2022 SSPD

London, UK and Online

2022 SENSOR SIGNAL PROCESSING FOR DEFENCE (SSPD)

13th and 14th September 2022

2022 Sensor Signal Processing for Defence Conference (SSPD)

Proceedings

London, UK

13 - 14 September 2022

IEEE Catalog Number: CFP22SPD-ART (Xplore)
ISBN: 978-1-6654-8348-3 (Xplore)
IEEE Catalog Number: CFP22SPD-USB (USB)
ISBN: 978-1-6654-8347-6 (ISBN)

2022 Sensor Signal Processing for Defence Conference (SSPD)

IEEE Catalog Number: CFP22SPD-ART (Xplore)
ISBN: 978-1-6654-8348-3 (Xplore)
IEEE Catalog Number: CFP22SPD-USB (USB)
SBN: 978-1-6654-8347-6 (ISBN)

Printed copies of this publication are available from:

Curran Associates, Inc
57 Morehouse Lane
Red Hook, NY 12571 USA
Phone: (845) 758-0400
Fax: (845) 758-2633
E-mail: curran@proceedings.com

IEEE eXpress
Conference
Publishing

Produced by **IEEE eXpress Conference Publishing**

For information on producing a conference proceedings and receiving an estimate, contact
conferencepublishing@ieee.org
http://www.ieee.org/conferencepublishing

Table of Contents

Session 1: Applications and Implementation

Session 2: Panel Discussion and Lightning Posters

Session 3: Networking and Communications

Session 4: Machine Learning

Session 6: Radar Sonar and Acoustics

Back to Contents

SSPD Conference 2022 - Welcome

Dear Colleagues,

We warmly welcome you to this year's SSPD Conference, our second hybrid conference. This event is the 11th conference of the Sensor Signal Processing for Defence series and provides a chance to present, listen to and discuss the latest scientific findings in signal processing for defence.

We are privileged to have our two keynote speakers, Frédéric Barbaresco from Thales Land & Air Systems, France and Lieutenant General Tom R Copinger-Symes CBE, Deputy Commander UK Strategic Command. The SSPD 2022 conference also welcomes our invited speakers; Lance M. Kaplan, ARL, Simon Godsill, University of Cambridge, Jon Spencer, Dstl.

A welcome also extends to our panel speakers from Defence, Industry and Academia and the presenters of scientific papers presenting their novel research through live oral presentations. We look forward to some interesting debate and discussion throughout the conference.

We would like to take this opportunity to thank the speakers, reviewers, session chairs and the technical committee for their contribution to this event.

We hope you enjoy our conference.

Mike Davies
Steve McLaughlin
Jordi Barr
Gary Heald

Chairs, SSPD 2022

Technical sponsorship is provided by IEEE Signal Processing Society. Proceedings will be submitted to the Xplore Digital Library. The conference is organised by the University Defence Research Collaboration (UDRC) in Signal Processing, sponsored by the Defence Science and Technology Laboratory (Dstl) and the Engineering and Physical Sciences Research Council (EPSRC).

Sensor Signal Processing for Defence Programme

SSPD Conference

Sensor Signal Processing for Defence Conference 2022 Programme

Location: IET: London Savoy Place / Link to online conference sent in an email to delegates

Note all questions and answers will be managed using Zoom chat. The questions in the poster session will be managed using https://www.sli.do/ – the code for the main conference is #SSPD22 and the password will be sent in an email to delegates.

Tuesday 13th September 2022

8:30 to 9:00 Refreshments

Session 1 – Applications and Implementation – Chair – Mike Davies, University of Edinburgh

9:00 Introduction and Welcome to Day 1/Session 1 – Mike Davies, University of Edinburgh.

9:10 – 10:10 **Defence Keynote Speaker:** Information Challenges in Multi-Domain Integration, Lt Gen Tom Copinger-Symes CBE, UK Strategic Command.

10:10 – 10:40 **Invited Speaker:** Dealing with Epistemic Uncertainty in Information Fusion Systems, Lance Kaplan, ARL.

10:40 – 11:05 Automatic Approximation for 1-Dimensional Feedback-Loop Computations: a PID Benchmark, Yun Wu[1], Yun Zhang[1], Anis Hamadouche[1], Joao Mota[1], Andrew M Wallace[1], [1]Heriot-Watt University.

11:05 – 11:35 Refreshments

11:35 – 12:00 Efficient Joint Surface Detection and Depth Estimation of Single-photon Lidar Data using assumed Density Filtering, Kristofer Drummond[1], Dan Yao[1], Agata Pawlikowska[2], Robert Lamb[2], Steve McLaughlin[1], Yoann Altmann[1], [1]Heriot-Watt University, [2]Leonardo.

Session 2 – Panel Discussion and Lightning Posters – Chair – Jordi Barr - Dstl

12:00 Introduction and Welcome to Session 2 – Jordi Barr, Dstl

12:00 – 13:00 Panel Discussion: Open Source intelligence

13:00 – 13:30 **Lightning Poster Presentations**

- **P1.** An Extension to the Frenet-Serret and Bishop Invariant Extended Kalman Filters for Tracking Accelerating Targets, Joe Gibbs[1], David Anderson[1], Matt MacDonald[2], John Russell[2], [1]University of Glasgow, [2]Leonardo.
- **P2.** Joint Undervolting and Overclocking Power Scaling Approximation on FPGA, Yun Wu[1], Joao Mota[1], Andrew M Wallace[1], [1]Heriot-Watt University.
- **P3.** State Estimation of the Spread of COVID-19 in Saudi Arabia using Extended Kalman Filter, Lamia Alyami[1], Saptarshi Das[1], [1]University of Exeter.
- **P4.** Optimal Bernoulli Point Estimation with Applications, Alexey Narykov[1], Murat Uney[1], Jason F. Ralph[1], [1]University of Liverpool.

- **P5.** High Resolution DOA Estimation for Contiguous Target with Large Power Difference, Murtiza Ali[1], Karan Nathwani[1], [1]Indian Institute of Technology.

- **P6.** Compressive Self-Noise Cancellation in Underwater Acoustics, Pawan Kumar[1], Karan Nathwani[1], Vinayak Abrol[2], Suresh Kumar[3], [1]Indian Institute of Technology, [2]University of Oxford, [3]DRDO, India.

- **P7.** Non-Coherent Discrete Chirp Fourier Transform for Modulated LFM Parameter Estimation, Kaiyu Zhang[1], Fraser K Coutts[1], John Thompson[1], [1]University of Edinburgh.

- **P8.** Unsupervised Expectation Propagation Method for Large-Scale Sparse Linear Inverse Problems, Dan Yao[1], Steve McLaughlin[1], Yoann Altmann[1], [1]Heriot-Watt University.

- **P9.** Movement Classification and Segmentation Using Event-Based Sensing and Spiking Neural Networks, Paul Kirkland[1], Gaetano Di Caterina[1], [1]University of Strathclyde.

- **P10.** Enhanced Space-Time Covariance Estimation Based on a System Identification Approach, Faizan Khattak; Ian Proudler[1], Stephan Weiss[1], [1]University of Strathclyde.

13:30 – 14:45 Lunch and Poster Presentations – There will be an opportunity to view posters either online or at Savoy Place (Q & A will use https://www.sli.do)

Session 3 Networking and Communications – Chair – Steve McLaughlin, Heriot-Watt University

14:45 Introduction and Welcome to Session 3 – Steve McLaughlin, Heriot-Watt University

14:45 OMASGAN: Out-of-distribution Minimum Anomaly Score GAN for Anomaly Detection, Nikolaos Dionelis[1], Sotirios Tsaftaris[1], Mehrdad Yaghoobi[1], [1]University of Edinburgh.

15:10 Refreshments

15:45 Fast Trajectory Forecasting With Automatic Identification System Broadcasts, Yicheng Wang[1], Murat Uney[1], [1]University of Liverpool.

16:10 Deep Learning for Spectral Filling in Radio Frequency Applications, Michael Girard[1], Matthew Setzler[1], Elizabeth Coda[1], Jeremiah Rounds[1], Michael Vann[1], [1]Pacific Northwest National Laboratory.

16:35 Closing remarks

19:30 Conference Reception Drinks - IET Savoy Place

20:00 Conference Dinner

Back to Contents

Sensor Signal Processing for Defence Programme

Wednesday 14th September 2022

8:30 to 9:00 Refreshments

Session 4 Machine Learning – Chair – James Hopgood, University of Edinburgh

9:00 Introduction and Welcome to Day 2/Session 4 – Machine Learning – James Hopgood, University of Edinburgh

9:05 – 10:05 Academic Keynote Speaker: Lie Groups Statistics and Machine Learning for Military Sensors based on Symplectic Structures of Information Geometry, Frédéric Barbaresco, Thales

10:05 – 10:35 Invited Speaker: Signal Processing for Military Communications, Jon Spencer, Dstl.

10:35 – 11:00 Robust DOA Estimation Based on Deep Neural Networks in Presence of Array Phase Errors, Xuyu Gao[2], Aifei Liu[2], Yutao Xiong[2], [1]Harbin Engineering University, [2]Northwestern Polytechnical University.

11:00 – 11:25 Refreshments

Session 5 – Panel Discussion – Chair – Jordi Barr - Dstl

11:25 Introduction and Welcome to Session 5 – Jordi Barr, Dstl

11:25 – 12:25 Panel Discussion: Should defence be more university friendly or should universities be more defence friendly?

12:25 – 13:25 Lunch

Session 6 – Radar Sonar and Acoustics – Chair – Gary Heald, Dstl

13:25 Introduction and Welcome to Session 6 – Gary Heald, Dstl

13:25– 13:55 Invited Speaker: Points, Particles and Positions: Recent Advances in Distributed Processing of Agile Objects, Simon Godsill, University of Cambridge.

13:55 – 14:20 A Polynomial Subspace Projection Approach for the Detection of Weak Voice Activity, Vincent W Neo[1], Stephan Weiss[2], Patrick A Naylor[1], [1]Imperial College London, [2]University of Strathclyde.

14:20 – 14:45 Optimizing Sonobuoy Placement using Multiobjective Machine Learning, Christopher M Taylor[1], Simon Maskell[1], Jason F. Ralph[1], [1]University of Liverpool.

14:45 – 15:10 Refreshments

15:10 – 15:35 Image Quality SAR Refocus of Moving Targets undergoing Complicated Rolling Maneuvers, David A. Garren[1], [1]Naval Postgraduate School.

15:35 – 16:00 Learning Low-Rank Models From Compressive Measurements for Efficient Projection Design, Fraser K Coutts[1], John Thompson[1], Bernard Mulgrew[1], [1]University of Edinburgh.

16:00 – 16:25 LoRaWAN Performance Evaluation and Resilience under Jamming Attacks, Vaia Kalokidou[1], Manish Nair[1], Mark Beach[1], [1]University of Bristol.

