_{vj} - P_{ij}P_{jn} \end{array} \right\} \\
+r_j^2\{P_{ii}P_{vn}+P_{in}P_{vv}-P_{vi}P_{vn}-P_{in}P_{vi}\} \\
+r_ir_j\left\{ \begin{array}{c} P_{jn}P_{vi}+P_{ii}P_{vn}+P_{in}P_{ij}+2P_{in}P_{vv} \\ +P_{ij}P_{vn}-2P_{vi}P_{vn}-P_{ii}P_{jn}-P_{in}P_{vi}-2P_{in}P_{vj} \end{array} \right\} \\
+t(r_i+r_j)\left\{ \begin{array}{c}
P_{in}P_{vj}P_{ij}+P_{in}P_{vj}P_{vi}+P_{in}P_{jj}P_{vv}+P_{ij}P_{vn}P_{vi}+P_{ij}P_{vn}P_{vj} \\
+P_{jn}P_{vi}P_{vj}+P_{ii}P_{jn}P_{vv}+P_{ij}P_{jn}P_{vi}+P_{ii}P_{jj}P_{vn}-P_{jn}P_{vi}^2 \\
-P_{ii}P_{jn}P_{vj}-P_{jj}P_{vi}P_{vn}-P_{ii}P_{vj}P_{vn}-P_{ij}^2P_{vn}-P_{in}P_{ij}P_{vv} \\
-P_{in}P_{vj}^2-P_{in}P_{jj}P_{vi}-P_{ij}P_{jn}P_{vv}
\end{array} \right\}
\end{array} \right\}
\end{aligned}
\tag{A.131}
$$

10. $P_{jn}(t)$

$$
\begin{aligned}
P_{jn}(t) = & \frac{(e^{\alpha t}+e^{-\alpha t})}{2} P_{jn} \\
& + \frac{(e^{\alpha t}+e^{-\alpha t})}{2(r_i+r_j)} t \left\{ P_{in}P_{jj} + P_{jn}P_{ii} - P_{in}P_{ij} - P_{jn}P_{ij} \right\} \\
& - \frac{(e^{\alpha t}-e^{-\alpha t})}{2(r_i+r_j)^2} \frac{\alpha}{q} \left\{ \begin{array}{c}
r_i^2\{ P_{vj}P_{vn}+P_{jn}P_{vj}-P_{jn}P_{vv}-P_{jj}P_{vn} \} \\
+r_j^2\{ P_{ij}P_{in}+P_{vj}P_{vn}+2P_{jn}P_{vi}-P_{in}P_{vj}-P_{ij}P_{vn}-P_{jn}P_{vv}-P_{jn}P_{ii} \} \\
+r_i r_j \left\{ \begin{array}{c} 2P_{vj}P_{vn}-2P_{jn}P_{vv}+2P_{jn}P_{vi}+P_{jn}P_{vj}+P_{jj}P_{in} \\ -P_{in}P_{vj}-P_{ij}P_{vn}-P_{jn}P_{ij}-P_{jj}P_{vn} \end{array} \right\} \\
+t(r_i+r_j) \left\{ \begin{array}{c} P_{in}P_{vj}^2+P_{ij}^2P_{vn}+P_{ii}P_{vj}P_{vn}+P_{jn}P_{vi}^2+P_{jn}P_{vv}P_{ij} \\ +P_{jn}P_{vj}P_{ii}+P_{jj}P_{vn}P_{vi}+P_{jj}P_{in}P_{vi}+P_{ij}P_{in}P_{vv}-P_{ij}P_{in}P_{vj} \\ -P_{in}P_{vi}P_{vj}-P_{ij}P_{vi}P_{vn}-P_{ij}P_{vj}P_{vn}-P_{jn}P_{ij}P_{vi}-P_{jn}P_{vv}P_{ii} \\ -P_{jn}P_{vj}P_{vi}-P_{jj}P_{vn}P_{ii}-P_{jj}P_{in}P_{vv} \end{array} \right\}
\end{array} \right\}
\end{aligned}
\tag{A.132}
$$

Bibliography

[1] Ravindra K. Ahuja, Thomas L. Magananti, and James B. Orlin. *Network Flows.* Prentice Hall, 1993.

