ADAPTIVE MEMORY AND PARTICLE BASED SEQUENTIAL IMPLEMENTATION RESAMPLING FOR PARTICLE FILTERING

WAN MOHD YA'AKOB BIN WAN BEJURI

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy in (Computer Science)

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

JULY 2019

DEDICATION

This thesis is dedicated to my family, who taught me that the best kind of knowledge to have is that which is learned for its own sake. Although they are not involved in research and writing, but they involved in the journey. The list are; Dr. Sharifah Maryam binti Wan Hasan, Dr. Syed Bilal bin Wan Mohd Ya'akob, Dr. Wan Bejuri bin Wan Hamid, Dr. Sharifah binti Mohd Yusop, Dr. Sharifah Norkiah binti Wan Abang, Dr. Wan Hasan bin Syed Jamil, and Dr. Sharifah Habibah binti Wan Dillah.

ACKNOWLEDGEMENT

I would like to express grateful by saying praise be to Allah for the accomplishment of this thesis. I also would like to give special thanks to my supervisors; Assoc. Prof. Dr. Mohd Murtadha bin Mohamad and Dr. Raja Zahilah binti Raja Mohd Radzi for his support and guidance throughout this study. My sincere appreciation also extends to all my postgraduate friends especially team-mate who have provided assistance and discussions at various occasions. I am deeply indebted to my family who's my driving force towards the completion of this study. Their understanding and thoughtfulness throughout my study are appreciated. Last but not least, I would like to express my gratitude to Ministry of Education (MOE), Malaysia and Universiti Teknikal Malaysia Melaka (UTeM) for the financial support through the SLAI Scholarship.

ABSTRACT

The particle filter provides numerical approximation to a nonlinear filtering problem, especially during signal or data transmission. In a heterogeneous environment, reliable state estimation is a critical issue due to the unbalanced particle distribution called sample degeneracy and impoverishment. To address such a problem, sequential implementation resampling (SIR) considers the cause and environment of every specific resampling task decision. However, SIR only considers the cause and environment in a specific situation, which cannot generates reliable state estimation during filtering process. Apart from that, the developed SIR may suffer with unbalanced memory usage, which is reflected in the overall consumed system memory and time. Therefore, this research designed a resampling scheme that generates reliable state estimation and balances the resampling memory usage during particle filtering. To achieve this aim, an adaptive memory and particle sequential implementation resampling (AMPSIR) scheme was designed for different sample impoverishment environments, introduced three enhanced schemes to ensure reliable final state estimation and balanced theresampling memory allocation. The first scheme was the adaptive noise and sample size special strategies resampling (ANSSSR), which combined resampling task from three different types of special strategies resampling, and then reduced state estimation error in different situations in high sample impoverishment. Secondly, the scheme known as adaptive noise and sample size sequential implementation resampling (ANSSIR) combined resampling tasks from three different types of sequential implementation resampling, and then produced a reduction of state estimation error in different stages of sample impoverishment. Finally, the third scheme was the adaptive memory single distribution resampling (AMSDR), which combined resampling tasks from two different types of single distribution resampling, and then generated optimization of resampling memory. All of these enhanced schemes reacted based on measurement detection of particle noise, particle sample size and resampling memory. Simulation results showed that AMPSIR scheme achieved improved performance in termsof reducing state estimation error in different situations in high sample impoverishment by 7.26%, reduced state estimation error in different stages of sample impoverishment by 24.78%, and optimized resampling memory by 28.73% as compared to the existing resampling schemes. The findings showed that the AMPSIR scheme has the capability to do different kinds of resampling tasks, and choose a suitable scheme based on detected noise, sample size and memory measurements. In conclusion, the AMPSIR scheme has been proven to be a valuable solution for different sample impoverishment environments and different resampling memory usage. Besides, it has the ability to adapt the end user’s application memory usage with the scheme to determine the most suitable resampling scheme based on the application memory usage.

ABSTRAK

Penurasan zarah memberikan penganggaran berangka kepada permasalahan penurasan tidak linear, terutamanya ketika penghantaran zarah atau isyarat. Di dalam persekitaran yang pelbagai, pengiraan nilai yang dipercayai merupakan isu kritikal. Ini disebabkan oleh taburan zarah yang tidak seimbang atau dikenali sebagai kemerosotan dan ketidak pelbagaian berat sampel. Untuk mengatasinya, pensampelan semula perlaksanaan berjujukan (SIR) mengambilkira punca dan persekitaran untuk setiap hasil kerja pensampelan semula. Walaubagaimanapun, SIR cuma hanya mempertimbangkan punca dan persekitaran untuk situasi yang spesifik, yang seterusnya sebaliknya gagal dalam penghasilan pengiraan nilai yang boleh dipercayai semasa proses penurasan. Selain itu, SIR yang dibangunkan tidak terlepas daripada masalah penggunaan memori yang tidak seimbang, yang mana ianya boleh meninggalkan kesan terhadap penggunaan memori dan masa secara keseluruhan terhadap sistem. Oleh itu, penyelidikan kajian ini menghasilkan suatu skema pensampelan semula yang boleh dipercayai semasa penurasan zarah. Untuk mencapai matlamat ini, suatu skema pensampelan semula perlaksanaan berjujukan berdasarkan memori dan zarah mudahsuai (AMPSIR) bagi persekitaran ketidakpelbagaian berat sampel yang pelbagai, yang terdiri daripada tiga skema yang dipertingkatkan, bagi memastikan pengiraan nilai yang boleh dipercayai serta menyeimbangkan memori pensampelan semula. Skema yang pertama ialah, pensampelan semula strategi khas berdasarkan hingar dan saiz sampel mudah suai (ANSSSR), yang menggabungkan tiga jenis pensampelan semula strategi khas yang berbeza, dan kemudiannya menghasilkan pengurangan ralat pengiraan nilai di pelbagai situasi ketidakpelbagaian berat sampel peringkat tinggi. Skema yang kedua ialah, pensampelan semula perlaksanaan berjujukan berdasarkan hingar dan saiz sampel mudah suai (ANSSIR), yang menggabungkan tiga jenis pensampelan semula perlaksanaan berjujukan yang berbeza, dan kemudiannya menghasilkan pengurangan ralat pengiraan nilai di pelbagai peringkat ketidakpelbagaian berat sampel. Yang terakhir, skema yang ketiga, ialah, pensampelan semula teragih tunggal berdasarkan memori mudah suai (AMSDR), yang menggabungkan dua jenis pensampelan semula teragih tunggal yang berbeza, dan kemudiannya menghasilkan pengoptimuman memori pensampelan semula. Kesemuanya bertindak berdasarkan pengukuran hingar zarah, saiz sampel zarah dan memori. Hasil simulasi menunjukkan AMPSIR mencapai prestasi yang baik dengan mengurangkan ralat pengiraan nilai di pelbagai situasi pertidihan berat sampel tinggi sebanyak 7.26%, mengurangkan ralat penentuan nilai di pelbagai peringkat pertindihan berat sampel sebanyak 24.78% dan mengoptimumkan memori pensampelan semula sebanyak 28.73% berbanding sebelumnya. Hasil penemuan telah menunjukkan AMPSIR mempunyai kebolehan melakukan pelbagai kerja dan mampu memilih kesesuaiannya. Kesimpulanya, SIR yang dicadangkan membuktikannya ianya berharga bagi pelbagai persekitaran pertindihan berat sampel dan penggunaan memori. Selain itu, ianya mempunyai kebolehan menyesuaikan diri berdasarkan penggunaan memori yang digunakan oleh pengguna akhir, supaya dapat menentukan pensampelan semula yang sesuai berdasarkan penggunaan memori.