16:25 Closing remarks

Back to Contents

Keynote Speakers

Lieutenant General T R Copinger-Symes CBE Deputy Commander UK Strategic Command

Tom spent his early career with The Rifles on operations in Northern Ireland, Bosnia, Kosovo, Iraq and Afghanistan, and in operational and strategy posts at the Permanent Joint Headquarters and the Ministry of Defence.

For the past 10 years he has focused on how the Army and Defence can make better use of its data and information, whether in supporting traditional warfighting or employed as a weapon in its own right - especially in the context of 'sub-threshold' competition. This has included command at brigade (1 ISR Bde) and divisional levels (Force Troops Command - now 6th (UK) Div), as well as, in his last post as Director of Military Digitisation, leading Defence's Digital Transformation portfolio.

In May 2022 Tom was promoted to Lieutenant General, on appointment as the Deputy Commander of UK Strategic Command.

Frédéric Barbaresco, THALES KTD PCC SENSING SEGMENT LEADER (Key Technology Domain: Processing Control & Cognition), THALES Land & Air Systems, Meudon, FRANCE

Senior THALES Expert in Artificial Intelligence at the Technical Department of THALES Land & Air Systems. SMART SENSORS Segment Leader for the THALES Corporate Technical Department (Key Technology Domain "Processing, Control & Cognition"). THALES representative at the AI Expert Group of ASD (AeroSpace and Defense Industries Association of Europe). 2014 Aymée Poirson Prize of the French Academy of Science for the application of science to industry. Ampère Medal, Emeritus Member of the SEE, and President of the SEE ISIC club "Information and Communication Systems Engineering". He is French MC representative of European COST CaLISTA (Cartan geometry, Lie, Integrable Systems, quantum group Theories for Applications) (https://www.cost.eu/actions/CA21109/). General Chair of the following events: the "Geometric Science of Information" international conferences (https://franknielsen.github.io/GSI/), MaxEnt'22 conference at Institut Henri Poincaré (https://maxent22.see.asso.fr/), Ecole de Physique des Houches SPIGL'20 in July 2020 on « Joint Structures and Common Foundations of Statistical Physics, Information Geometry and Inference for Learning » (https://franknielsen.github.io/SPIG-LesHouches2020/) and FGSI'19 Conference "Foundations of Geometric Structures of Information" in February 2019 at IMAG "Institut Montpellierain Alexander Grothendieck" (https://fgsi2019.sciencesconf.org/). CIRM Luminy Seminar organizer of TGSI'17 "Topological and Geometrical Structures of Information" (http://forum.cs-dc.org/topic/361/tgsi2017-presentation-organisation-abstract-submission). Guest Editors of Special Issues "Lie Group Machine Learning and Lie Group Structure Preserving Integrators". Author of more than 200 scientific publications and more than 20 patents.

Abstract: Lie Groups Statistics and Machine Learning for Military Sensors based on Symplectic Structures of Information Geometry

In a first part, we will present pioneering THALES Sensors/Radars algorithms: Geometric Matrix CFAR based on Jean-Louis Koszul's Information Geometry and its extension for STAP, Complex-Valued Convolutional Neural Networks and Covariance-Matrix-Valued HPDNet for Micro-Doppler ATDR, Lie Group-based Convolutional Equivariant Neural Network from Geometric Deep Learning for Doppler clutter map, IEKF (Invariant Extended Kalman Filter) Frenet-Serret Tracker based on Lie Groups for hyper-maneuvering targets, Tracker parameters tuning by Deep Learning and finally, Multi-Agent Reinforcement Learning for Radar Task Scheduling and Active-Track/TWS collaborative Resources Management. In a second part, we will present Avant-Garde tools using statistics on Lie Groups for different sensors applications (detection, tracking and recognition). From French Jean-Marie Souriau's Symplectic Model of Statistical Physics and Russian Kirillov's Representation Theory of Lie Groups, we will introduce Gaussian statistical density for Lie Groups defined as Maximum Entropy Gibbs density on coadjoint orbits though moment map. This Symplectic model of Information gives new geometric foundation for Entropy, defined purely geometrically (and no longer axiomatically) as Casimir Invariant Function in Coadjoint Representation. We will conclude with new perspectives opened by this new Symplectic Theory of Heat and Information.

Invited Speakers

Lance M. Kaplan, ARL

Lance M. Kaplan received his undergraduate degree at Duke University in 1989 and a PhD degree from the University of Southern California in 1994, all in Electrical Engineering. H e held a National Science Foundation Graduate Fellowship and a USC Dean's Merit Fellowship from 1990–1993. Dr. Kaplan previously worked at the Georgia Tech Research Institute (1987-1990) and the Hughes Aircraft Company (1994-1996). He was a faculty member in the Department of Engineering at Clark Atlanta University from 1996-2004. Currently, he is a team leader in the Context Aware Processing branch of the DEVCOM Army Research Laboratory (ARL). Dr. Kaplan serves as VP Publications for the IEEE Aerospace and Electronic Systems (AES) Society (2021-Present) and as VP Conferences for the International Society of Information Fusion (ISIF) (2014-Present). Previously, he served as Editor-In-Chief for the IEEE Transactions on AES (2012-2017), on the Board of Governors for the IEEE AES Society (2008-2013, 2018-2020) and on the Board of Directors of ISIF (2012-2014). He is a Fellow of IEEE and of ARL. His current research interests include information/data fusion, reasoning under uncertainty, network science, resource management and signal and image processing.

Abstract: Dealing with Epistemic Uncertainty in Information Fusion Systems

Information fusion is basically the weighted averaging of data from different sources where the weights are inversely proportional to the uncertainty for the data sources. Generally, the uncertainty is aggregated from likelihood models to characterize the probability of the unknown states in light of the observations. In many fusion systems, the likelihood functions are presumed to be known, but in practice they must be machine learned via a calibration process. In Army applications, there can be little training data to accurately learn these likelihoods. This talk will address the epistemic uncertainty as a second-order uncertainty about the likelihoods in cases where very little training exists. Specifically, the talk will highlight new methods to compute error bars around probabilistic outputs of Bayesian and neural networks. Furthermore, it enables new paradigms for establishing prediction sets of feasible hypotheses rather than the most likely hypothesis, which can be very misleading in cases of imbalance of epistemic uncertainty.

Professor Simon Godsill, University of Cambridge

Simon Godsill is Professor of Statistical Signal Processing in the Engineering Department at Cambridge University. He is also a Professorial Fellow and tutor at Corpus Christi College Cambridge. He coordinates an active research group in Signal Inference and its Applications within the Signal Processing and Communications Laboratory at Cambridge, specializing in Bayesian computational methodology, multiple object tracking, audio and music processing, and financial time series modeling. A particular methodological theme over recent years has been the development of novel techniques for optimal Bayesian filtering and smoothing, using Sequential Monte Carlo or Particle Filtering methods. Prof. Godsill has published extensively in journals, books and international conference proceedings, and has given a number of high profile invited and plenary addresses at conferences such as the Valencia conference on Bayesian Statistics, the IEEE Statistical Signal Processing Workshop and the Conference on Bayesian Inference for Stochasrtic Processes (BISP). He co-authored a seminal Springer text Digital Audio Restoration with Prof. Peter Rayner in 1998. He was technical chair of the successful IEEE NSSPW workshop in 2006 on sequential and nonlinear filtering methods, and has been on the conference panel for numerous other conferences/workshops. Prof. Godsill has served as Associate Editor for IEEE Tr. Signal Processing and the journal Bayesian Analysis. He was Theme Leader in Tracking and Reasoning over Time for the UK's Data and Information Fusion Defence Technology Centre (DIF-DTC) and Principal Investigator on many grants funded by the EU, EPSRC, QinetiQ, General Dynamics, MOD, Microsoft UK, Citibank and Mastercard. In 2009-10 he was co-organiser of an 18 month research program in Sequential Monte Carlo Methods at the SAMSI Institute in North Carolina. He is a Director of CEDAR Audio Ltd. (which has received numerous accolades over the years, including a technical Oscar).

Abstract: Points, particles and positions: recent advances in distributed processing of agile objects

In this talk I will discuss models developed under the SIGNeTS project for agile motion of objects. I will describe new motion and observation models based on point process theory and Levy processes, as well as new advances in Gaussian process models for nonparametric modelling of motion, and will further discuss methods for distributed processing of sensor data using these models, as well as inference about target detection rates and clutter rates. The methodology is probabilistic and implemented using combinations of particle filtering and variational methods.

Back to Contents

Jon Spencer CPhys FInstP, Dstl Comms & Nets Programme Chief Scientist

Jon is the Chief Communications and Networks Scientist at the Defence Science and Technology Laboratory (Dstl), part of the UK Ministry of Defence. Jon leads the delivery of communications research spanning all military domains from subsea to space, focusing on development of next-generation and generation-after-next resilient systems to enable information driven operations in the most challenging environments.

As lead scientist for the Communications and Networks programme Jon coordinates research to develop and demonstrate the advanced concepts that will enable Information Advantage in the contested environments of the future. Working with allies and partners from UK industry and academia we are investing in research both to bring forward the military capabilities essential for future operations and to stimulate the development of skills and facilities in the supply chain.

The work is wide ranging. It stretches from fundamental physical research into the propagation environment; maturing novel communications concepts such as Quantum communications; developing new ideas for networking in very congested and dynamic environments through to developing the architectures needed to enable rapid integration and adaptation.

Jon has been active in the development of tactical communications and networking capabilities for 25 years, both in government research and in industry where he led a number of successful product developments. Jon is a Fellow of the Institute of Physics.

Abstract: Multi-Spectral and Multi-Modal Underwater Acoustic Imaging

Communications and Networks are fundamental enablers to military capability. This talk will explain some of the fundamental threats and technical challenges faced when delivering communications and networks capability for military operations. UK Ministry of Defence has recently announced a significant investment in communications and networks research to address these challenges and an overview of that programme will be presented along with opportunities to contribute. Access to appropriate signal processing techniques is essential to this and the talk will discuss some of the signal processing challenges to enable covert and overt communications.