[2] Brian D.O. Anderson and John B. Moore. *Optimal Control, Linear Quadratic Methods.* Prentice-Hall, 1990.

[3] David Austin and Patric Jensfelt. Using multiple gaussian hypotheses to represent probability distributions for mobile robot localization. In *IEEE Intl. Conf. on Robotics and Automation*, 2000.

[4] T. Bailey and E. Nebot. Localization in large-scale environments. *Robotics and Autonomous Systems*, 37(4):261–281, 2001.

[5] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association.* Academic Press, 1988.

[6] Yaakov Bar-Shalom and Xiao-Rong Li. *Estimation and Tracking: Principles, Technique, and Software.* YBS, 1998.

[7] Yaakov Bar-Shalom, Xiao-Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation.* Wiley, 2001.

[8] Dimitri P. Bertsekas and John N. Tsitsiklis. *Introduction to Probability.* Athena Scientific, 2002.

[9] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller. An atlas framework for scalable mapping. In *Proc. IEEE Int. Conf. Robotics and Automation*, 2003.

[10] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller. An atlas framework for scalable mapping. *Int. J. Robotics Research*, 2004.

[11] Michael Carsten Bosse. *ATLAS: A Framework for Large Scale Automated Mapping and Localization.* PhD thesis, Massachusetts Institute of Technology, 2004.

[12] R. A Brooks. Aspects of mobile robot visual map making. In *Second Int. Symp. Robotics Research*, pages 287–293, Tokyo, Japan, 1984. MIT Press.

[13] Robert Grover Brown and Patrick Y.C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering.* Wiley, third edition, 1997.

[14] J. D. Tardos C. Estrada, J. Neira. Hierarchical slam: real-time metric mapping of large environment, 2004.

[15] J. A. Castellanos, J. M. M. Montiel, J. Neira, and J. D. Tardos. The SPmap: A probabilistic framework for simultaneous localization and map building. *IEEE Trans. Robotics and Automation*, 15(5):948–952, 1999.

[16] J. A. Castellanos, J. Neira, and J. D. Tardos. Multisensor fusion for simultaneous localization and map building. *IEEE Trans. Robotics and Automation*, 17(6):908–914, 2001.

[17] J. A. Castellanos and J. D. Tardos. *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach.* Kluwer Academic Publishers, Boston, 2000.

[18] J. A. Castellanos, J. D. Tardos, and G. Schmidt. Building a global map of the environment of a mobile robot: The importance of correlations. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 1053–1059, 1997.

[19] K. S. Chong and L. Kleeman. Sonar based map building in large indoor environments. Technical Report MECSE-1997-1, Department of Electrical and Computer Systems Engineering, Monash University, 1997.

[20] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. *ieeetra*, 17(2):125–137, 2001.

[21] H. Choset and K. Nagatani. Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization. *IEEE Transactions on Robotic and Automation*, 17(2):125–137, April 2001.

[22] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms.* MIT Press, second edition, 2001.

[23] Michael Csorba. *Simultaneous Localisation and Map Building.* PhD thesis, University of Oxford, 1997.

[24] A. J. Davison. *Mobile Robot Navigation Using Active Vision.* PhD thesis, University of Oxford, 1998.

[25] F. Dellaert, D. Fox, and and S. Thrun W. Burgard. Monte carlo localization for mobile robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, 1999.

[26] M. W. M. G. Dissanayake, P. Newman, H. F. Durrant-Whyte, S. Clark, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. Technical Report ACFR-TR-01-99, University of Sydney, March 1999.

[27] M. W. M. G. Dissanayake, P. Newman, H. F. Durrant-Whyte, S. Clark, and M. Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on Robotic and Automation*, 17(3):229–241, June 2001.

[28] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice.* Springer, 2001.

[29] H. Durrant-Whyte, S. Majumder, M. Battista, and S. Scheding. A bayesian algorithm for simultaneous localisation and map building. Technical report, University of Sydney, ??

[30] James R. Evans and Edward Minieka. *Optimization Algorithms for Networks and Graphs.* Marcel Dekker, second edition, 1992.

[31] H. J. S. Feder and J. J. Leonard. Decoupled stochastic mapping part II: performance analysis, 1999. Submitted for consideration for publication to the IEEE Transactions on Robotics and Automation.