TABLE OF CONTENTS

	TITLE	PAGE
	DECLARATION	**ii**
	DEDICATION	**iii**
	ACKNOWLEDGEMENT	**iv**
	ABSTRACT	**v**
	ABSTRAK	**vi**
	TABLE OF CONTENTS	**vii**
	LIST OF TABLES	**xii**
	LIST OF FIGURES	**xiii**
	LIST OF ABBREVIATIONS	**xvii**
	LIST OF SYMBOLS	**xviii**
CHAPTER 1	**INTRODUCTION**	**1**
1.1	Backgroung of Study	1
1.2	Problem Background	2
1.3	Problem Statements	7
1.4	Research Questions	8
1.5	Research Aim	9
1.6	Research Objectives	10
1.7	Research Contributions	10
1.8	Research Scope	11
1.9	Significant of Research	11
1.10	Structure of Thesis	12
CHAPTER 2	**LITERATURE REVIEW**	**13**
2.1	Introduction	13
2.2	Adaptive Control Concept	15
	2.2.1 Adaptive Control Principles and Design	16

2.2.2 Application of Adaptive Control in Particle Filtering 17
2.3 Fundamental of Particle Filters 18
2.3.1 Particle Filters Principles and Design 19
2.3.2 Application of Particle Filter in Mobile Navigation System 22
2.4 Brief Description of the Particle Filter Resampling Mechanism 24
2.5 Resampling Classification for Particle Filtering 27
2.6 Parallel Implementation Resampling 29
2.7 Sequential Implementation Resampling 32
2.7.1 Single Distribution Resampling 34
2.7.1.1 Resampling based on Normalized Particle Weight Cumulative Sum 34
2.7.1.2 Resampling based on Residual 36
2.7.2 Compound Resampling 38
2.7.2.1 Dynamic Threshold based Resampling 36
2.7.2.2 Fixed Threshold based Resampling 40
2.7.2.3 Resampling That Takes into Account Particle Values 42
2.7.3 Special Strategies Resampling 43
2.7.3.1 Modified based Resampling 43
2.7.3.2 Variable Size based Resampling 44
2.7.3.3 Roughening based Resampling 45
2.8 Critical Review 47
2.8.1 Critical Review of Special Strategies Resampling 47
2.8.2 Critical Review of Sequential Implementation Resampling 51
2.8.3 Critical Review of Single Distribution Resampling 56
2.9 Discussion 59
2.10 Summary 61

CHAPTER 3	**RESEARCH METHODOLOGY**	**62**
3.1	Introduction	62
3.2	Overall Research Methodology Framework	64
	3.2.1 Problem Analysis and Formulation	66
	3.2.2 Design and Development	68
	3.2.2.1 Adaptive Noise and Sample Size based Special Strategies Resampling Scheme	70
	3.2.2.2 Adaptive Noise and Sample Size based Sequential Implementation Resampling Scheme	71
	3.2.2.3 Adaptive Memory based Single Distribution Resampling Scheme	72
	3.2.3 Performance Evaluation and Justification	74
	3.2.3.1 Simulation	76
	3.2.3.2 Performance Metrics	77
	3.2.3.3 Level and Factors	79
3.3	Summary	79
CHAPTER 4	**ADAPTIVE NOISE AND SAMPLE SIZE SPECIAL STRATEGIES RESAMPLING SCHEME**	**80**
4.1	Introduction	80
4.2	Concept of Special Strategies Resampling	82
4.3	Proposed Adaptive Noise and Sample Size Special Strategies Resampling (ANSSSR)	82
	4.3.1 Particle and Noise Measurement Adaptation	86
	4.3.2 Resampling Decision Making	86
	4.3.3 Generalized Resampling	87
	4.3.4 KLD Variation Resampling	90
	4.3.5 Bootstrap	93
4.4	Simulation Results	96
	4.4.1 Particle Weight, State and Sample Size	96

	4.4.2	State Estimation Error	101
	4.4.3	Root Mean Square Error (RMSE)	103
4.5		Summary	107
CHAPTER 5		**ADAPTIVE NOISE AND SAMPLE SIZE SEQUENTIAL IMPLEMENTATION RESAMPLING SCHEME**	**108**
5.1		Introduction	108
5.2		Concept of Sequential Implementation Resampling	110
5.3		Proposed Adaptive Noise and Particle Sequential Implementation Resampling (ANSSIR)	111
	5.3.1	Particle and Noise Measurement Adaptation	115
	5.3.2	Integrated Resampling of Generalized KLD Variation and Bootstrap Resampling in ANSSSR Scheme	115
	5.3.3	Resampling Decision Resampling	116
	5.3.4	Systematic Resampling	116
	5.3.5	Reallocation Resampling	119
5.4		Simulation Results	122
	5.4.1	Particle Weight, State and Sample Size	122
	5.4.2	State Estimation Error	132
	5.4.3	Time of Resampling	135
	5.4.3	Root Mean Square Error (RMSE)	138
5.5		Summary	141
CHAPTER 6		**ADAPTIVE MEMORY SINGLE DISTRIBUTION RESAMPLING SCHEME**	**143**
6.1		Introduction	143
6.2		Concept of Single Distribution Resampling	146
6.3		Proposed Adaptive Memory Single Distribution Resampling (AMSDR)	146
	6.3.1	Particle Sample Size and Noise Measurement	149
	6.3.2	Main Resampling Decision Making	149
	6.3.3	Bypass Resampling Decision Making	150
	6.3.4	Systematic Resampling	150

6.3.5 Rounding Copy Resampling 153
6.4 Simulation Results 155
6.4.1 Memory Allocation during Resampling Process by Using 2000 Particle 155
6.4.2 State Estimation Error During Resampling Process 162
6.4.3 Time Recorded During Resampling Process 164
6.4.4 Memory Efficiency and Memory Consumption 166
6.5 Implementation of Proposed Resampling in Case Study 170
6.6 Summary 173

CHAPTER 7 CONCLUSION AND RECOMENDATIONS 175
7.1 Introduction 175
7.2 Overall Result of Proposed Method 175
7.3 Research Achievement 176
6.4.2 New Special Strategies Resampling 177
6.4.2 New Single Implementation Resampling 177
6.4.2 New Single Distribution Resampling 178
7.4 Recommendations 179

REFERENCES 181

LIST OF PUBLICATIONS 194

LIST OF TABLES

TABLE NO.	TITLE	PAGE
Table 2.1	Classsification of Sequential Implementation Resampling	33
Table 2.2	Comparison and Summary of Special Strategies Resampling	50
Table 2.3	Comparison and Summary of Sequential Implementation Resampling	53
Table 2.4	Comparison and Summary of Single Distribution Resampling	58
Table 3.1	Development and Performance of the Phase of Research	63
Table 3.2	Experimental Framework	74
Table 3.3	Parameter Value Used in Simulation for Proposed Research	75
Table 4.1	Particle Weight, State and Sample Size of the Special Strategies Resampling based on Appendix A	100
Table 4.2	RMSE Value of Special Strategies Resampling in Different Situations in High Sample Impoversihment	105
Table 5.1	Particle Weight, State and Sample Size of the Sequential Implementation Resampling based on Online Technical Report	131
Table 5.2	Summary of Particle Weight, State and Sample Size from Table 5.1	132
Table 5.3	Run Time Values of Sequential Implementation Resampling in Different Stages of Sample Impoverishment	137
Table 5.4	RMSE Value of Sequential Implementation Resampling in Different Stages of Sample Impoverishment	141
Table 6.1	Analysis of Memory Optimization during Dummy Program below and above 500MB	170
Table 2.3	Comparison and Summary of Sequential Implementation Resampling	53

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
Figure 2.1	Adaptive Control Structure versus Conventional Control Design	16
Figure 2.2	Design and development phases of the proposed scheme (Muhamad, 2018) Bayesian Methods Knowledge Categorization Fundamental System Architecture of Inertial Positioning Systems Across All Environments. The Systems Operated by the Data Sensed by Inertial Sensor Will is Processed by CPU, Before Lookup Location Data Inside Database. Finally, the Final Positioning Information Will Projected it into Map.	20
Figure 2.4	Fundamental of Standard Particle Filter Flow Chart	21
Figure 2.5	Typical fully developed patterns on Shewhart control chart (Cheng, 1989) Comparison of Road Map in Mobile Navigation System which is Using Particle Filter (Red) and Not Using Particle Filter (Blue)	23
Figure 2.6	Illustration of Sample Degeneracy and Impoverishment with Various Situation and Stage	25
Figure 2.7	Illustration of Memory Efficiency and Memory Consumption in Application's Memory	26
Figure 2.8	Taxonomy of Resampling Algorithm for Particle Filters	28
Figure 3.1	Reserach Framework	65
Figure 3.2	Flow Chart of Sequential Important Sampling (SIS)	67
Figure 3.3	Research Methodology Flowchart for Adaptive Memory and Particle based Sequential Implementation Resampling (AMPSIR)	69
Figure 4.1	Illustration of Situation in High Sample Impoverishment	81
Figure 4.2	Taxonomy of Resampling Algorithm for Particle Filters Proposed Special Strategies Resampling Known as Adaptive Noise and Sample Size Special Strategies Resampling (ANSSSR Scheme)	84
Figure 4.3	Flow Chart of Generalized Resampling for ANSSSR scheme	89
Figure 4.4	Flow Chart of KLD Variation Resampling for ANSSSR scheme	92