Back to Contents

SSPD2022 Conference Committee

General Chairs

- Mike Davies - University of Edinburgh
- Stephen McLaughlin - Heriot-Watt University
- Jordi Barr - Dstl
- Gary Heald - Dstl

Publicity and Local Arrangements Chair

- Janet Forbes - University of Edinburgh

Technical Programme Committee

Abderrahim Halimi - Heriot-Watt University

Alasdair Hunter - Dstl

Andreas Ahrens - Hochschule Wismar

Andrew Wallace - Heriot-Watt University

Andy Stove - Stove Specialties

Athanasios Gkelias - Imperial College London

Augusto Aubry - Universita degli studi di Napoli

Bernard Mulgrew - University of Edinburgh

Brian Barber - Dstl

Bruno Clerckx - Imperial College London

Carmine Clemente - University of Strathclyde

Chris Baker - University of Birmingham

Christoph Wasserzier - Fraunhofer Institute for High Frequency Physics and Radar Techniques FHR

Christos Ilioudis - University of Strathclyde

Cristian Rusu - University of Edinburgh

Back to Contents

Dave Harvey - Thales

David Blacknell - Dstl

David Cormack - Leonardo

David Garren - Naval Postgraduate School

David Greig - Leonardo

Domenico Gaglione - Centre for Maritime Research and Experimentation (CMRE)

Duncan Williams - Dstl

Emma Goldstein - Dstl

Geert Leus - Delft University of Technology

Harvey Alison - Leonardo

Henry Gouk - University of Edinburgh

Ian Proudler - University of Strathclyde

Ivo Bukovsky - Czech Technical University in Prague

James Hopgood - University of Edinburgh

Jason Ralph - University of Liverpool

Joao Mota - Heriot-Watt University

John Thompson - University of Edinburgh

Julian Deeks - Dstl

Ken McEwan - Dstl

Krishnaprasad Nambur Ramamohan - Delft University of Technology

Lyudmila Mihaylova - University of Sheffield

Mahesh Banavar - Clarkson University

Maria Greco - University of Pisa

Back to Contents

Mark Hadley - Kaon Limited

Mathini Sellathurai - Heriot-Watt University

Mehrdad Yaghoobi - University of Edinburgh

Murat Uney - University of Liverpool

Neil Cade - Leonardo

Nikolaos Dionelis - University of Edinburgh

Oliver Sims - Leonardo

Paul Thomas - Dstl

Richard Jackson - Dstl

Sami Aldalahmeh - Al-Zaytoonah University of Jordan

Sen Wang - Heriot-Watt University

Simon Godsill - University of Cambridge

Simon Maskell - University of Liverpool

Stephan Weiss - University of Strathclyde

Stephen Ablett - Dstl

Suresh Jacob - Dstl

Vladimir Stankovic - University of Strathclyde

Wenwu Wang - University of Surrey

Wolfgang Koch - Fraunhofer FKIE

Yoann Altmann - Heriot-Watt University

Back to Contents

Automatic Approximation for 1-Dimensional Feedback-Loop Computations: a PID Benchmark

Yun Wu, Yun Zhang, Anis Hamadouche, João F. C. Mota, Andrew M. Wallace
School of Engineering and Physical Sciences
Heriot-Watt University, Edinburgh, UK
{y.wu, y.zhang, ah225, j.mota, a.m.wallace}@hw.ac.uk

***Abstract*—The analysis and optimization of computational precision is crucial when using approximation in hardware implementations of algorithms. Mainstream methods are based on either dynamic or static analysis of arithmetic errors, but only static analysis can guarantee the desired worst-case accuracy. In this paper we describe an automated approach to estimate the arithmetic binary representations and compare the computational sensitivities for 1-dimensional feedback-loop algorithms, enabling both customized floating-point and fixed-point approximation by affine arithmetic.**

Using typical benchmarks for iterative Proportional Integral Derivative (PID) control, an automated approach has been developed to obtain the appropriate approximation for both the exponent and mantissa of floating-point, and the integer and fraction parts of fixed-point signals. This reduces the circuit area and power consumption of an FPGA implementation. For the approximate PID controller implemented on a Xilinx FPGA platform, we were able to reduce area and power, as compared to standard uniform bit-widths, by 62% and 27% on average respectively.

***Index Terms*—Approximate Computing, Affine Arithmetic, PID Controller, Field Programmable Gate Array**

I. Introduction

Field Programmable Gate Arrays (FPGAs) enable customized bit-width for algorithm approximation. In order to reduce the area and increase the speed, the numerical representation must be chosen carefully to meet the design requirements. In resource constrained systems [1], the challenge is to select a less precise representation that reduces area and power consumption, while maintaining algorithmic performance.

Finding the optimal precision is NP-hard, whether for static or dynamic analysis. It is has been claimed that offline, static analysis provides a overly conservative estimate [2]. To estimate the precision in bit-width, some works have focused on empirical learning from the signal or data variation during simulation [3], while others have focused on analysis of the signal and data uncertainty during processing [4], [5]. The latter approach has shown more accurate estimation of arithmetic bit-width, using either Interval Arithmetic (IA) or Affine Arithmetic (AA) [6].

However, previous research on automated bit-width estimation has been applied to simple feed-forward linear functions such as simple polynomials [7], the Discrete Cosine Transform (DCT) (8×8) [5], and small (2×2) matrix multiplication [4].

PID control is widely adopted in robotics and autonomous systems in both the commercial and defence sectors. We investigate a more complex procedure, proportional, integral, derivative (PID) controller, which stabilizes open loop systems by a feedback mechanism [8]. We have implemented an embedded PID controller on an FPGA to reduce power consumption and gain better performance. Previous authors have determined the required precision heuristically [9]–[11]. In this paper, we apply automated approximation of precision to PID benchmarks ([12]) using AA, addressing a problem with considerably more complex computational uncertainty than previous work.

Contributions. We summarize our contributions as follows:

- We develop a framework for automated estimation of precision for feedback-loop computations.
- We evaluate the results of our estimates of precision using established PID benchmarks.
- We demonstrate the approximated PID on an FPGA and demonstrate significant resource/power savings.

In Section II, we briefly introduce Affine Arithmetic. In Section III, we briefly discuss embedded PID implementations and present our method for modeling the iterative error and to estimate precision. Using the PID controller benchmarks, we demonstrate the effectiveness of our approach in Section IV. Conclusions are given in Section V.

II. Affine Arithmetic

Affine arithmetic (AA) is one of many proposed models for function self-validation, which overcomes the explosion of errors in standard Interval Arithmetic (IA) [6]. In AA, the uncertainty of variable a can be represented as

$$\hat{a} = a_0 + \sum_{i=1}^{n} a_i \cdot \epsilon_i, \tag{1}$$

where each ϵ_i is an interval $[-1, 1]$, and each a_i is the partial deviation. The term $a_i \cdot \epsilon_i$ represents the uncertainty on a caused by some underlying uncertainty i. By using the form in Eq.(1), it not only captures the correlations during computation, but also realizes the symbolic error cancellation of uncertainties.

Addition and subtraction in AA can be expressed by

$$\hat{a} \pm \hat{b} = (a_0 \pm b_0) + \sum_{i=1}^{n} (a_i \pm b_i) \cdot \epsilon_i, \tag{2}$$

Yun Zhang is a visiting scholar at Heriot-Watt University, from the Faculty of Information, Ocean University of China. This work is supported by EPSRC Grant number EP/S000631/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

while multiplication with nonlinear terms can be approximated as

$$\hat{a}\cdot\hat{b} = (a_0\cdot b_0)+\sum_{i=1}^{n}(a_0\cdot b_i+b_0\cdot a_i)\cdot\epsilon_i+(\sum_{i=1}^{n}|a_i|)\cdot(\sum_{i=1}^{n}|b_i|). \quad (3)$$

There have been many efforts to create fast and efficient AA implementations, such as libaa [13], libaffa [14], YalAA [15], and hafar [16]. The latest YalAA library in C/C++, supports most mathematical functions. Given the lower and upper bounds on elementary functions, YalAA can use Chebyshev interpolation to approximate non-affine functions. Hence, for each input data, an interval must be pre-defined to allow further range analysis.

III. PID Control using Affine Arithmetic

Embedded PID implementations on FPGAs have used various levels of precision. Lima [9] evaluated various bit-width, fixed-point PID implementations using repeated simulation to achieve optimal precision. Kocur [10] used a specific low precision for a PID on an FPGA constrained by the allowable bit-width of the DAC/ADC interface. Recent research on more complex PID controllers with neural network self-tuning [11], [17] also considered specific bit-width fixed-point implementations, determined heuristically to satisfy the performance requirements.

In our work, we present automated precision design of a PID controller applied to benchmark designs using arbitrary bit-widths for both floating- and fixed-point binary representations. In Fig 1, the control flow for a set of PID benchmarks plant models uses pre-defined AA variables.

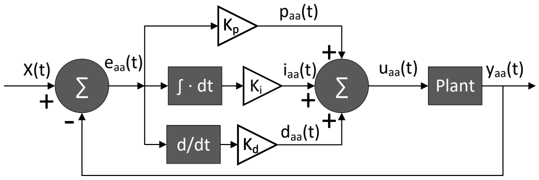


Fig. 1: PID Control Flow using AA

The PID controller assumes that the target system state, $X(t)$, is known as prior knowledge. We use the YalAA library [15] as the foundation of the AA computations. The interval of the AA output is used to derive the posterior information for determining the binary representation of floating-point and fixed-point arithmetic.

With reference to Fig. 1, the control variable $u_{aa}(t)$ is based on the difference between a function of the error between the desired system output state $X(t)$ and the current output state $y_{aa}(t)$, having proportional, derivative and integral terms.

$$u_{aa}(t) = K_p e_{aa}(t) + K_i \int e_{aa}(t)dt + K_d \frac{de_{aa}(t)}{dt}, \quad (4)$$

where K_p, K_i, and K_d are the controller coefficients of the proportional, integral and derivative paths. The plant (system) model to be controlled by the PID is represented commonly by a transfer function (TF), taking the division of the given system output state $Y(s)$ over the control input $U(s)$ using Laplace transforms.

A. Iterative Uncertainty

The targeted iterative uncertainty comes from the approximate precision, which is mainly caused by the propagated error of the arithmetic operations during the feedback link as shown in Fig. 1. The approximate precision uses the truncated bits of the fractional arithmetic representation, where the overflow or the saturated bits of the integer arithmetic representation are guaranteed by the given accuracy of the chosen bit-width. We focus on minimizing the bit-width of the mantissa and fraction bits for either floating-point or fixed-point arithmetic correspondingly. The posterior error information is obtained through AA computation of an iterative PID plant controller, where the interval of the output is used to adjust the proportional precision within a certain tolerance.

From the interval analysis of the computational outcome by AA, the guaranteed upper and lower bounds of each term in the PID controller are derived, e.g. $[lb, ub]$. The difference between the lower bound lb and upper bound ub is adopted as the range of potential computational error since AA provides a tighter outcome. Eq.(5) links the the interval to the precision error

$$|\frac{\upsilon-\hat{\upsilon}}{\upsilon}| \leq \omega, \quad (5)$$

where $\upsilon = (ub - lb)/2$ is the derived output interval from AA computation, $\hat{\upsilon}$ is the approximated value after adjusting the bit-width, and ω is the targeted relative error threshold (10^{-3} in this work). The corresponding bit-width adjustment is introduced in the next section, where Eq.(5) is used as the criterion to determine the precision.

B. Automated Bit-Width Adjustment

Adopting the output interval, the binary representations of both floating-point and fixed-point arithmetic are adjusted automatically, maintaining sufficient bits in the exponent and integer parts and truncating the mantissa and fraction parts. Fig. 2 shows the overall design flow from application kernel through bit-width estimation to accelerator generation for the Xilinx FPGA platform.