[32] H. J. S. Feder, J. J. Leonard, and C. M. Smith. Adaptive mobile robot navigation and mapping. *Int. J. Robotics Research*, 18(7):650–668, July 1999.

[33] Arthur Gelb. *Applied Optimal Estimation.* MIT Press, 1974.

[34] P. W. Gibbens, M. W. M. G. Dissanayake, and H. F. Durrant-Whyte. A closed form solution to the single degree of freedom simultaneous localisation and the map building (slam) problem. In *IEEE Int. Conference on Decision and Control (CDC)*, pages 191–196, Sydney, Australia, December 2000.

[35] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering, Theory and Practivce using MATLAB.* Wiley, second edition, 2001.

[36] J. Guivant and E. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. *IEEE Transactions on Robotic and Automation*, 2000. (submitted for consideration for publication, Novemeber 2000.

[37] J. Guivant and E. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. *IEEE Transactions on Robotic and Automation*, 17(3):242–257, June 2001.

[38] Jose Guivant. *Efficient Simultaneous Localization and Mapping in Large Environments.* PhD thesis, University of Sydney, 2002.

[39] J-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *International Symposium on Computational Intelligence in Robotics and Automation*, 1999.

[40] D. Hahnel, D. Schulz, and W. Burgard. Map building with mobile robots in populated environments. In *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, 2002.

[41] David Harville. *Matrix Algebra From A Statistian's Perspective.* Springer, 1997.

[42] Roger A. Horn and Charles R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[43] J. D. Tardos J. A. Castellanos, J. Neira. Limits to the consistency of ekf-based slam. In *5th IFAC Symposium on Intelligent Autonomous Vehicle, IAV*, 2004.

[44] S. J. Julier. A sparse weight kalman filter approach to simultaneous localisation and map building. In *Sensor Fusion and Decentralized Control in Robotic Systems IV.* SPIE, 2001.

[45] Dieter Jungnickel. *Graphs Networks and Algorithms*, volume 5 of *Algorithms and Computation in Mathematics.* Springer, 1999.

[46] T. Kailath, A. Sayed, and B. Hassibi. *Linear Estimation.* Prentice-Hall, 2000.

[47] J. Knight. *Towards Fully Autonomous Visual Navigation.* PhD thesis, University of Oxford, 2002.

[48] B. Kuipers and P. Beeson. Bootstrap learning for place recognition. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002. AAAI.

[49] B. J. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 2000.

[50] J. Leonard and P. Newman. Consistent, convergent, and constant-time SLAM. In *IJCAI*, 2003.

[51] J. J. Leonard and H. F. Durrant-Whyte. Simultaneous map building and localization for an autonomous mobile robot. In *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, pages 1442–1447, Osaka, Japan, 1991.

[52] J. J. Leonard and H. J. S. Feder. Decoupled stochastic mapping. Technical Report Marine Robotics Laboratory 99-1, Massachusetts Institute of Technology, 1999. http://oe.mit.edu/~jleonard/pubs/tr9901.ps.

[53] J. J. Leonard and H. J. S. Feder. Decoupled stochastic mapping part I: theory, 1999. Submitted for consideration for publication to the IEEE Transactions on Robotics and Automation.

[54] J. J. Leonard and H. J. S. Feder. A computationally efficient method for large-scale concurrent mapping and localization. In D Koditschek and J. Hollerbach, editors, *Robotics Research: The Ninth International Symposium*, pages 169–176, Snowbird, Utah, 2000. Springer Verlag.

[55] J. J. Leonard and H. J. S. Feder. Decoupled stochastic mapping. *IEEE J. Ocean Engineering*, 26(4):561–571, 2001.

[56] J. J. Leonard, R. J. Rikoski, P. M. Newman, and M. Bosse. Mapping partially observable features from multiple uncertain vantage points. *Int. J. Robotics Research*, 2002. To Appear.

[57] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping, 1997.

[58] Helmut Lutkepohl. *Handbook of Matrices.* Wiley, 1996.

[59] Peter S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, 1979.

[60] P. H. Milne. *Underwater Acoustic Positioning Systems.* London: E. F. N. Spon, 1983.

[61] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002. AAAI.