Figure 4.5	Flow Chart of Bootstrap Variation Resampling for ANSSSR scheme	95
Figure 4.6	Error During Simulation for N=10	102
Figure 4.7	Error During Simulation for N=50	102
Figure 4.8	Error During Simulation for N=80	102
Figure 4.9	Error During Simulation for N=500 with Noise Measurement=0.1	102
Figure 4.10	Error During Simulation for N=500 with Noise Measurement=0.05	103
Figure 4.11	RMSE of Special Strategies Resampling in Different Situation During High Sample Impoverishment	106
Figure 5.1	Illustration of Stage of Sample Impoverishment	109
Figure 5.2	Error During Simulation for N=80 Proposed Sequential Implementation Resampling Scheme Known as Adaptive Noise and Sample Size Sequential Implementation Resampling (ANSSIR)	112
Figure 5.3	Flow Chart of Systematic Resampling for ANSSIR	118
Figure 5.4	Flow Chart of Reallocation Resampling for ANSSIR	121
Figure 5.5	Error During Simulation for N=10	133
Figure 5.6	Error During Simulation for N=50	133
Figure 5.7	Error During Simulation for N=80	133
Figure 5.8	Error During Simulation for N=100	133
Figure 5.9	Error During Simulation for N=200	134
Figure 5.10	Error During Simulation for N=300	134
Figure 5.11	Error During Simulation for N=400	134
Figure 5.12	Error During Simulation for N=500	134
Figure 5.13	Error During Simulation for N=500 with noise measurement=0.1	135
Figure 5.14	Error During Simulation for N=500 with noise measurement=0.05	135
Figure 5.15	Time of Resampling of Sequential Implementation Resampling in Different Stages of Sample Impoverishment	136

Figure 5.16	Error During Simulation for N=80 RMSE of Sequential Implementation Resampling in Different Stages of Sample Impoverishment	140
Figure 6.1	Illustration of Unoptimized Memory in Application's Memory	145
Figure 6.2	Proposed Single Distribution Resampling Known as Adaptive Memory Single Distribution Resampling (AMSDR)	148
Figure 6.3	Flow Chart of Systematic Resampling for AMSDR	152
Figure 6.4	Flow Chart of Rounding Copy Resampling for AMSDR	154
Figure 6.5	Memory Used During Resampling Process by Using 2000 Particle	157
Figure 6.6(a)	Memory Used During Systematic Resampling Process by Using 2000 Particle	158
Figure 6.6(b)	Memory Used During Multinomial Resampling Process by Using 2000 Particle	158
Figure 6.6(c)	Memory Used During Stratified Resampling Process by Using 2000 Particle	159
Figure 6.6(d)	Memory Used During Residual Resampling Process by Using 2000 Particle	159
Figure 6.6(e)	Memory Used During Residual Systematic Resampling Process by Using 2000 Particle	160
Figure 6.6(f)	Memory Used During Branching Resampling Process by Using 2000 Particle	160
Figure 6.6(g)	Memory Used During Rounding Copy Resampling Process by Using 2000 Particle	161
Figure 6.6(h)	Memory Used During Multinomial Single Distribution Resampling Process by Using 2000 Particle	161
Figure 6.7	State Estimation Error During Resampling Process by Using 2000 Particle	163
Figure 6.8	Time Recorded During the Resampling Process Using a 2000 Particle Sample Size	165
Figure 6.9	Memory Optimization of Single Distribution Resampling during Dummy Program below and above 500MB	169
Figure 6.10	Proposed Sequential Implementation Resampling Known as Adaptive Memory and Particle based Sequential Implementation Resampling Scheme (AMPSIR)	171

Figure 6.11 Predicted Value of X Position 172

Figure 6.12 Predicted Value of Y Position 172

LIST OF ABBREVIATIONS

SIR	-	Sequential Implementation Resampling
SSR	-	Special Strategies Resampling
SDR	-	Single Distribution Resampling
RNPWCS	-	Resampling based Normalized Particle Weights Cumulative Sum
RBR	-	Resampling based on Residual
CR	-	Compound Resampling
RMSE	-	Root Mean Square Error
ANSSSR	-	Adaptive Noise and Sample Size Special Strategies Resampling
AMPSIR	-	Adaptive Memory and Particle Sequential Implementation Resampling
ANSSIR	-	Adaptive Noise and Sample Size Sequential Implementation Resampling
AMSDR	-	Adaptive Memory Single Distribution Resampling
SIS		Sequential Important Sampling
FPGA		Field Programmable Gate Array
CPU		Central Processing Units
KLD		Kullback Leibler Distance
VLSI		Very Large-Scale Integrations
GP-GPU		General-Purpose computing on Graphics Processing Units
Open CL		Open Computing Language
CUDA		Compute Unified Device Architecture
RPA		Resampling with a Proportional Allocation
RNA		Resampling with Non-Proportional Allocation
PE		Processing Element
DART		Distributed Particle Filter Algorithm with A Resampling Tree
RSR		Residual Systematic Resampling
RC		Rejection Control
PR		Partial Resampling
RLR		Remarkable Local Resampling
ADR		Adaptive Deterministic Resampling
MCMC		Markov Chain Monte Carlo
WBASN		Wireless Body Area Sensor Network

LIST OF SYMBOLS

m	-	Current Particle Identification (before resampling)
M	-	Final Particle Identification (before resampling)
n	-	Current Particle Identification (after resampling) Final Particle Identification (after resampling) or Particle Sample Size
N	-	Final Particle Identification (after resampling) or Particle Sample Size
u	-	Random Number
x	-	Particle Distribution
t	-	Time (in step)
m	-	Current Particle Identification
Q		Cumulative Sum of Normalized Weight
U		Uniform Distribution
j		Resampling Cycle Identification
w		Particle Weight
R		Replication Number
c		Resampling Threshold
P		Probabilities
σ		Sigma
K		Positive Tuning Constant
D		Difference between Maximal and Minimal State Component Values
d_x		State Dimension

CHAPTER 1

INTRODUCTION

1.1 Background of Study

The rapid development of ubiquitous computing technology has made it possible to connect various ‘smart’ objects via the internet as a way of providing more data interoperability for its application purposes. However, the retrieved data must be analysed, manipulated and interpreted by a particle filter prior to being used for computer processing purposes as shown by the examples of target tracking (Ahmadi and Salari, 2017; Wang and Nguang, 2016), pollution monitoring (Metia et al., 2018, 2016), communications (Yao et al., 2015; Zhou et al., 2018), audio engineering (Ding et al., 2017; Muñoz-Romero et al., 2018), finance (Chauhan and Huseynov, 2018; Finlay et al., 2018), econometrics (González-Fernández and González-Velasco, 2018; Wilcox and Hamano, 2017) and other fields (Cappe et al., 2007; Doucet et al., 2001, 2000)). It is, however, often the case that by the time these data are observed or obtained, it was contaminated by the presence of noise, which makes it difficult to analyse the true data and retrieve relevant information. This raises important questions about the inferences and conclusions that can be drawn from the data. The practice of stochastic filtering attempts to understand and answer these questions. One of these methods is known as particle filters.

The general aim of the particle filter is to smoothen or approximate the data or particles for it to be easily read by the end user. In a ubiquitous computing environment however, the particle filter is required to process the data or particles from different types (in term of specification) of sensor or real time platforms, where the retrieved data or particles may be corrupted and consequently, affect the particle’s value (for example the particle weight or particle state) or its sample size. This will in turn lead to an unbalanced particle distribution or in other words, sample degeneracy and impoverishment.

However, the solution using resampling scheme in particle filter just considering the environment in a specific case, which is cannot generates reliable state estimation. Apart from that, the usage of resampling in particle filter also consuming their own memory and time, which is it reflect to the overall consumed system memory and time.

The main problem is how to optimise the accuracy of state estimation, and the same time also resampling's memory in various sample impoverishment situations. To resolve this, a resampling scheme can be adopted to counteract different forms of sample degeneracy and impoverishment. However, the solution of using a resampling scheme for a particle filter considers only the environment of a specific case, and so cannot generate reliable state estimation. In addition, the use of resampling in a particle filter also consumes memory and time, which is reflected in the overall memory and time consumed by the system.

During the implementation of the particle filter, it is desirable to ensure its compatibility within the system. The development of adaptive resampling scheme in a particle filter is argued to counteract different forms of sample degeneracy and impoverishment relating to unique features and conditions. The outcome of this study is expected to contribute to the modernisation of the particle filter by providing a significant number of new resampling for particle filtering.

1.2 Problem Background

The particle filter is a one (1) of crucial component in a software application, particularly in the mobile navigation systems. This is because it is ability to performing state estimation tasks using signal filtering. In the ubiquitous computing environment, it is more practical for a particle filter to work or cooperate simultaneously with different kinds (in term of specification) of sensor and real time platform system requirements. During the particle filtering process, the particle distribution will be unbalanced; this is known as sample degeneracy and impoverishment. In cases of sample degeneracy, the majority of the particles in a distribution was dominated by low weighted particles (where the weight is almost zero). By contrast, sample impoverishment can be described as the situation where most of the particles in a distribution was dominated by replicated high weighted particles. The resampling, such as sequential implementation resampling (SIR), can address this issue and balance the particle distribution. A situation of ideal or balanced particle weight distribution situation can lead to better state estimation. However, SIR considers the cause and environment at a specific time only, which cannot generate reliable state estimation during resampling in various computing configurations, and especially in the memory usage of various applications. In addition, the use of resampling in a particle filter also consumes memory and time, which is reflected in the overall memory used by the system. In a normal situation, a user application will require various memory usages. A developed and completed resampling scheme also experiences this issue, as its implementation relies on the memory of various applications.