This assumes that the application kernel, the PID controller in our case, is given by the algorithm developer as a C++ template, which can be applied to arbitrary arithmetic types. Calling the YalAA library, the affine arithmetic data type is adopted and the application is computed with the pre-defined interval of the affine arithmetic data. The AA based control process provides a tighter interval of the application outcome, which is used to estimate the required bit-width as described in section III-B1 and III-B2. With the estimated bit-width, the application kernel is transformed into custom precision and synthesized using the Xilinx tool-set to generate the description of the hardware accelerator. The entire process is fully automated with the exception of the user input application kernel.

1) Floating-Point: The typical floating-point binary representation has three parts: sign, exponent, and mantissa, where the single precision floating-point value of υ, using 32 bits, can be represented as Eq (6).

$$\upsilon_{fp} = (-1)^S \times M \times 2^{127-E}, \quad (6)$$

Back to Contents

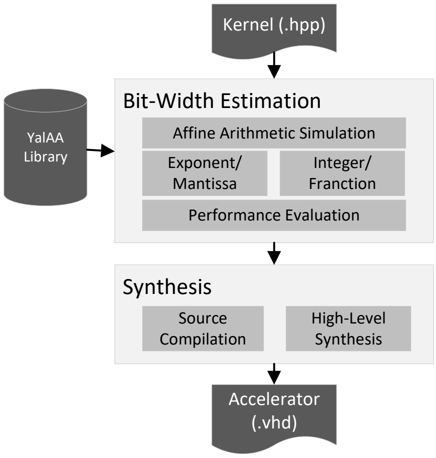


Fig. 2: Automated Design Flow

where S has 1-bit, M has 23-bits (bw_m) and E has 8-bits (bw_e).

To determine the bit-width, the binary representation of υ is truncated for both the exponent and mantissa to an approximate value $\hat{\upsilon}$, fulfilling the condition of Eq. (5).

$$\hat{\upsilon}_{fp} = (-1)^S \times \hat{M} \times 2^{(2^{(b\hat{w}_e - 1)} - 1) - \hat{E}}, \tag{7}$$

where $\hat{M}$ and $\hat{E}$ are the estimated exponent and mantissa with the new bit-widths $b\hat{w}_m$ and $b\hat{w}_e$ respectively. Algorithm 1 describes the process to derive $b\hat{w}_m$ and $b\hat{w}_e$, as well as $\hat{E}$ and $\hat{M}$.

Algorithm 1 Floating-Point Bit-Width Estimation

```
Input: $\upsilon$
Output: $\hat{M}, \hat{E}, b\hat{w}_m, b\hat{w}_e$
1: $b\hat{w}_e \leftarrow \arg\min_{b\hat{w}_e}((2^{b\hat{w}_e - 1} - 1) - (127 - E)) > 0$
2: $b\hat{w}_e \leftarrow b\hat{w}_e + 1$
3: $\hat{E} \leftarrow (2^{b\hat{w}_e - 1} - 1) - (127 - E)$
4: $b\hat{w}_m \leftarrow \arg\max_{b\hat{w}_m} \left\| \frac{\upsilon_{fp} - \hat{\upsilon}_{fp}}{\upsilon_{fp}} \right\|_1 \leq \omega,$
5: $\hat{M} = (M >> b\hat{w}_m) \& (2^{(b\hat{w}_m + 1)} - 1)$
```

2) Fixed-Point: The typical fixed-point binary representation has three parts: sign, integer, and fraction, where a fixed-point value of υ can be represented as Eq (8).

$$\upsilon_{fxp} = (-1)^S \times (2^I + 2^{-F}), \tag{8}$$

and S has 1-bit, I has bw_i bits and E has bw_f bits.

Similarly, the binary representation of υ is truncated at both the integer and fraction parts into the approximate value $\hat{\upsilon}$, fulfilling the condition of Eq. (5).

$$\hat{\upsilon}_{fxp} = (-1)^S \times (2^{\hat{I}} + 2^{-\hat{F}}), \tag{9}$$

where $\hat{I}$ and $\hat{F}$ are the estimated integer and fraction parts with the new bit-width $b\hat{w}_i$ and $b\hat{w}_f$ respectively. Algorithm 2 describes the process to derive $b\hat{w}_i$ and $b\hat{w}_f$, as well as $\hat{I}$ and $\hat{F}$.

Algorithm 2 Fixed-Point Bit-Width Estimation

```
Input: $\upsilon$
Output: $\hat{I}, \hat{F}, b\hat{w}_i, b\hat{w}_f$
1: $b\hat{w}_i \leftarrow \arg\min_{b\hat{w}_i}(2^{b\hat{w}_i - 1} - I) > 0$
2: $b\hat{w}_i \leftarrow b\hat{w}_i + 2$
3: $b\hat{w}_f \leftarrow \arg\max_{b\hat{w}_f} \left\| \frac{\upsilon_{fxp} - \hat{\upsilon}_{fxp}}{\upsilon_{fxp}} \right\|_1 \leq \omega,$
4: $\hat{F} = (F >> b\hat{w}_f) \& (2^{(b\hat{w}_f + 1)} - 1)$
```

IV. Experiments

Using a PID benchmark with various plant models, our approach to automated approximation has been evaluated. We derive firstly the system state model from the transfer function as shown in Table I, and build the PID control process using Eq. (4) using the C++ templates. This is considered as the application kernel for both the later affine arithmetic computation and hardware accelerator generation. Note that the aim of this work is not to optimize the PID controller itself but the computational precision of its iterative control process. Hence, the optimal control coefficients K_p, K_i, and K_d are assumed to be given.

A. Affine Arithmetic Computation

The developed kernels are executed by calling the YalAA library to produce the interval information of the system outcome. To adopt the YalAA library, we need to set up an initial interval for each input variable to be updated in the PID control system. We record the differential error term e_{aa}, all control feedback terms $p_{aa}, i_{aa}, d_{aa}, pid_{aa}$ and the system output y_{aa} for later analysis, where we set the initial intervals of $p_{aa}, i_{aa}, d_{aa}, pid_{aa}$ to [0, 2] and e_{aa} to [-1, 1] empirically. Using affine arithmetic through the YalAA library, we generate both the affine form of the variables as well as its final interval outcome, which can be used as the variable interval information to estimate the bit-width in Section III-B1 and III-B2.

Table I records the resulting intervals of the PID controller for nine benchmarks, marked as c1-c9, using affine arithmetic. Using AA to execute the PID control process, the final intervals are tightened to small ranges. e_{aa} and p_{aa} are generally similar, but the other variables are different for the different system models. The interval of the control output pid_{aa} and the system output y_{aa} converge within similar ranges.

Using the approach of Section III-B, each term can be used to estimate a set of bit-widths for both floating-point and fixed-point arithmetic. The maximum widths for all the estimations are chosen as the final values.

B. Controller Performance

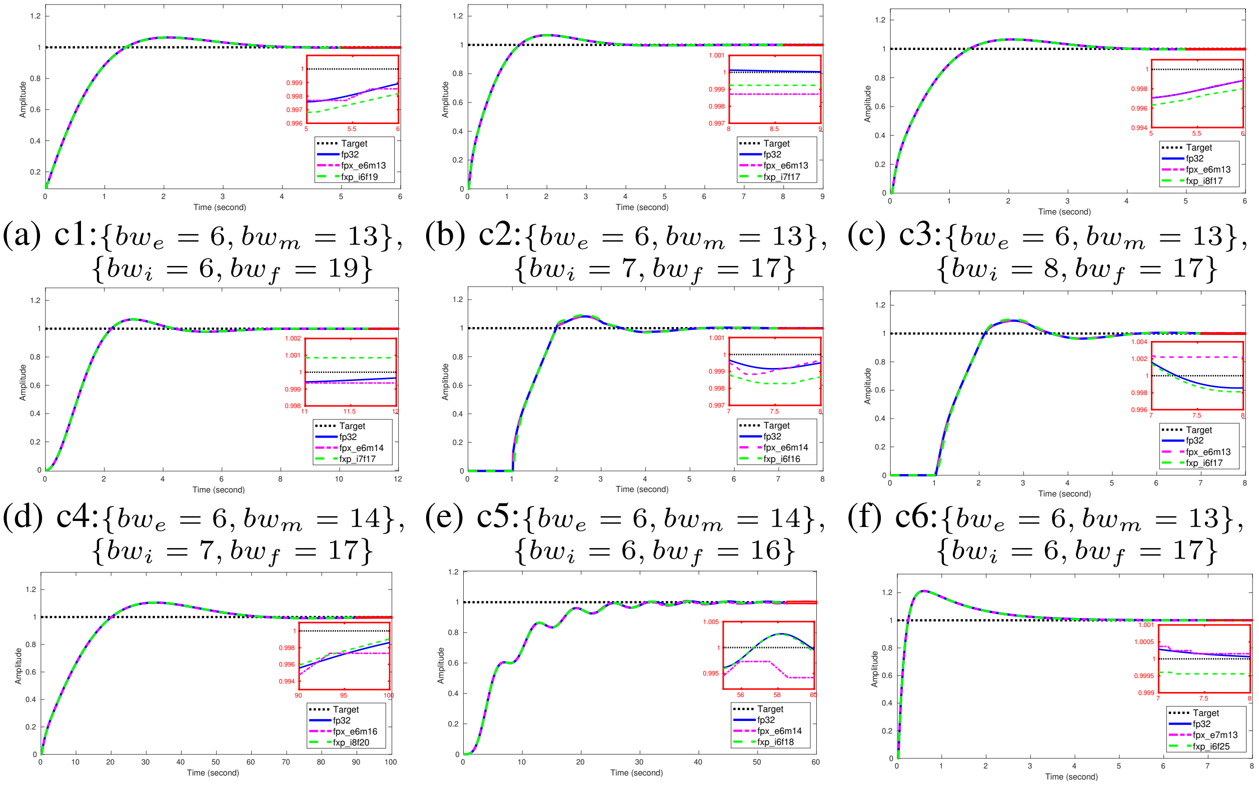
Using the estimated bit-width, the control performance is evaluated by executing at the given precision for the nine PID benchmarks in Table I. Fig. 3 shows the control performance of the approximated PID controller on various plant models, zoomed-in on the later iterations to show more detailed performance comparisons.