[62] P. Moutarlier and R. Chatila. Stochastic multi-sensory data fusion for mobile robot location and environment modeling. In *5th Int. Symposium on Robotics Research*, pages 207–216, 1989.

[63] J. Neira and J.D. Tardos. Data association in stochastic mapping: the fallacy of the nearest neighbor. *IEEE transactions on robotics and automation*, 20(Y), 1999.

[64] J. Neira and J.D. Tardós. Data association in stochastic mapping using the joint compatibility test. *IEEE Trans. on Robotics and Automation*, 17(6):890–897, 2001.

[65] J. Neira, J.D. Tardós, and J.A. Castellanos. Linear time vehicle relocation in SLAM. Technical Report RR-2002-08, Universidad de Zaragoza, Depto. de Informática e Ing. de Sistemas, Zaragoza, Spain, 2002.

[66] P. Newman, , M. Bosse, and J. Leonard. Autonomous feature-based exploration. In *Proc. IEEE Int. Conf. Robotics and Automation*, 2003.

[67] P. M. Newman. *On the structure and solution of the simultaneous localization and mapping problem.* PhD thesis, University of Sydney, 1999.

[68] Paul M. Newman, John J. Leonard, and Richard Rikoski. Towards constant-time slam on an autonomous underwater vehicle using synthetic aperture sonar. In *International Symposium on Robotics Research*, 2003.

[69] AOFNC Experimental Validation of the Moving Long Base Line Navigation Concept. Moving base line navigation. In *AUV 2004*, June 2004.

[70] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, third edition, 1991.

[71] Peyton Z. Peebles. *Probability, Random Variables, and Random Signal Principles.* McGraw-Hill, third edition, 1993.

[72] R. Rikoski and J. Leonard. Sonar trajectory perception. In *Proc. IEEE Int. Conf. Robotics and Automation*, Taiwan, 2003.

[73] Richard James Rikoski. *Dynamic Sonar Perception.* PhD thesis, Massachusetts Institute of Technology, 2003.

[74] R. Smith and P. Cheeseman. On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 5(4):56, 1987.

[75] R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *4th International Symposium on Robotics Research.* MIT Press, 1987.

[76] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I. Cox and G. Wilfong, editors, *Autonomous Robot Vehicles*, pages 167–193. Springer-Verlag, 1990.

[77] Gilbert Strang. *Introduction to Applied Mathematics.* Wellesley-Cambridge Press, 1986.

[78] J.D. Tardós, J. Neira, P.M. Newman, and J.J. Leonard. Robust mapping and localization in indoor environments using sonar data. *Int. J. Robotics Research*, 21(4):311–330, April 2002.

[79] J.D. Tardós, J. Neira, P.M. Newman, and J.J. Leonard. Robust mapping and localization in indoor environments using sonar data. *Int. J. Robotics Research*, 21(4):311–330, April 2002.

[80] Juan D. Tardos, Jose Neira, Paul M. Newman, and John J. Leonard. Robust mapping and localization in indoor environment using sonar data. *International Journal of Robotics Research*, December 2001.

[81] S. Thrun. An online mapping algorithm for teams of mobile robots. *Int. J. Robotics Research*, 2001. To Appear.

[82] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002. to appear.

[83] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. Technical Report CMU-CS-00-125, Carnegie Mellon University, 2000.

[84] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and Ng. A.Y. Simultaneous mapping and localization with sparse extended information filters. In J.-D. Boissonnat, J. Burdick, K. Goldberg, and S. Hutchinson, editors, *Proceedings of the Fifth International Workshop on Algorithmic Foundations of Robotics*, Nice, France, 2002. Forthcoming.

[85] Sebastian Thrun, Daphne Koller, Zoubin Ghahramani, Hugh Durrant-Whyte, and Andrew Y. Ng. Simultaneous mapping and localization with sparse extended information filters: Theory and initial results. Technical report, Carnegie Mellon University, Stanford University, University College London, University of Sydney, 2002.

[86] S.B Williams, G. Dissanayake, and H Durrant-Whyte. An efficient approach to the simultaneous localization and mapping problem. *Proc. IEEE Int. Conf. Robotics and Automation*, pages 406–411, 2002.

[87] Roy D. Yates and David J. Goodman. *Probability and Stochastic Processes*. Wiley, 1999.