Although SIR is generally able to balance particle weight to avoid the sample degeneracy effect, it is not able to completely overcome sample impoverishment. This is because SIR must consider various situations and stages of sample impoverishment during the implementation process. Two (2) important factors must be considered in regard to sample impoverishment; situation, and stages. The situation refers to the cause or factor affecting sample impoverishment; stages refers to the levels of sample impoverishment, which consist of high sample impoverishment, low sample impoverishment, and very low sample impoverishment. The following will elaborate further on the three (3) different issues relating to existing SIR schemes

The first (1^{st}) issue of this research is inaccurate state estimation under situations in high sample impoverishment. In particle filtering, providing reliable and accurate state estimation during situations in high sample impoverishment is essential. This is due to the different factors that can affect the particle distribution. This situation is initially caused by three (3) possible factors; low particle sample size, moderate particle sample size, and low measurement noise (Pak et al., 2016, 2017; Fox et al., 2001). Low and moderate particle sample size are caused by variations in real time system architecture, while low measurement noise is caused by high accuracy sensor integration. As mentioned, sample impoverishment will lead to the particle weight distribution being, unbalanced, ultimately causing inaccurate state estimation. Specific SIR schemes have been developed to rectify this problem (Li et al., 2015a). The use of SSR can address this issue. SSR is a resampling scheme that can help to balance the particle distribution during situations in high sample impoverishment. There are three (3) main type of SSR scheme; modified based resampling, variable size based resampling, and roughening based resampling. Each type is suitable for a specific situation in high sample impoverishment.

As mentioned before, there are different contributing factors to or causes of sample impoverishment, primarily; low particle sample size, moderate particle sample size, and low measurement noise. In a situation of low particle sample size, the solution is variable size based resampling. Meanwhile, for low measurement of noise, the solution is roughening based resampling. Finally, for moderate particle sample size, modified based resampling is the most suitable solution. However, the usage of resampling in different sample impoverishment situations can increase state estimation error. For example, if the modified based resampling is applied to other situations (such as low particle sample size or low noise measurement), the particle distribution will not change much. On the other hand, if variable size based resampling is used in other situations (such as moderate particle sample size or low measurement noise), it can adjust to a small number of differences in particle distribution. Finally, the use of roughening based resampling in other situations (such as low measurement noise and particle sample size) will have a minimal effect on particle distribution and also be time consuming. Each categorisation of SSR will be explained in more detail in the Section 2.7 (critical review).

Based on the above discussion, it can be concluded that special strategies resampling can be implemented in the specific situations in high sample impoverishment only (Li et al., 2015a). Accordingly, in order to carry out more comprehensive resampling in other, different situations in high sample impoverishment, the factor of different situation is an important consideration in high sample impoverishment, especially state estimation error. In addition, an adaptive particle and noise measurement input is required in order to help balance state estimation accuracy in different situations in high sample impoverishment.

The second (2^{nd}) issue of this research is inaccurate state estimation under stages of sample impoverishment. The use of SSR as a single solution by means to increase the accuracy of state estimation during high sample impoverishment can be considered inefficient, since it is overly time consuming. Other SIR resampling schemes can better address this issue. There are three (3) main type of SIR scheme; single distribution resampling (SDR), compound resampling (CR), and special strategies resampling (SSR). Each of these schemes is able to counteract certain stages of sample impoverishment (Pak et al., 2016, 2017; Li et al., 2015a). Nevertheless, there still a need to develop schemes which is can achieve reliable state estimation accuracy across different stages of sample impoverishment.

As mentioned before, there are different stages of sample impoverishment; high sample impoverishment, low sample impoverishment, and very low sample impoverishment. The solution to high sample impoverishment is special strategies resampling (SSR); low sample impoverishment, the solution is compound resampling (CR); and, finally, for very low sample impoverishment, single distribution resampling (SDR) is the most suitable solution. However, the use of resampling at different sample impoverishment stages can make it difficult to achieve accurate state estimation. For example, if SSR is used at other sample impoverishment stages, state estimation error might be increased. Similarly, if CR or SDR is used at inappropriate sample impoverishment stages, state estimation error can be increased. The Section 2.7 (critical review) will be provided more details for each categorisation of SIR.

Based on these, it can be concluded that different forms of sequential implementation resampling can be implemented at specific stages of sample impoverishment only (Li et al., 2015a). Accordingly, in order to carry out more comprehensive resampling during the different stages of sample impoverishment, the state estimation error should be assessed. In addition, an adaptive particle and noise measurement input is required in order to help balance state estimation accuracy at different stages of sample impoverishment.

Finally, the third (3^{rd}) issue of this research is non optimized resampling's memory during implementation in various application's memory usage. To ensure optimized resampling's memory usage in various applications, existing SIR schemes, also known as single distribution resampling (SDR), produce unbalanced memory usage during resampling in various application's memory usage. This is due to different memory required during resampling in different applications (Ikuzawa et al., 2016; Li et al., 2015a; Orsila et al., 2007; Hightower and Borriello, 2004; Grisetti et al., 2007; Bolić et al., 2004; Hong et al., 2010).

Two (2) main types of SDR scheme have been proposed by other researchers; resampling based on normalised particle weights cumulative sum (RNPWCS), and resampling based on residual (RBR). The SDR scheme has a specific resampling's frequency rate, which requires a specific memory rate that is suitable for certain applications. The unbalanced memory usage by resampling occurs when an application's memory usage changes from high to low, or vice versa, whereas the memory rate of resampling is fixed. A difference in memory usage between the application and resampling will lead to unbalanced resampling memory, where it will either consume a high amount of memory, or have low memory efficiency

As mentioned earlier, there are two (2) different levels of memory usage in an application, high application's memory usage and low application's memory usage. According to (Qiu et al., 2015), most of the end user application will needs at least one (1) GB (maximum) memory, for their memory allocation purpose. This is will make any required memory of 500MB and below; considered as low memory usage and any required memory above than 500MB; considered as high memory usage. This required application's memory will be change from time to time based on the allocated memory of a application. The implementation of the single distribution resampling scheme in the application (especially in mobile navigation system), may result unoptimized resampling's memory allocation (Ikuzawa et al., 2016; Li et al., 2015a; Orsila et al., 2007; Hightower and Borriello, 2004; Grisetti et al., 2007; Bolić et al., 2004; Hong et al., 2010). For high application's memory usage, the solution is resampling based on normalised particle weights cumulative sum (RNPWCS). On the other hand, the solution to low application's memory usage is resampling based on residual (RBR). The use of such resampling for different memory usage of applications can lead to unoptimised resampling. For example, if the resampling is based on normalised particle weights cumulative sum (RNPWCS) and applied to the condition of low application's memory usage, this will lead to inefficient memory usage. On the other hand, the use of resampling based on residual (RBR) under the condition of high application's memory usage will lead to high memory consumption. Each categorisation of SDR will be explained in more detail in the Section 2.7 (critical review).

Based on these, it can be concluded that single distribution resampling can be implemented in applications with specific application's memory usage. Accordingly, in order to carry out more comprehensive resampling in applications with different application's memory usage, the factor of memory allocation should be considered in relation to different resampling schemes. In addition, an adaptive memory input is required in order to balance the memory allocation for resampling in applications with different application's memory usage.

1.3 Problem Statement

This research addresses a number of critical problems faced by the existing sequential implementation resampling (SIR) of particle filtering under different sample impoverishment environments. The implementation of resampling in the particle filter will indirectly exposes the particle distribution to the various other computing aspects, such as different sensor and real time platform. It was also discovered to have resulted in an unbalanced particle distribution and consequently, inaccuracies in the state estimations. Apart from the above, the existing resampling method was also revealedto suffer from an unbalanced memory usage when being executedin the end user application.

The best SIR resampling scheme, which is able to counteract sample impoverishment, is special strategies resampling (SSR). However, current SSR schemes are only able to accurately carry out resampling task during specific situations in high sample impoverishment. Additionally, there are different situations in high sample impoverishment situations in which SSR cannot establish state estimation accurately.

The usage of SSR in SIR is important to establish better accuracy of state estimation generally. However, current SIR schemes are only able to accurately apply resampling at specific levels of sample impoverishment. Additionally, there are different stages of sample impoverishment situations in which SIR cannot establish state estimation accurately.

As a resampling scheme, the particle selection process may require a certain amount of memory. The key resampling scheme that can control the resampling cycle is single distribution resampling (SDR). However, current SDR schemes are only applicable to balance the resampling's memory in certain application's memory usage. Additionally, there are different level of application's memory in which SIR cannot optimize the resampling's memory properly.