Single floating-point (32 bits), marked as $fp32$, is used as a baseline for full precision. The custom floating-point (fp), and fixed-point (fxp) are marked with their corresponding bit-width number. Since S is always 1 bit, only

Back to Contents

TABLE I: AA outcome for PID Parameters ($[lb, ub]$)

Plant Transfer Function	e_{aa}	p_{aa}	i_{aa}	d_{aa}	u_{aa}	y_{aa}
c1: $\frac{1}{s+1}$	[-0.001705, 0.001238]	[-0.001705, 0.001238]	[0.384654, 0.387893]	[-0.007070, 0.003848]	[0.997140, 1.006484]	[0.998771, 1.001712]
c2: $\frac{1}{(0.1s+1)(s+1)}$	[-0.010378, 0.020331]	[-0.010378, 0.020331]	[0.352566, 0.359014]	[-0.036406, 0.014746]	[0.994849, 1.016622]	[0.979850, 1.010305]
c3: $\frac{1}{(0.01s+1)(0.1s+1)(s+1)}$	[-0.002425, 0.001583]	[-0.002425, 0.001583]	[0.365121, 0.370106]	[-0.009494, 0.005258]	[0.996281, 1.007964]	[0.998431, 1.002432]
c4: $\frac{1-0.1s}{(s+1)^3}$	[-0.006933, 0.006462]	[-0.006933, 0.006462]	[1.183874, 5.947544]	[-0.002716, 0.004606]	[0.999921, 5.001432]	[0.993552, 1.00691]
c5: $\frac{1}{(0.1s+1)}e^{-s}$	[-0.004284, 0.008109]	[-0.004284, 0.008109]	[1.314815, 1.349334]	[-0.004763, 0.002516]	[0.982410, 1.004009]	[0.991915, 1.004271]
c6: $\frac{1}{(0.1s+1)^2}e^{-s}$	[-0.005145, 0.011130]	[-0.005145, 0.011130]	[1.400613, 1.443956]	[-0.006247, 0.002888]	[0.980136, 1.004560]	[0.988901, 1.005131]
c7: $\frac{100}{(s+10)^2}(\frac{1}{(s+1)} + \frac{0.5}{(s+0.05)})$	[0.004182, 0.005610]	[0.004182, 0.005610]	[5.336462, 15.987360]	[-0.000419, 0.000166]	[0.091070, 0.271908]	[0.994390, 0.995820]
c8: $\frac{1}{(s+1)(s^2+0.2s+1)}$	[-0.071520, 0.004774]	[-0.071520, 0.004774]	[8.021174, 8.025332]	[-0.000687, 0.008533]	[0.999168, 0.999939]	[0.995228, 1.071478]
c9: $\frac{1}{(s^2-1)}$	[-0.000135, 0.000082]	[-0.000135, 0.000082]	[-0.201893, -0.201722]	[-0.000092, 0.000159]	[-1.000015, -0.999961]	[0.999918, 1.000134]



(a) c1:$\{bw_e = 6, bw_m = 13\}$, $\{bw_i = 6, bw_f = 19\}$

(b) c2:$\{bw_e = 6, bw_m = 13\}$, $\{bw_i = 7, bw_f = 17\}$

(c) c3:$\{bw_e = 6, bw_m = 13\}$, $\{bw_i = 8, bw_f = 17\}$

(d) c4:$\{bw_e = 6, bw_m = 14\}$, $\{bw_i = 7, bw_f = 17\}$

(e) c5:$\{bw_e = 6, bw_m = 14\}$, $\{bw_i = 6, bw_f = 16\}$

(f) c6:$\{bw_e = 6, bw_m = 13\}$, $\{bw_i = 6, bw_f = 17\}$

(g) c7:$\{bw_e = 6, bw_m = 16\}$, $\{bw_i = 8, bw_f = 20\}$

(h) c8:$\{bw_e = 6, bw_m = 14\}$, $\{bw_i = 6, bw_f = 18\}$

(i) c9:$\{bw_e = 7, bw_m = 13\}$, $\{bw_i = 6, bw_f = 25\}$

Fig. 3: PID Benchmarks: the performance on each test case is shown. The bit widths are those calculated by Algorithms 1 and 2. In general, a smooth and rapid convergence to the target solution is preferred.

the bit-widths of the {exponent(bw_e),mantissa(bw_m)} and {integer(bw_i),fraction(bw_f)} are shown in Fig. 3.

The control process assumes signal sampling at 0.01 second intervals, which is also the update interval of the PID controller. By giving the normalized target reference signal as 1, the iterative control process stops at a convergence error of 10^{-3}, where all benchmarks are convergent after at least 600 iterations.

As shown, for floating-point, the estimated exponent bw_e is 6 bits for all cases of c1-c8 except c9 with 7 bits, while the estimated mantissa bw_m varies between 13 and 16 bits. The estimated integer bw_i varies between 6 to 8 bits, while the estimated fraction bw_f varies between 16 to 25 bits.

The larger the number of iterations, the more uncertain is the computational approximation, which tends to require larger bit-width to maintain accuracy. For example, c7 has the largest bw_m and bw_f bit-width, 16 and 20 bits respectively, given the largest number of iterations for all the benchmarks. Similarly, c4, and c8 have more iterations, and larger bw_m and bw_f, 14 and 17-18 bits relatively. Another observation from the AA based control flow is: the larger the variation of the parameters and the smaller the intervals, the larger the bit-width of fraction bw_f tends to be, such as for c1, c4 and c9. Due to the non-linear representation of the floating-point representation, the mantissa bw_m varies less than bw_f for all the benchmarks.

Generally, by considering the relative error of approximated control flow based on AA, our approach can achieve automated

Back to Contents

precision estimation in terms of the bit-width with iterative computational uncertainty. The automated accelerator generation is then implemented based on the estimated precision.

C. Hardware Accelerators

Table II shows the resource utilization and dynamic power consumption of automatically implemented PID controllers for the baseline single floating-point $fp32$ as well as custom floating-point fp, fixed-point fxp with corresponding precision in Section IV-B. Typical usage of logic elements on the FPGA, such as Look-Up-Tables (LUT), Registers (Reg.), and DSP blocks (DSP48e) are recorded, as well as the system clock frequency in MHz and the power consumption in mW.

TABLE II: PID Accelerator Cost and Performance

Case	LUT			Reg.			DSP48e			Freq. (MHz)			Power (mW)		
	fp32	fpx	fxp	fp32	fpx	fxp	fp32	fpx	fxp	fp32	fpx	fxp	fp32	fpx	fxp
c1	1624	868	508	2782	1557	729	10	4	10	554	455	430	291	236	225
c2	1746	939	598	2942	1650	745	10	4	12	497	481	432	285	241	228
c3	1938	1048	704	3304	1979	914	16	8	18	506	484	415	305	248	238
c4	2456	1582	1001	3982	2496	1240	15	6	27	471	470	409	314	265	256
c5	1558	839	1232	2651	1437	835	8	4	6	489	500	359	273	238	223
c6	1969	1071	664	3304	1979	746	16	8	14	462	473	431	295	253	232
c7	2072	1404	1175	3401	2355	1212	16	8	37	482	448	386	303	260	266
c8	1771	1058	628	2975	1725	793	10	4	15	442	462	448	276	242	235
c9	1704	988	965	2911	1691	1035	10	4	24	552	471	393	294	241	247

With automatically estimated precision, the resource cost and the power consumption are both significantly reduced. On average, compared to the baseline, about 46% of LUT, 40% of Registers, and 55% of DSP are saved across the estimated floating-point cases. On average, about 62% of LUT and 73% of Registers are saved for the estimated fixed-point cases, but about 63% more DSP blocks are used. This might be due to the nature of the fixed-point Accumulated Logic Unit (ALU) for DSP blocks on the Xilinx FPGA, where the tool-set tends to use more DSPs to accelerate fixed-point computations. Besides the resource cost, the clock frequencies of all the implemented PID controllers are slightly reduced, on average by about 15%, and the power consumption is reduced by about 27%. However, we stress that the design is not yet fully optimized for hardware performance.

V. Conclusions

We have presented an automated framework for optimal precision estimation for both floating-point and fixed-point binary formats, leading to custom accelerator generation on an Xilinx FPGA platform. Through experiment, we demonstrate the completeness of estimated precision for a one-dimensional iterative application, a PID controller, with parameter interval analysis through AA computation and constrained precision decision based on the relative error of parameter intervals. The implemented approximate PID controller on the FPGA leads to significant reductions in both resource cost and power consumption, by as much as 62% and 27% respectively, compared to the usual, baseline single floating-point implementation, while there is a minor speed reduction in clock frequency. Further exhaustive hardware design optimization would improve the final performance results. Our work is potentially applicable to multiple dimension iterative applications, such as multi-variable PID controller and iterative solvers of convex optimization, with more complex computational uncertainty.

References

[1] H.-s. Suh, J. Meng, T. Nguyen, and et al., "Algorithm-hardware co-optimization for energy-efficient drone detection on resource-constrained fpga," in *2021 International Conference on Field-Programmable Technology (ICFPT)*, 2021, pp. 1–9.

[2] G. Constantinides, P. Cheung, and W. Luk, "Wordlength optimization for linear digital signal processing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 10, pp. 1432–1442, 2003.

[3] S. Roy and P. Banerjee, "An algorithm for trading off quantization error with hardware resources for matlab-based fpga design," *IEEE Transactions on Computers*, vol. 54, no. 7, pp. 886–896, 2005.

[4] D.-U. Lee, A. Gaffar, O. Mencer, and W. Luk, "Minibit: bit-width optimization via affine arithmetic," in *Proceedings. 42nd Design Automation Conference, 2005.*, 2005, pp. 837–840.

[5] W. Osborne, R. Cheung, J. Coutinho, W. Luk, and O. Mencer, "Automatic accuracy-guaranteed bit-width optimization for fixed and floating-point systems," in *2007 International Conference on Field Programmable Logic and Applications*, 2007, pp. 617–620.

[6] J. Stolfi and L. de Figueiredo, "An introduction to affine arithmetic," 2003.

[7] J. Cong, K. Gururaj, B. Liu, C. Liu, Z. Zhang, S. Zhou, and Y. Zou, "Evaluation of static analysis techniques for fixed-point precision optimization," in *2009 17th IEEE Symposium on Field Programmable Custom Computing Machines*, 2009, pp. 231–234.

[8] W. Farag, "Complex trajectory tracking using pid control for autonomous driving," *International Journal of Intelligent Transportation Systems Research*, vol. 18, no. 2, pp. 356–366, 2020.

[9] J. Lima, R. Menotti, J. M. P. Cardoso, and E. Marques, "A methodology to design fpga-based pid controllers," in *2006 IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, 2006, pp. 2577–2583.

[10] M. Kocur, S. Kozak, and B. Dvorscak, "Design and implementation of fpga - digital based pid controller," in *Proceedings of the 2014 15th International Carpathian Control Conference (ICCC)*, 2014, pp. 233–236.

[11] J. Wang, M. Li, W. Jiang, Y. Huang, and R. Lin, "A design of fpga-based neural network pid controller for motion control system," *Sensors*, vol. 22, no. 3, p. 889, Jan 2022. [Online]. Available: http://dx.doi.org/10.3390/s22030889

[12] K. Åström and T. Hägglund, "Benchmark systems for pid control," *IFAC Proceedings Volumes*, vol. 33, no. 4, pp. 165–166, 2000, iFAC Workshop on Digital Control: Past, Present and Future of PID Control, Terrassa, Spain, 5-7 April 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667017382381

[13] J. Stolfi, *libaa*, accessed on March 1, 2022. [Online]. Available: http://www.ic.unicamp.br/ stolfi/

[14] O. Gay, D. Coeurjolly, and N. J. Hurst, *libaffa*, accessed on March 1, 2022. [Online]. Available: http://www.nongnu.org/libaffa/

[15] S. Kiel, "Yalaa: Yet another library for affine arithmetic," *Reliab. Comput.*, vol. 16, pp. 114–129, 2012.