1.4 Research Questions

Based on the discussion provided in Section 1.2, the research questions can be formulated as follows;

i. How to reduce state estimation error over situations during high sample impoverishment?
   a. How to minimize RMSE in various situation in high sample impoverishment?
   b. How to choose suitable resampling for various situation in high sample impoverishment?

ii. How to reduce state estimation error over sample impoverishment stages?
   a. How to minimize RMSE in various stages of sample impoverishment?

b. How to choose suitable resampling for various stages of sample impoverishment?

iii. How to optimize memory usage between resampling's memory in different application's memory usage?

a. How to minimize resampling's memory consumption as well preserve resampling's memory efficiency in different application's memory usage?

b. How to choose suitable resampling for different application's memory usage?

1.5 Research Aim

The aim of this research is to design a new resampling scheme of adaptive memory and particle sequential implementation resampling (AMPSIR) with; adaptive noise and sample size special strategies resampling (ANSSSR), adaptive noise and sample size sequential implementation resampling (ANSSIR), and adaptive memory single distribution resampling (AMSDR), which employ current resampling scheme in order to, improve state estimation accuracy in different situation in high sample impoverishment, stages of sample impoverishment, and also balance resampling's memory usage in particle filtering.

1.6 Research Objectives

For this purpose, the different sub objectives that need to be addressed are in the following manner;

i. To design an adaptive noise and sample size based special strategies resampling (ANSSSR) scheme, to reduce RMSE while preserving it in different situation in high sample impoverishment.

ii. To design an adaptive noise and sample size based sequential implementation resampling (ANSSIR) scheme, to reduce RMSE while preserving it in different stages of sample impoverishment.

iii. To design and implement adaptive memory based single distribution resampling (AMSDR) scheme in sequential implementation resampling, to reduce memory consumption and preserving memory efficiency.

1.7 Research Contributions

The main contribution of this research is the use of adaptive memory and the particle sequential implementation resampling (AMPSIR) scheme in particle filtering, which had provided a reliable state estimation in applications with different memory usage. For this reason, the AMPSIR is thus seen as a combined outcome from the following three (3) contributions.

The first (1^{st}) contribution is, the adaptive noise and sample size based special strategies resampling (ANSSSR) scheme, which is used to reduce and preserve RMSE in different situations of high sample impoverishment. Current SSR schemes are only able to accurately carry out resampling task during specific situations in high sample impoverishment. Additionally, there are different situations in high sample impoverishment situations in which SSR cannot establish state estimation accurately. Therefore, there is a need for a trade off between RMSE in different situations of high sample impoverishment. This scheme will minimise RMSE in different high sample impoverishment situations.

Second (2^{nd}) contribution is the adaptive noise and sample size based sequential implementation resampling (ANSSIR) scheme, which is used to reduce and preserve RMSE at different levels of sample impoverishment. Current sequential implementation resampling (SIR) schemes are only able to accurately apply resampling at specific levels of sample impoverishment. Additionally, there are different stages of sample impoverishment situations in which SIR cannot establish state estimation accurately. Therefore, there is a need for a trade off between RMSE at different levels of sample impoverishment. This scheme will minimise RMSE at different levels of sample impoverishment.

The third (3^{rd}) contribution is the adaptive memory based single distribution resampling (AMSDR) scheme for sequential implementation resampling, which is used to reduce memory consumption and preserve memory efficiency. Current SDR schemes are only applicable to balance the resampling's memory in certain application's memory usage. Additionally, there are different level of application's memory in which SIR cannot optimize the resampling's memory properly. Therefore, there is a trade off between resampling when uniform across applications with different memory usages. This scheme will reduce memory consumption and preserve memory efficiency.

1.8 Research Scope

This research presents an adaptive memory and particle sequential implementation resampling (AMPSIR) scheme for particle filtering. The scope of this research covers the following;

i. This study addresses particle filtering in mobile navigation system scenario.
ii. The value of particle sample size and noise measurement is in MATLAB parameter value. The value is based on previous experiment that has been done other researcher.
iii. This study excludes sensor sensitivity or method to obtain noise measurements from real sensor or simulated sensor.

1.9 Significant of Research

This research contributes to the field of particle filtering by focusing on the development of adaptive memory and particle sequential implementation resampling (AMPSIR). This is to provide reliable state estimation across the various application memory. Meanwhile, the proposed resampling solution is suitable for various levels of sample impoverishment, a situation commonly faced by particle filters in ubiquitous computing environments where the end user application (for example

Waze, Papago, Google Maps, or other mobile navigation software) will be integrated with different sensor configurations (internal or external sensor) and system architecture (whether using real time or non real time positioning). As addition, it is provides balanced state estimation that is suitable for implementation in small devices or other devices on which a heavy application is installed. Furthermore, the proposed resampling scheme has the ability to adapt the end user application's memory usage, where the proposed resampling solution is can determine the most suitable resampling scheme based on the application's memory usage.

1.10 Structure of Thesis

This thesis is divided into seven (7) chapters. Chapter 1 has introduced the study, by highlighting the background to the study, stating the objectives, problem statements, and contributions of the study. Chapter 2 will present a literature review of particle filter resampling and an overview of existing sequential implementation resampling schemes. Chapter 3 will discuss the research methodology used to achieve the research objectives. Chapter 4 will formally introduce the adaptive noise and sample size based special strategies resampling scheme (ANSSSR), while Chapter 5 will introduce the adaptive noise and sample size based sequential implementation resampling (ANSSIR), and Chapter 6 will present the adaptive memory based single distribution sequential implementation resampling (AMSDR). Finally, Chapter 7 will conclude the thesis and make recommendations for possible future works.

REFERENCES

Ahmadi, K., Salari, E., (2017) 'Social-spider optimised particle filtering for tracking of targets with discontinuous measurement data', *IET Computer Vision*, 11 (3), 246–254.

Andrieu, C., Doucet, A., Singh, S.S., Tadic, V.B., (2004) 'Particle methods for change detection, system identification', *Proceedings of IEEE,* 92(3), 423–438.

Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T., (2002) 'A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking', *IEEE Transactions on Signal Processing*, 50(2), 174–188.

Åström, K.J., Wittenmark, B., (2013) .*Adaptive Control: Second Edition*. New York: Dover Publications.

Balasingam, B., Bolić, M., Djurić, P.M., Míguez, J., (2011) Efficient Distributed Resampling For Particle Filters. *International Conference on Acoustics, Speech and Signal Processing*. 12 July. Prague, Czech Republic: IEEE. 3772–3775.

Bate, M.R., (2005) 'The dependence of the initial mass function on metallicity and the opacity limit for fragmentation', *Monthly Notices of the Royal Astronomical Society*, 363(2), 363–378.

Beadle, E.R., Djuric, P.M., (1997) 'A fast-weighted Bayesian bootstrap filter for nonlinear model state estimation', *IEEE Transactions on Aerospace. Electronic Systems,* 33(1), 338–343.

Bejuri, W.M.Y.W., (2019) *Appendix of Objective 1 and 2 Result for Ph.D. Thesis (Title: Adaptive Memory and Particle based Sequential Implementation Resampling for Particle Filtering)*. Universiti Teknologi Malaysia Institutional Repository*,* Universiti Teknologi Malaysia, Skudai.

Bolić, M., Djurić, P.M., Hong, S., (2004) 'Resampling algorithms for particle filters: a computational complexity perspective', *EURASIP Journal Advance. Signal Processing*, 2004(15), 2267-2277.

Bolić, M., Djuric, P.M., Sangjin Hong, (2005) 'Resampling algorithms and architectures for distributed particle filters', *IEEE Transactions on Signal Processing*. 53(7), 2442–2450.

Bolić, M., Hong, S., Djuric, P.M., (2002) Performance and Complexity Analysis of Adaptive Particle Filtering for Tracking Applications. *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*. 3-6 November. Pacific Grove, USA: IEEE. 853–857.

Budhiraja, A., Chen, L., Lee, C., (2007) 'A survey of numerical methods for nonlinear filtering problems'. *Physica D: Nonlinear Phenomena*, 230(1-2), 27–36.

Caesarendra, W., Niu, G., Yang, B.-S., (2010) 'Machine condition prognosis based on sequential monte carlo method'. *Expert Systems with Application* 37(3), 2412–2420.

Cappe, O., Godsill, S.J., Moulines, E., (2007) 'An overview of existing methods and recent advances in sequential monte carlo'. *Proceedings of IEEE*, 95(5), 899–924.

Chau, T.C.P., Luk, W., Cheung, P.Y.K., Eele, A., Maciejowski, J., (2012) Adaptive Sequential Monte Carlo Approach For Real-Time Application's. *22nd International Conference on Field Programmable Logic and Application*. 29-31 August. Oslo, Norway: IEEE. 527–530.

Chau, T.C.P., Niu, X., Eele, A., Luk, W., Cheung, P.Y.K., Maciejowski, J., (2013) Heterogeneous Reconfigurable System for Adaptive Particle Filters in Real-Time Applications. *International Symposium on Applied Reconfigurable Computing*. 2-4 May, Santorini, Greece: Springer, 1–12.

Chau, T.C.P., Niu, X., Eele, A., Maciejowski, J., Cheung, P.Y.K., Luk, W., (2014) 'Mapping adaptive particle filters to heterogeneous reconfigurable systems', *ACM Transactions Reconfigurable Technolgy System*, 7(4), 1–17.

Chauhan, G.S., Huseynov, F., (2018) 'Corporate financing and target behavior: new tests and evidence', *Journal of Corporate Finance*, 48, 840–856.

Cliffordy, J.C.P., Fearnhead, P., (1999) 'An improved particle filter for non-linear problems', *IEE Proceedings-Radar, Sonar and Navigation*, 146(1), 2-7.

Crisan, D., Lyons, T., (1999) 'A particle approximation of the solution of the Kushner–Stratonovitch equation', *Probability Theory and Related Fields*, 115(4), 549-578.

Daum, F., 2005. 'Nonlinear filters: beyond the Kalman filter', *IEEE Aerospace and Electronic Systems Magazine*, 20(8), 57-69.

Ding, Y., Huang, J., Pelachaud, C., (2017) 'Audio-driven laughter behavior controller', *IEEE Transactions on Affecting Computing*, 8(4), 546–558.

Djuric, P.M., Kotecha, J.H., Zhang, J., Huang, Y., Ghirmai, T., Bugallo, M.F., Miguez, J., (2003) 'Particle filtering'. *IEEE Signal Processing Magazine,* 20(5), 19–38.

Douc, R., Cappe, O.,(2005) Comparison Of Resampling Schemes For Particle Filtering. *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*. 15-17 September. Zagreb, Croatia: IEEE, 64–69.

Doucet, A., Freitas, N. de, Gordon, N., (2001) '*An Introduction to Sequential Monte Carlo Methods*', in Doucet A., De-Freitas N., Gordon N. (eds.) *Sequential Monte Carlo Methods in Practice, Statistics for Engineering and Information Science*. New York: Springer, pp. 3–14.

Doucet, A., Godsill, S., Andrieu, C., (2000) 'Sequential monte carlo sampling methods for bayesian filtering'. *Statistics and Computing*, 10(3), 197–208.

Doucet, A., Johansen, A.M., 2011a. A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later. *Handbook of Nonlinear Filtering*,12(3), 1-39.

Dülger, Ö., Oğuztüzün, H., Demirekler, M., (2018) 'Memory coalescing implementation of metropolis resampling on graphics processing unit'. *Journal of Signal Processing Systems*, 90(3), 433-447.

EL Bakkali, J., EL Bardouni, T., Safavi, S., Boukhal, H., Kaddour, M., Benaalilou, K., Chham, E., (2016) ERSN-OpenMC, a java-based GUI for openmc monte carlo code. *Journal of Radiation Research and Applied Sciences*, 9(3), 234-241.

Elvira, V., Martino, L., Luengo, D., Bugallo, M.F., (2017) 'Improving population monte carlo: alternative weighting and resampling schemes'. *Signal Processing*, 131, 77-91.

Fearnhead, P., Clifford, P., (2003). 'Online inference for hidden markov models via particle filters'. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(4), 887-899.

Fetzer, T., Ebner, F., Deinzer, F., Grzegorzek, M., (2017) Recovering from sample impoverishment in context of indoor localisation. *International Conference* *on Indoor Positioning and Indoor Navigation.* 18-21 September. Sapporo, Japan: IEEE. 1–8.

Finlay, W., Marshall, A., McColgan, P., (2018) 'Financing, fire sales, and the stockholder wealth effects of asset divestiture announcements', *Journal of Corporate Finance*, *50*, 323-348.

Fox, D., (2003) 'Adapting the sample size in particle filters through KLD-Sampling', *The International Journal of Robotics Research*, 22(12), 985-1003.

Fox, D., Thrun, S., Burgard, W., Dellaert, F., (2001) 'Particle Filters for Mobile Robot Localization', in: Sequential Monte Carlo Methods in Practice, Statistics for Engineering and Information Science. Springer, New York, NY, 401–428.

Fraccaro, M., Paquet, U., Winther, O., (2016) 'An adaptive resample-move algorithm for estimating normalizing constants'. *arXiv preprint arXiv:1604.01972,* 1-11.

Gandy, A., Lau, F.D.H., (2016) 'The chopthin algorithm for resampling. *IEEE Transactions on Signal Processing*, 64(16), 4273-4281.

Gilks, W. R., Berzuini, C., (2002). 'Following a moving target—monte carlo inference for dynamic bayesian models'. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(1), 127-146.

González-Fernández, M., González-Velasco, C., (2018).' Can google econometrics predict unemployment? evidence from Spain', *Economics Letters*, 17*0*, 42-45

Gordon, N.J., Salmond, D.J., Smith, A.F.M., (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation, *IEE Proceedings F- Radar and Signal Processing*. 140 (2), 107–113.

Grisetti, G., Tipaldi, G.D., Stachniss, C., Burgard, W., Nardi, D., (2007). 'Fast and accurate SLAM with rao–blackwellized particle filters', *Robotics and Autonomous Systems*, 55(1), 30-38.

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., Nordlund, P.J., (2002) 'Particle filters for positioning, navigation, and tracking', *IEEE Transactions on signal processing*, 50(2), 425-437.

Havangi, R., (2015) 'Unscented H-infinity filtering based simultaneous localization and mapping with evolutionary resampling'. *Journal of the Franklin Institute*, 352(11), 4801-4825.

Hendeby, G., Hol, J.D., Karlsson, R., Gustafsson, F., (2007) A graphics processing unit implementation of the particle filter. *15th European Signal Processing Conference*. 3-7 September. Poznan, Poland. 1639–1643.

Hendeby, G., Karlsson, R., Member, F.G. (2010) 'Particle filtering: the need for speed'. *EURASIP Journal on Advances in Signal Processing*, 2010 (22), 1-9.

Hightower, J., Borriello, G., (2004) Particle Filters for Location Estimation in Ubiquitous Computing: A Case Study. *UbiComp 2004: Ubiquitous Computing*. 7-10 September. Nottingham, UK: Springer, 88–106.

Hlinka, O., Hlawatsch, F., Djuric, P.M., (2013) 'Distributed particle filtering in agent networks: A survey, classification, and comparison'. *IEEE Signal Processing Magazine*, *30*(1), 61-81.

Hol, J.D., Schon, T.B., Gustafsson, F., (2006) On resampling algorithms for particle filters. *IEEE Nonlinear Statistical Signal Processing Workshop*. 13-15 September. Cambridge, UK: IEEE. 79–82.

Hong, S., Chin, S.-S., Djurić, P.M., Bolić, M., (2006) 'Design and implementation of flexible resampling mechanism for high-speed parallel particle filters'. *Journal of VLSI signal processing systems for signal, image and video technology*, *44*(1-2), 47-62.

Hong, S.-H., Shi, Z.-G., Chen, J.-M., Chen, K.-S., (2010) 'A low-power memory-efficient resampling architecture for particle filter'. *Circuits, Systems and Signal Processing*, *29*(1), 155-167

Hwang, K., Sung, W., (2013) 'Load balanced resampling for real-time particle filtering on graphics processing units'. *IEEE Transactions on Signal Processing*, *61*(2), 411-419.

Ikuzawa, T., Ino, F., Hagihara, K., (2016) 'Reducing memory usage by the lifting-based discrete wavelet transform with a unified buffer on a GPU', *Journal of Parallel and Distributed Computing*, 93, 44-55.

Islam, Z., Oh, C., Lee, C., (2010). 'Effect of resampling steepness on particle filtering performance in visual tracking', *International. Arab Journal. Information Technology*, 10(1), 102-109.

Jiang, Z., Zhou, W., Li, H., Mo, Y., Ni, W., Huang, Q., (2018) 'A new kind of accurate calibration method for robotic kinematic parameters based on the extended kalman and particle filter algorithm'. *IEEE Transactions on Industrial Electronics*, 65(4), 3337-3345.

Jianping, Z., Baoming, B., Xinmei, W., (2009) 'Increased-diversity systematic resampling in particle filtering for BLAST'. *Journal of Systems Engineering and Electronics*, 20(3), 493-498.

Jiao, W., (2016) *An Overview of Efficient Nonlinear Filtering - From Kalman Filter To Particle Filters To EIS*. PhD Thesis. University of Pittsburgh, United States.

Jouin, M., Gouriveau, R., Hissel, D., Péra, M.-C., Zerhouni, N., (2016) 'Particle filter-based prognostics: review, discussion and perspectives'. Mechanical. System Signal Processing, 72(73), 2–31.