[16] J. Jääger, *hafar*, accessed on March 1, 2022. [Online]. Available: https://hackage.haskell.org/package/hafar

[17] Y.-S. Kung, H. Than, and T.-Y. Chuang, "Fpga-realization of a self-tuning pid controller for x–y table with rbf neural network identification," *Microsystem Technologies*, vol. 24, no. 1, pp. 243–253, Jan 2018. [Online]. Available: https://doi.org/10.1007/s00542-016-3248-x

Efficient joint surface detection and depth estimation of single-photon Lidar data using assumed density filtering

K. Drummond$^{(1,2)}$, D. Yao$^{(1)}$, A. Pawlikowska$^{(2)}$, R. Lamb$^{(2)}$, S. McLaughlin$^{(1)}$, Y. Altmann$^{(1)}$
$^{(1)}$School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, UK
$^{(2)}$Leonardo UK, Edinburgh, UK
kd122@hw.ac.uk

***Abstract*—This paper addresses the problem of efficient single-photon Lidar (SPL) data processing for fast 3D scene reconstruction. Traditional methods for 3D ranging from Lidar data construct a histogram of the time of arrival (ToA) values of photon detection events to obtain final depth estimates for a desired target. However processing large histogram data volumes over long temporal sequences results in undesirable costs in memory requirement and computational time. By adopting a Bayesian formalism, we combine the online estimation strategy of Assumed Density Filtering (ADF) with joint surface detection and depth estimation methods to eventually process SPL data on-chip without the need for histogram data construction. We also illustrate how the data processing efficiency can be increased by reducing the set of unknown discrete variables based on posterior distribution estimates after each detection event, reducing computational cost for future detection events. The benefits of the proposed methods are illustrated using synthetic and real SPL data for targets at up to 3 km**

***Index Terms*—Single-photon Lidar, Bayesian estimation, Detection, Ensemble estimation, Assumed Density Filtering.**

I. Introduction

Single-Photon Lidar (SPL) is a promising technology that has found many applications in different fields such as autonomous vehicles [1], agriculture [2] and defence [3]. Three-dimensional (3D) scene reconstruction using SPL data benefits from the key advantages, including the use of low-power, eye-safe laser sources [3], picosecond timing resolution allowing greater surface-to-surface resolution at ranges up to 200 km [4] or imaging in extreme conditions, such as fog/smoke [5] or underwater [6], [7].

SPL systems are based on time-correlated single-photon counting (TCSPC) [8]–[10] methods, which record a time of flight (ToF) value for a each detected photon corresponding to the time between the emission of light pulses from the Lidar laser source and the detection event occurring at the receiver. A detection event associated with a desired target occurs when a photon travels from the laser source to the target and is reflected off that target back to the receiver. A depth estimate of the target can then be calculated from this recorded ToF. However, this depth estimate can be adversely affected by background detection events, which result from ambient illumination and dark counts. The depth estimate is improved by repeating the pulse emission and photon detection process to acquire a sufficient number of target detection events relative to the number of background detection events.

Advances in single-photon avalanche diode (SPAD) array technology, resulting in the acquisition of data at video rates or higher [11], [12], have put greater interest on processing the data at real time speeds to obtain reliable 3D reconstructions of target scenes. However, traditional 3D scene reconstruction methods [13]–[17] can suffer from a computational bottleneck due to their reliance on the construction of ToF histograms. These methods require histograms to be built pixel-by-pixel before the estimation and then process the large data volume over long temporal sequences, leading to large memory costs and computation times.

In a previous histogram-based method [18], the processing speed was improved by collectively detecting objects/surfaces and treating some unknown model parameters as discrete rather than continuous random variables. This method also provided conservative posterior uncertainties. Alternative methods have recently improved the efficiency of 3D scene reconstruction using on-chip online processing methods. A more recent algorithm [19] compresses the SPL data by using sketches that can be computed in an online fashion. However, this online method suffers with having to determine the surface detection and depth estimation sequentially, and only provides the point estimates of unknown parameters, without uncertainty quantification. Another method [20] adopts an approximate Bayesian estimation strategy based on Assumed Density Filtering (ADF) [21] to find the approximating posterior distribution of the depth of moving surfaces. The mean and variance of the approximating distribution directly provide the point estimate of the depth and its uncertainty quantification.

This paper combines the online estimation strategy of ADF as described in [20] and our previous work [18] on depth estimation using ensemble estimators, to propose a novel, pixel-wise, online processing method for joint surface and depth estimation from single-photon Lidar data using ensemble estimators. We employ ADF for online processing, overcoming the need to build ToF histograms. The method is compatible with on-chip processing. We look to reduce computational costs when processing future detection events by using the posterior distribution generated after each detection event to help reduce the number of discrete unknown parameter values.

The remainder of this paper is organized as follows. Section II recalls the statistical observation model used for SPL and describes the proposed method for ensemble estimation with

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

discrete parameter reduction. Results of simulations conducted with synthetic single-pixel SPL data and real data are presented and discussed in Section IV. Conclusions are finally reported in Section V.

II. Bayesian model and related work

A. Bayesian model

In this paper, for a 3D scene observed by the SPAD detector, we consider a sequence of N binary frames with duration T, such that at most a single detection event is recorded per binary frame in each pixel. Since our method processes pixels independently, we derive all the equations for a single pixel, omitting pixel indices. The observations consist of a set of $K \leq N$ photon time of arrival (ToA) values in $\boldsymbol{y} = \{y_k\}_{k=1}^K$ associated with each detection event, such that $y_k \in (0, T_r)$ where the repetition period of the laser source $T_r = T$ [22]. Given the unknown target depth d, the probability density function for a photon ToA, y_k, for a given pixel is given by

$$f(y_k|d, w) = w \; h_0\left(y_k - \frac{2d}{c}\right) + (1-w)\; \mathcal{U}_{(0,T_r)}, \qquad (1)$$

such that c is the speed of light. The variable w represents the probability of the detection event being associated with a photon returning from the target surface, while $(1-w)$ is the probability that the detection event is a noisy detection event. The function $h_0(\cdot)$ is the normalised Impulse Response Function (IRF) of the Lidar system which can be approximated by a Gaussian profile $\mathcal{N}(2d/c, s^2)$, and $\mathcal{U}_{(0,T_r)}$ defined on $(0, T_r)$ represents a distribution of noisy detection events. This can be a non-uniform distribution, for example as a result of pile up effects from high ambient illumination conditions or imaging in high scattering media. However for simplicity we assume this to be uniform in this work.

Assuming the dead-time of the SPAD detector is negligible, the joint likelihood of the K detection events can be given as

$$f(\boldsymbol{y}|d, w) = \prod_{k=1}^{K} f(y_k|d, w). \qquad (2)$$

Suppose for now that the target detection probability w is known, and that the depth parameter is assigned the following Gaussian prior distribution

$$f(d) = \mathcal{N}(\mu_0, \sigma_0^2). \qquad (3)$$

Using $f(d)$ and the joint likelihood in (2), the posterior distribution of d is given by

$$f(d|\boldsymbol{y}, w) = \frac{\prod_{k=1}^{K} f(y_k|d, w) f(d)}{\int \prod_{k=1}^{K} f(y_k|d, w) f(d) \mathrm{d}d}. \qquad (4)$$

The posterior mean and variance of d can be computed analytically (although often at a significant cost in a real-time context), assuming that w is known [18]. However, w is unknown in practice and needs to be estimated as its value can have a dramatic impact on the quality of the depth estimate.

B. Previous/related work

When w is unknown, a prior distribution can be assigned to w. The classical approach considers w as continuous and the posterior distribution of depth can be derived from the marginal posterior distribution

$$f(d|\boldsymbol{y}) = \int f(d|\boldsymbol{y}, w) f(w|\boldsymbol{y}) \mathrm{d}w, \qquad (5)$$

which requires computing the integral over w. To overcome this difficulty, as in [18], w is assumed to be discrete instead and can take a user-defined finite number, M, of values from $\{w_1, w_2, \ldots, w_M\}$, whose prior $f(w_m)$ $(m = 1, \ldots, M)$ follows a uniform distribution. Based on this discretization, $f(d|\boldsymbol{y})$ in (5) becomes tractable by computing

$$f(d|\boldsymbol{y}) = \sum_{m=1}^{M} f(d|\boldsymbol{y}, w_m) f(w_m|\boldsymbol{y}) \qquad (6)$$

where $f(w_m|\boldsymbol{y})$ is computed via

$$f(w_m|\boldsymbol{y}) = \frac{f(\boldsymbol{y}|w_m) f(w_m)}{\sum_{m=1}^{M} f(\boldsymbol{y}|w_m) f(w_m)}, \qquad (7)$$

where $f(\boldsymbol{y}|w_m)$ is the denominator of (4) with $w = w_m$. This depth estimation method provides satisfactory results and relies on an ensemble estimator as final depth estimate. However, in (6), the mean and variance of the mth base estimator $f(d|\boldsymbol{y}, w_m)$ and the weight $f(w_m|\boldsymbol{y})$ are computed by using the whole set $\boldsymbol{y}$ at once, which prevents its application for real-time depth estimation. In this paper, a new depth estimation method using ensemble estimators is proposed to estimate $f(d|\boldsymbol{y}, w_m)$ and $f(w_m|\boldsymbol{y})$ in (6) online by Assumed Density Filtering without the requirement of ToF histograms, as will be presented next. In contrast to [18], here d is assumed to be continuous, which simplifies computations.

III. Online depth estimation using Assumed Density Filtering

A. Online estimation of depth posterior mean and variance

Instead of computing $f(d|\boldsymbol{y}, w)$ after having all the observations $\{y_k\}_{k=1}^K$, an online depth estimation strategy using ADF is proposed to obtain a posterior approximation $q(d)$ such that

$$q(d) \approx f(d|\boldsymbol{y}, w), \qquad (8)$$

where $q(d) \propto \mathcal{N}(\mu_d, \sigma_d^2)$ is a normalized Gaussian distribution. In ADF, the approximated posterior mean μ_d and variance σ_d^2 are found sequentially, after each detection, by minimizing the following Kullback-Leibler (KL) divergences

$$q^{(k)}(d) = \underset{q^{(k)}(d)}{\arg\min}\; KL(\hat{p}^{(k)}(d|\boldsymbol{y}_k, w) || q^{(k)}(d)), \qquad (9)$$

where $\boldsymbol{y}_k = \{y_i\}_{i=1,..k}$, $q^{(k)}(d) \propto \mathcal{N}(\mu_d^{(k)}, (\sigma_d^{(k)})^2)$ and

$$\hat{p}^{(k)}(d|\boldsymbol{y}_k, w) = \frac{f(y_k|d, w) q^{(k-1)}(d)}{\int f(y_k|d, w) q^{(k-1)}(d) \mathrm{d}d}, \qquad (10)$$

which is a tilted distribution consisting of the product of likelihood function for the kth observation and posterior

Back to Contents

approximation $q^{(k-1)}(d)$ that has been computed from the first $(k-1)$ observations. The solution to (9) is given by

$$\mu_d^{(k)} = \mathbb{E}_{\hat{p}^{(k)}}[d],\ (\sigma_d^{(k)})^2 = \mathbb{E}_{\hat{p}^{(k)}}[d^2] - (\mathbb{E}_{\hat{p}^{(k)}}[d])^2. \qquad (11)$$

When $q^{(k)}(d)$ is updated at $k = K$, $q^{(K)}(d)$ provides the final posterior approximation of the exact posterior distribution $f(d|\boldsymbol{y})$ in (4), i.e., $q^{(K)}(d) \approx f(d|\boldsymbol{y}, w)$.

B. Online estimation of the model evidence

Using the procedure presented in the previous section, the detection events y_k are processed online from $k = 1$ to $k = K$ in ADF and the approximating posterior distribution of d is updated sequentially by only propagating the mean and variance of $q^{(k-1)}(d)$ from previous frames.

In (10), incorporating each term $f(y_k|d, w)$ produces a normalizing constant $\int f(y_k|d, w)q^{(k-1)}(d)\mathrm{d}d$ that can be used to approximate the corresponding term of model evidence in (4) associated with $\boldsymbol{y}_k$, i.e.,

$$\begin{aligned} s^{(k)}(w) &:= s^{(k-1)}(w) \int f(y_k|d, w)q^{(k-1)}(d)\mathrm{d}d \\ &\approx \int \prod_{i=1}^{k} f(y_i|d, w) f(d)\mathrm{d}d, \end{aligned} \qquad (12)$$

where $s^{(k-1)}(w)$ is the approximation of the model evidence after having observed $\boldsymbol{y}_k$ and $s^{(0)} = 1$. When $s^{(k)}$ is updated at $k = K$, $s(w) = s^{(K)}(w)$ provides the final approximation of the exact model evidence $f(\boldsymbol{y}|w)$ in Eq. (4), i.e.,

$$s(w) \approx f(\boldsymbol{y}|w). \qquad (13)$$

These results from the ADF method can then be applied to the method proposed in Drummond et al. [18] for depth inference with unknown w. For the discrete w parameter, $w \in \{w_1, w_2, ..., w_M\}$, where we allow $w_1 = 0$ to be in the admissible set of w and M is a user-defined parameter, we obtain $\boldsymbol{s}(w) \in \{s(w_1), \ldots, s(w_M)\}$. Finally, the marginal posterior $f(w_m|\boldsymbol{y})$ can be computed by

$$f(w_m|\boldsymbol{y}) = \frac{\boldsymbol{s}(w_m) f(w_m)}{\sum_{m=1}^{M} \boldsymbol{s}(w_m) f(w_m)}. \qquad (14)$$

C. Final depth estimation using ensemble estimators

As described in [18], the final depth and variance is computed using

$$\begin{cases} \bar{\mu} = \sum_{m=1}^{M} f(w_m|\boldsymbol{y})\mu_m, \\ \bar{\sigma}^2 = \left(\sum_{m=1}^{M} f(w_m|\boldsymbol{y})(\sigma_m^2 + \mu_m^2)\right) - \bar{\mu}^2. \end{cases} \qquad (15)$$

where $\{\mu_m = \mu(w_m)\}_{m=1}^{M}$ and $\{\sigma_m^2 = \sigma^2(w_m)\}_{m=1}^{M}$ are the sets of means and variances of $f(d|\boldsymbol{y}, w)$ obtained by ADF.

D. Reduction of discrete w parameter list

As stated in the previous subsection, the normalizing constant estimate $\boldsymbol{s}^{(k)}(w)$, and consequently the marginal posterior $f(w|\boldsymbol{y}_k)$, is updated after each detection event for all $w \in \{w_1, w_2, ..., w_M\}$. As the values of $\boldsymbol{s}^{(k)}(w)$ are can be computed independently for all w values, the posterior probabilities for w can be easily calculated and used after each detection event. After a number of detection events have occurred and a probability distribution of w has been computed on a pre-defined grid, a decision can then be made as to which w values can be eliminated form the set, based on their corresponding $f(w|\boldsymbol{y})$ value.

	Depth		w	Time
	$\bar{\mu}(m)$	$\bar{\sigma^2}(m^2)$		(s)
ADF basic $(M = 20)$	17.99 (7.67e-3)	5.18e-5 (4.71e-6)	0.22 (0.02)	0.034
ADF basic $(M = 100)$	17.99 (7.20e-3)	5.18e-5 (4.96e-6)	0.22 (0.02)	0.133
ADF * warm-start	17.99 (7.20e-3)	5.18e-5 (7.83e-3)	0.20 (0.01)	0.194
Reduction mthd. 1 **	17.99 (7.67e-3)	5.18e-5 (7.70e-3)	0.20 (0.01)	0.141
Reduction mthd. 2 **	**17.99 (7.20e-3)**	**5.18e-5 (7.87e-3)**	**0.20 (0.01)**	**0.100**
Drummond [18] $(M = 20)$	17.99 (6.24e-3)	4.03e-5 (4.60e-5)	0.20 (0.01)	0.012
Drummond [18] $(M = 100)$	17.99 (6.24e-3)	4.03e-5 (4.03e-5)	0.20 (0.01)	0.059
cross correlation	17.99 (7.67e-3)	N/A (no std.)	0.2 (w known)	0.001

TABLE I: Comparison of the different depth and w estimates for different methods. Values in brackets represent standard deviations over 1000 results. The actual value of (d, w) is $(17.99m, 0.2)$.$^{(*)}$: this method uses $M = 100$. $^{(**)}$: these methods start with $M = 100$ and warm-start.

.

After the subset $W' \subseteq \{w_1, ..., w_M\}$ has been determined, the corresponding $f(w'|\boldsymbol{y})$ values are re-normalised, such that $w' \in W'$. The future detection events are used to upgrade the prior distributions corresponding to the remaining w values to be used for the final output estimations. All the intermediate variables corresponding to discarded w values are also discarded, thereby reducing computational cost for future detection events. In the next section, we will investigate two methods to prune the w grid.

IV. RESULTS

We first evaluate the performance of the proposed algorithm using synthetic single-pixel data and then using the real SPL college tower dataset, provided by Leonardo UK [3].

A. Single-Pixel analysis

We first generate synthetic detection event data, where $K = 1000$ detection events are randomly generated from a normal distribution with mean $d = 750$ bins $= 17.99m$. The spatial bin length is $0.024m$, and variance $s^2 = 50$ bins$^2 = 0.029m^2$, for $T = 2.4 \times 10^{-7}s$. This mimics a single-photon Lidar system whose depth resolution (due to time binning) is 2.4cm, such that T corresponds to a depth range of around $36m$ (the minimum depth being set to 0 without loss of generality). We set the ground truth signal photon probability to $w = 0.2$. Due to the signal randomness when generating the data, each investigation was repeated 1000 times and we present the means and standard deviations obtained over the 1000 repetitions.

The first two rows of Table I show the results for an initial, basic method that applies ADF to the method proposed in

Back to Contents

Drummond et al. [18] for $M = 20$ and $M = 100$. We use $\mu_d^{(0)} = 0$ and $\sigma_d^{(0)} = 1 \times 10^6$ for the ADF initialised parameters.

The first 50 detection events were then used to "warm-start" the ADF method, where Drummond et al. [18] is used to obtain way-point estimates $\mu_d^{(50)}$ and $\sigma_d^{(50)}$. These are used as new initial parameters $\mu_d'^{(0)}$ and $\sigma_d'^{(0)}$ to be applied for the remaining detection events to obtain our final results. The third row of Table I shows how the "warm-start" method produces just as accurate depth estimates as previously with improved w estimates at the same processing times.

While there are many different ways to eliminate different w values, in this paper we investigate two specific ways on reducing the set of $w \in \{w_1, ..., w_M\}$, where $M = 100$. For the first reduction method, we reduce the w parameter list to a set amount after a defined number of observations. We define the set $R = \{80, 60, 40, 20\}$ as the set of values which define the number of w values to retain based which values correspond to the R_i largest $\boldsymbol{s}(w)$ values. We also define the set $E = \{200, 400, 600, 800\}$ as the number of observations after which the reduction of the w parameter set takes place, i.e. the set of w parameter values is reduced to R_i values at detection event E_i. For the second reduction method, we remove values of w whose posterior probabilities fall below a pre-defined threshold $\gamma = 1 \times 10^{-4}$. This method is initiated after the first 100 detection events after the warm-up. The fourth and fifth rows of Table I show that with pruning the w-grid, we can get accurate results quicker. As the second reduction method was quickest, this method was chosen for future investigations for the remainder of this paper.

The three bottom rows show the results for the method proposed in Drummond et al. [18], with $M = 20$ and $M = 100$ and a typical cross correlation method. For the cross correlation, the w value is assumed to be known ($w = 0.2$), which is over optimistic as it is often unknown in practice. The times presented are the times for processing the histogram data and do not include histogram construction times.

Fig. 1 depicts μ and σ^2 for the "warm-started" method using ADF, both with and without w set reduction. The top and second rows of Fig. 1 illustrate how the estimated depth variance decreases as K increases (true w set to 0.2). All plots are restricted to $K > 200$ for visualisation purposes. It is worth noting that here with $K = 200$, the mean depth results are already accurate. The third and bottom rows illustrate how the estimated depth variance decreases as the signal-to-background ratio (SBR), w, used to generate the data increases (K set to 1000). These plots are restricted to $w > 0.1$ below which the means degrade and deviate from the ground truth drastically in this scenario.

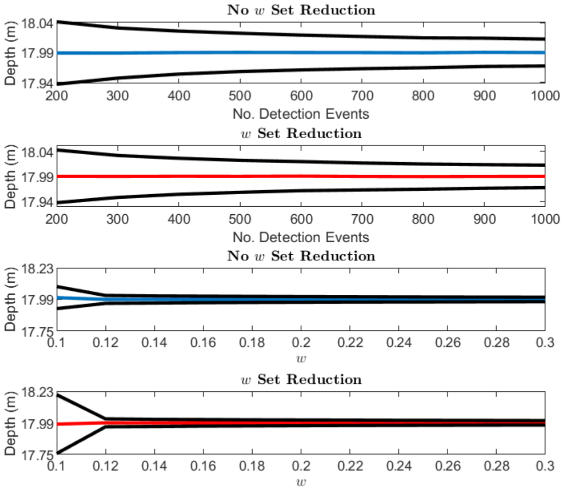


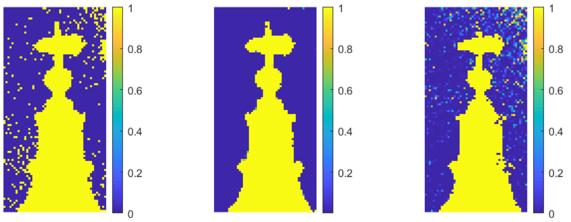
Fig. 1: Graphic results of depth estimates with (red) and without (blue) w set reduction with error range plots (black) for a different number of detection events without (top) and (second), and for synthetic data with different SBR, w, values (third) and (bottom). The actual value of d is $17.99m$.