Kantas, N., Doucet, A., Singh, S.S., Maciejowski, J.M., (2009) 'An overview of sequential monte carlo methods for parameter estimation in general state-space models'. 15th IFAC Symposium on System Identification, 42(10), 774–785.

Kitagawa, G., (1996) 'Monte carlo filter and smoother for non-gaussian nonlinear state space models'. Journal of Computational and Graphical Statistics, 5(1), 1–25.

Koller, D., Fratkina, R., (1998) Using Learning for Approximation in Stochastic Processes. *Proceedings of the International Conference on Machine Learning*. 24-27 July. San Francisco, USA: ACM. 287–295.

Kostanjčar, Z., Jeren, B., Cerovec, J., (2009) 'Particle filters in decision making problems under uncertainty'. *Automatika: Journal for Control, Measurement, Electronics, Computing and Communications*. 50(3-4), 245–251.

Kouritzin, M.A., (2017) 'Residual and stratified branching particle filters'. *Computational Statistics & Data Analysis*, 111, 145-165.

Landau, I.D., Lozano, R., M'Saad, M., Karimi, A., (2011) *Adaptive Control: Algorithms, Analysis and Applications*. Springer Science & Business Media: London.

Lee, Y., Majda, A.J., (2016) 'State Estimation And Prediction Using Clustered Particle Filters' *Proceedings of the National Academy of Sciences*, 113(51), 1 4609-14614.

Li, F., Bonnifait, P., Ibañez-Guzmán, J., (2018) 'Map-aided dead-reckoning with lane-level maps and integrity monitoring'. *IEEE Transactions on Intelligent Vehicles*, 3(1), 81-91.

Li, K., Wu, J., Zhang, Q., Su, L., Chen, P., (2017) 'New particle filter based on ga for equipment remaining useful life prediction'. *Sensors*, 17(4), 696-671.

Li, T.C., Corchado, J.M., Prieto, J., (2017) 'Convergence of distributed flooding and its application for distributed bayesian filtering'. *IEEE Transactions on Signal and Information Processing over Networks*, 3(3), 580-591.

Li, T.C., Sun, S., Corchado, J.M., Sattar, T.P., Si, S., (2016) 'Numerical fitting-based likelihood calculation to speed up the particle filter', *International Journal of Adaptive Control and Signal Processing*, 30(11), 1583-1602.

Li, T.C., Sun, S., Duan, J., (2010) Monte carlo localization for mobile robot using adaptive particle merging and splitting technique. *IEEE International Conference on Information and Automation*, 19 July. Harbin, China: IEEE. 1913–1918.

Li, T.C., Bolić, M., Djuric, P.M., (2015a) 'Resampling methods for particle filtering: classification, implementation, and strategies'. *IEEE Signal Processing Magazine*, 32(3), 70-86.

Li, T.C., Villarrubia, G., Sun, S., Corchado, J.M., Bajo, J., (2015b) 'Resampling methods for particle filtering: identical distribution, a new method, and comparable study', *Frontiers of Information Technology & Electronic Engineering*, 16(11), 969-984.

Li, T.C., Sattar, T.P., Tang, D., (2013a) A Fast Resampling Scheme For Particle Filters. *Constantinides International Workshop on Signal Processing*. 25 January. London, UK: IEEE. 1-4.

Li, T.C., Sun, S., Sattar, T.P., (2013b) 'Adapting sample size in particle filters through KLD-resampling', *Electronics Letters*, 49(12), 740-742.

Li, T.C., Sattar, T.P., Sun, S., (2012) 'Deterministic resampling: unbiased sampling to avoid sample impoverishment in particle filters'. *Signal Processing*, 92(7), 1637-1645.

Li, Y., Coates, M., (2017) 'Particle filtering with invertible particle flow', *IEEE Transactions on Signal Processing*, 65(15), 4102-4116.

Li, Y., Sui, S., Tong, S., (2017) 'Adaptive fuzzy control design for stochastic nonlinear switched systems with arbitrary switchings and unmodeled dynamics', *IEEE Transactions on Cybernetics*, 47(2), 403-414.

Liu, H., Liu, Z., Lu, F., (2017) 'A systematic comparison of particle filter and enkf in assimilating time averaged observations', *Journal of Geophysical Research: Atmospheres*, 122(24), 155-173.

Liu, J.S., Chen, R., Logvinenko, T., (2001) *A Theoretical Framework for Sequential Importance Sampling with Resampling*, in: Doucet A., De-Freitas N., Gordon N. (eds.) *Sequential Monte Carlo Methods in Practice*. Springer: New York, pp. 225–246.

Liu, J.S., Chen, R., (1998a) 'Sequential monte carlo methods for dynamic systems', *Journal of the American statistical association*, 93(443), 1032-1044.

Liu, J.S., Chen, R., Wong, W.H., (1998b) 'Rejection control and sequential importance sampling', *Journal of the American Statistical Association*, *93*(443), 1022-1031.

Liu, Z., Shi, Z., Zhao, M., Xu, W., (2007) Mobile Robots Global Localization Using Adaptive Dynamic Clustered Particle Filters. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 29 October-2 November. San Diego, USA: IEEE. 1059–1064.

Lopes, H.F., Tsay, R.S., (2010) 'Particle Filters And Bayesian Inference In Financial Econometrics', *Journal of Forecasting*, 30(1), 168-209.

Ly-Tu, N., Le-Tien, T., Mai, L., (2017) A New Resampling Parameter Algorithm for Kullback-Leibler Distance with Adjusted Variance and Gradient Data Based on Particle Filter. *International Conference on Industrial Networks and Intelligent Systems*. 8-10 January. Bali, Indonesia: Springer. 347–358.

Ly-Tu, N., Le-Tien, T., Vo-Thi-Luu, P., Mai, L., (2015) Particle Filter Through Kullback-Leibler Distance Resampling with Adjusted Variance and Gradient Data for Wireless Biomedical Sensor Networks. *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*. 8-10 January. Bali, Indonesia: ACM. 1–5.

Magnusson, J., Winstral, A., Stordal, A.S., Essery, R., Jonas, T., (2017) 'Improving physically based snow simulations by assimilating snow depths using the particle filter', *Water Resources Research*, 53(2), 1125-1143.

Mahmoudi, Z., Wendt, S.L., Boiroux, D., Hagdrup, M., Nørgaard, K., Poulsen, N.K., Madsen, H., Jørgensen, J.B., (2016) 'Comparison of Three Nonlinear Filters for Fault Detection in Continuous Glucose Monitors', *2016 38th IEEE*

Engineering in Medicine and Biology Society, 16-20 August. Orlando, USA: IEEE. 3507–3510.

Mamagkakis, S., Baloukas, C., Atienza, D., Catthoor, F., Soudris, D., Mendias, J.M., Thanailakis, A., (2005) Reducing Memory Fragmentation with Performance-Optimized Dynamic Memory Allocators in Network Applications. *International Conference on Wired/Wireless Internet Communications*. 18-20 June. Boston, USA: Springer. 354–364.

Mei, X., Ling, H., (2009) Robust Visual Tracking using 1 Minimization. *12th International Conference on Computer Vision Workshop*. 29 September-2 October. Kyoto, Japan: IEEE. 1436-1443.

Metia, S., Ha, Q.P., Duc, H.N., Azzi, M., (2018) 'Estimation of power plant emissions with unscented kalman filter', *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(8), 2763-2772.

Metia, S., Oduro, S.D., Duc, H.N., Ha, Q., (2016) 'Inverse air-pollutant emission and prediction using extended fractional kalman filtering', *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(5), 2051-2063.

Míguez, J., (2007) 'Analysis of parallelizable resampling algorithms for particle filtering. signal process', *Signal Processing*, 87(12), 3155-3174.

Míguez, J., Bugallo, M.F., Djurić, P.M., (2004) 'A new class of particle filters for random dynamic systems with unknown statistics', *EURASIP Journal on Applied Signal Processing*, 2004(303619), 2278-2294.

Mihaylova, L., Carmi, A.Y., Septier, F., Gning, A., Pang, S.K., Godsill, S., (2014) 'Overview of bayesian sequential monte carlo methods for group and extended object tracking'. *Digital Signal Processing*, 25, 1-16.

Muñoz-Romero, S., Arenas-García, J., Gómez-Verdejo, V., (2018) 'Nonnegative OPLS for supervised design of filter banks: application to image and audio feature extraction', *IEEE Transactions on Multimedia*, 20(7), 1751-1766.

Murray, L., (2011) GPU Acceleration of the Particle Filter: The Metropolis Resampler. *DMMD 2011 Symposium: Distributed Machine Learning and Sparse Representation with Massive Data Sets*. 1 November. Sydney, Australia: CSIRO. 1-5.

Nakagawa, G., Kawata, H., Oikawa, S., (2015) Out of memory prevention based on memory allocation rate. *Third International Symposium on Computing and Networking*. 8-11 December. Hokkaido, Japan: IEEE, 566–570.

Orguner, U., Gustafsson, F., (2008) 'Risk-Sensitive particle filters for mitigating sample impoverishment', *IEEE Transactions on Signal Processing*, 56(10), 5001-5012.

Orsila, H., Kangas, T., Salminen, E., Hämäläinen, T.D., Hännikäinen, M., (2007) 'Automated memory-aware application distribution for multi-processor system-on-chips', *Journal of Systems Architecture*, 53(11), 795-815.

Pak, J.M., Ahn, C.K., Lim, M.T., Shmaliy, Y.S., (2016) Combined particle/FIR filtering for indoor localization based on wireless sensor networks. *International Conference on Circuits, Systems, Signal and Telecommunications*. 13-15 February. Barcelona, Spain: WSEAS. 103-109.

Pak, J.M., Ahn, C.K., Shi, P., Shmaliy, Y.S., Lim, M.T., (2017a) 'Distributed hybrid particle/FIR filtering for mitigating NLOS effects in TOA-based localization using wireless sensor networks', *IEEE Transactions on Industrial Electronics*, 64(6), 5182-5191.

Pak, J.M., Ahn, C.K., Shmaliy, Y.S., Shi, P., Lim, M.T., (2017b) 'Accurate and reliable human localization using composite particle/FIR filtering', *IEEE Transactions on Human-Machine Systems*, 47(3), 332-342.

Pan, Y., Zheng, N., Tian, Q., Yan, X., Huan, R., (2013). 'Hierarchical resampling algorithm and architecture for distributed particle filters', *Journal of Signal Processing Systems*, 71(3), 237-246.

Pantrigo, J.J., Sánchez, A., Montemayor, A.S., Gianikellis, K., (2009) 'Combining particle filter and population-based metaheuristics for visual articulated motion tracking', *ELCVIA: Electronic Letters On Computer Vision And Image Analysis*, 5(3), 68-83.

Park, S.H., Kim, Y.J., Lee, H.C., Lim, M.T., (2008) Improved Adaptive Particle Filter Using Adjusted Variance And Gradient Data. *International Conference on Multisensor Fusion and Integration for Intelligent Systems*. 20-22 August. Seoul, South Korea: IEEE. 650–655.

Park, S.H., Kim, Y.J., Lim, M.T., (2010) 'Novel adaptive particle filter using adjusted variance and its application', *International Journal of Control, Automation and Systems*, 8(4), 801-807.

Qiu, M., Ming, Z., Li, J., Gai, K., Zong, Z., (2015) 'Phase-change memory optimization for green cloud with genetic algorithm', *IEEE Transactions on Computers*, 64(12), 3528–3540.

Rabiei, E., Droguett, E.L., Modarres, M., (2018) 'Fully adaptive particle filtering algorithm for damage diagnosis and prognosis', *Entropy*, 20(100), 1-15.

Ristic, B., Arulampalam, S., Gordon, N., (2003) *Beyond the kalman filter: particle filters for tracking application*. United Kingdom: Artech House, 1-318.

Robert, S., Künsch, H.R., (2017) 'Localizing the ensemble kalman particle filter', *Tellus A: Dynamic Meteorology and Oceanography*, 69(1), 1-14.

Rosén, O., Medvedev, A., Ekman, M., (2010) Speedup And Tracking Accuracy Evaluation Of Parallel Particle Filter Algorithms Implemented On A Multicore Architecture. *International Conference on Control Applications*. 28 October. Yokohama, Japan: IEEE. 440–445.

Sankaranarayanan, A.C., Srivastava, A., Chellappa, R., (2008) 'Algorithmic and architectural optimizations for computationally efficient particle filtering'. *IEEE Transactions on Image Processing*,17(5), 737-748.

Schwiegelshohn, F., Ossovski, E., Hübner, M., (2016) 'A resampling method for parallel particle filter architectures', *Microprocessors and Microsystem,* 47, 314–320.

Shephard, N., Flury, T., (2009) 'Learning And Filtering Via Simulation: Smoothly Jittered Particle Filters', *Economics Series Working Papers*, 469, 1-27.

Strid, I., (2018) 'Computational Methods for Bayesian Inference in Macroeconomic Models' (PhD). Stockholm School of Economics, Stockholm, Sweeden.

Teixeira, F.C., Quintas, J., Maurya, P., Pascoal, A., (2017) 'Robust particle filter formulations with application to terrain aided navigation', *International Journal of Adaptive Control and Signal Processing*, 31(4), 608-651.

Thrun, S., (2002) Particle Filters in Robotics. *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*. 1-4 August. Alberta, Canada: ACM. 511–518.

Tian, Q., Pan, Y., Salcic, Z., Huan, R., (2017) 'DART: distributed particle filter algorithm with resampling tree for ultimate real-time capability', *Journal of Signal Processing Systems*, 88(1), 29-42 .

Tulsyan, A., Bhushan Gopaluni, R., Khare, S.R., (2016) 'Particle filtering without tears: a primer for beginners', *Computers & Chemical Engineering*, 95, 130-145.

Vadakkepat, P., Jing, L., (2006) 'Improved particle filter in sensor fusion for tracking randomly moving object. *IEEE Transactions on Instrumentation and Measurement*, 55(5), 1823-1832

Van, L.P.J., (2009) 'Particle filtering in geophysical systems'. *Monthly Weather Review*, 137(12), 4089-4114.

Van 't, H.R.J., Rose, L., Bassonga, E., Daroszewska, A., (2017) 'Open source software for semi-automated histomorphometry of bone resorption and formation parameters', *Bone*, 99, 69-79.

Vázquez, M.A., Míguez, J., (2017) 'A robust scheme for distributed particle filtering in wireless sensors networks'. *Signal Processing*, 131(C), 190-201.

Wang, H., Nguang, S.K., (2016) 'Multi-target video tracking based on improved data association and mixed kalman/H_∞ filtering'. *IEEE Sensors Journal,* 16(21), 7693–7704.

Wang, X., Li, T., Sun, S., Corchado, J.M., (2017) 'A survey of recent advances in particle filters and remaining challenges for multitarget tracking'. *Sensors*, 17(12), 1-21.

Wilcox, B.A., Hamano, F., (2017) 'Kalman's Expanding Influence in the Econometrics Discipline'. *IFAC-Papers On Line*, 50(1), 637–644.

Wu, H., Mei, X., Chen, X., Li, J., Wang, J., Mohapatra, P., (2017) 'A novel cooperative localization algorithm using enhanced particle filter technique in maritime search and rescue wireless sensor network', *ISA Transactions*, 78, 39-46

Xu, N., Zhang, Y., Zhang, D., Zhao, S., Fu, W., (2017) 'Moving target tracking in three dimensional space with wireless sensor network', *Wireless Personal Communications*, 94(4), 3403-3413.

Yao, S., Wang, Y., Niu, B., (2015) 'An efficient cascaded filtering retrieval method for big audio data', *IEEE transactions on multimedia*, 17(9), 1450-1459.

Yin, M., Zhang, J., Sun, H., Gu, W., (2011) 'Multi-Cue-Based camshift guided particle filter tracking', *Expert Systems with Applications*, 38(5), 6313-6318.

Yun, X., Jing, Z.-L., (2016) 'Kernel joint visual tracking and recognition based on structured sparse representation', *Neurocomputing*, 193, 181-192

Zhang, Y., Ji, H., Hu, Q., (2018) 'A fast ellipse extended target PHD filter using box-particle implementation', Mechanical System Signal Processing 99, 57–72.

Zhao, Z., Wang, T., Liu, F., Choe, G., Yuan, C., Cui, Z., (2017) 'Remarkable local resampling based on particle filter for visual tracking', *Multimedia Tools and Applications*, 76(1), 835–860.

Zhao, Z., Feng, P., Guo, J., Yuan, C., Wang, T., Liu, F., Zhao, Zhijian, Cui, Z., Wu, B., (2018) 'A hybrid tracking framework based on kernel correlation filtering and particle filtering', *Neurocomputing*, 297, 40-49.

Zhi, R.R., Li, T.C., Siyau, M.F., Sun, S.D., (2014) 'Applied technology in adapting the number of particles while maintaining the diversity in the particle filter', *Advanced Materials Research*, 951, 202-207

Zhou, J., Gu, G., Chen, X., (2018) 'Distributed kalman filtering over wireless sensor networks in the presence of data packet drops', *IEEE Transactions on Automatic Control*, Early Access, 1–1.

Zuo, J., (2013) 'Dynamic resampling for alleviating sample impoverishment of particle filter', *IET Radar, Sonar & Navigation*, 7(9), 968-977.