B. Real SPL College Tower data analysis

We now consider the college tower SPL histogram dataset acquired by Leonardo to illustrate the potential benefits of the proposed method. The SPL cube consists of 100×50 pixels and 5626 time bins, corresponding to an Edinburgh college tower, taken at $\approx$ 3km range, originally considered in [3]. This dataset has $SBR \approx 0.22$. For the purposes of this investigation, we gate this dataset and use only bins between bins 2001 and 4000, reducing the number of noisy detection events and increasing the average SBR to $SBR \approx 0.41$.

This gated data is then run using the first 400 detection events for the warm-starting process to obtain new initialised parameters for the ADF process, as described previously. The remaining detection event data was then used for the ADF process to obtain the final results.

We compare our estimated probability of target presence map with Sheehan et al. [19] and Drummmond et al. [18] displayed in Fig. 2, showing the proposed method can lead to better results than using Sheehan's method [19] for this scene. The admissible grid of w is set using $M = 100$ equality spaced $\{w_m\}_m$. A target is assumed to be present in each pixel if and only if $f(w > w_0|\boldsymbol{y}) = 0.9999$.



(a) Sheehan [19]

(b) Drummond $w_0 = 0.02$ [18]

(c) Proposed: $w_0 = 0.02$

Fig. 2: College tower data comparison of probability of detection results for the methods by Sheehan et al. [19] (a), and Drummond et al. [18] (b) and for the proposed method (c), both using $w_0 = 0.02$.

Back to Contents

The estimates of w are used to estimate the target intensity (number of signal photons $I = wK$) and the number of background photons $B = (1 - w)K$. The mean of $f(w|y)$, $\bar{w}$ can obtain the final reflectivity estimates $\bar{I}$ and background estimates $\bar{B}$. These results, along with the final mean depth estimates $\bar{\mu}$ are presented in Fig. 3, where we use $w_0 = 0.02$.

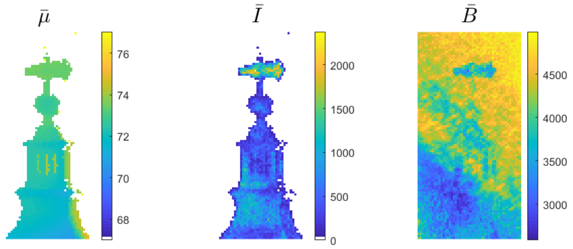


Fig. 3: Final mean depth (left), reflectivity (middle) and background (right) estimates for the tower data using the proposed method, for $w_0 = 0.02$.

V. Conclusion

In this paper, we proposed an extension to the ensemble estimator method and produced satisfactory results using ADF to obtain the posterior distribution profile for the final surface detection, depth estimation and uncertainty quantification estimates. Furthermore, we were able to further improve efficiency by eliminating values from the discrete variables depending on their corresponding posterior distribution profile value. We note if the SBR is relatively high, the online method performs as well as the batch-based method. However, having a long enough warm-up batch is crucial if the SBR is low, and in extremely low SBR regimes it is better to either use larger initial batches with better known prior distributions or to not to perform online estimation. Further investigation would be needed to define the best warm-up period (tradeoff between complexity and performance) as a function of the SBR. In the future we aim to propose a GPU implementation to enable reliable depth estimation and uncertainty quantification at real-time speeds. Furthermore, we plan to adapt the framework to richer approximations of the posterior distributions of d, allowing for multiple surface detections per pixel.

Acknowledgements

This work was supported by CENSIS and Leonardo UK. This work was supported by the Royal Academy of Engineering under the Research Fellowship scheme RF201617/16/31, the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing.

References

[1] J. Rapp, J. Tachella, Y. Altmann, S. McLaughlin, and V. K. Goyal, “Advances in single-photon Lidar for autonomous vehicles,” *IEEE Signal Processing Magazine*, vol. 37, no. 4, 2020.

[2] P. Yimyam and A. F. Clark, “3D reconstruction and feature extraction for agricultural produce grading,” *8th International Conference on Knowledge and Smart Technology (KST), Chiangmai*, no. 136-141, 2016.

[3] A. M. Pawlikowska, A. Halimi, R. A. Lamb, and G. S. Buller, “Single-photon three-dimensional imaging at up to 10 kilometers range,” *Opt. Express*, vol. 25, no. 11919–11931, 2017.

[4] Z.-P. Li, J.-T. Ye, X. Huang, P.-Y. Jiang, Y. Cao, Y. Hong, C. Yu, J. Zhang, Q. Zhang, C.-Z. Peng, F. Xu, and J.-W. Pan, “Single-photon imaging over 200km,” *Optica*, vol. 8, no. 3, pp. 344–349, Mar 2021. [Online]. Available: http://www.osapublishing.org/optica/abstract.cfm?URI=optica-8-3-344

[5] C. Fu, H. Zheng, G. Wang, Y. Zhou, H. Chen, Y. He, J. Liu, J. Sun, and Z. Xu, “Three-dimensional imaging via time-correlated single-photon counting,” *Applied Sciences*, vol. 10, no. 6, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/6/1930

[6] A. Maccarone, A. McCarthy, X. Ren, R. E. Warburton, A. M. Wallace, J. Moffat, Y. Petillot, and G. S. Buller, “Underwater depth imaging using time-correlated single-photon counting,” *Opt. Express*, vol. 23, no. 26, pp. 33 911–33 926, Dec 2015. [Online]. Available: http://opg.optica.org/oe/abstract.cfm?URI=oe-23-26-33911

[7] A. Halimi, A. Maccarone, A. McCarthy, S. McLaughlin, and G. S. Buller, “Object depth profile and reflectivity restoration from sparse single-photon data acquired in underwater environments,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 3, 2017.

[8] G. Buller and R. Collins, “Single-photon generation and detection,” *Measurement Science and Technology*, vol. 21, no. 1, 2010.

[9] A. McCarthy, X. Ren, A. D. Frera, N. R. Gemmell, N. J. Krichel, C. Scarcella, A. Ruggeri, A. Tosi, and G. S. Buller, “Kilometer-range depth imaging at 1550 nm wavelength using an ingaas/inp single-photon avalanche diode detector,” *Opt. Express*, vol. 21, no. 19, pp. 22 098–22 113, Sep 2013. [Online]. Available: http://www.opticsexpress.org/abstract.cfm?URI=oe-21-19-22098

[10] W. Becker, *Advanced Time-Correlated Single-Photon Counting Techniques*, ser. Springer Series in Chemical Physics. Springer, 2005.

[11] M. Entwistle, M. A. Itzler, J. Chen, M. Owens, K. Patel, X. Jiang, K. Slomkowski, and S. Rangwala, “Geiger-mode APD camera system for single-photon 3D LADAR imaging,” in *Advanced Photon Counting Techniques VI*, M. A. Itzler, Ed., vol. 8375, June 2012, p. 83750D.

[12] R. Henderson, N. Johnston, H. Chen, D. Li, G. Hungerford, R. Hirsch, P. Yip, D. McLoskey, and D. Birch, “A 192 x 128 time correlated single photon counting imager in 40nm CMOS technology,” in *44th European Solid-State Circuits Conference (ESSCIRC) 2018*. IEEE, Sept. 2018.

[13] J. Tachella, Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, S. McLaughlin, and J.-Y. Tourneret, “Bayesian 3d reconstruction of complex scenes from single-photon lidar data,” *SIAM Journal on Imaging Sciences*, vol. 12, no. 1, pp. 521–550, March 2019.

[14] D. Shin, A. Kirmani, V. K. Goyal, and J. H. Shapiro, “Photon-efficient computational 3D and reflectivity imaging with single-photon detectors,” *IEEE Trans. Comput. Imaging*, vol. 1, no. 2, pp. 112–125, Jun 2015.

[15] Y. Altmann and S. McLaughlin, “Range estimation from single-photon Lidar data using a stochastic EM approach,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, ser. European Signal Processing Conference (EUSIPCO). United States: IEEE, Dec. 2018, pp. 1112–1116.

[16] J. Rapp and V. K. Goyal, “A few photons among many: Unmixing signal and noise for photon-efficient active imaging,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 3, pp. 445–459, 2017.

[17] J. Rapp, Y. Ma, R. M. Dawson, and V. K. Goyal, “Dead time compensation for high-flux ranging,” *IEEE Trans. Signal Process.*, vol. 67, no. 13, pp. 3471–3486, 2019.

[18] K. Drummond, S. McLaughlin, Y. Altmann, A. Pawlikowska, and R. Lamb, “Joint surface detection and depth estimation from single-photon lidar data using ensemble estimators,” in *2021 Sensor Signal Processing for Defence Conference (SSPD)*, 2021, pp. 1–5.

[19] M. P. Sheehan, J. Tachella, and M. E. Davies, “A sketching framework for reduced data transfer in photon counting lidar,” *IEEE Transactions on Computational Imaging*, vol. 7, pp. 989–1004, 2021.

[20] Y. Altmann, S. McLaughlin, and M. Davies, “Fast online 3d reconstruction of dynamic scenes from individual single-photon detection events,” *IEEE Transactions on Image Processing*, vol. 29, nov 2019.

[21] T. P. Minka, “Expectation propagation for approximate bayesian inference,” in *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, ser. UAI’01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, p. 362–369.

[22] Q. Legros, S. McLaughlin, Y. Altmann, S. Meignen, and M. E. Davies, “Robust depth imaging in adverse scenarios using single-photon Lidar and beta-divergences,” in *2020 Sensor Signal Processing for Defence Conference (SSPD)*, 2020, pp. 1–5.

An extension to the Frenet-Serret and Bishop invariant extended Kalman filters for tracking accelerating targets

Joe Gibbs
James Watt School of Engineering
University of Glasgow
Glasgow, UK
j.gibbs.1@research.gla.ac.uk

David Anderson
James Watt School of Engineering
University of Glasgow
Glasgow, UK
dave.anderson@glasgow.ac.uk

Matt MacDonald
Sightline Control Systems group
Leonardo UK
Edinburgh, UK
matt.macdonald@leonardo.com

John Russell
Sightline Control Systems group
Leonardo UK
Edinburgh, UK
john.russell@leonardo.com

***Abstract*—This paper presents an extension to the original Frenet-Serret and Bishop frame target models used in the invariant extended Kalman filter (IEKF) to account for tangential accelerations for highly-manoeuvrable targets. State error propagation matrices are derived for both IEKFs and used to build the accelerating Frenet-Serret (FSa-LIEKF) and Bishop (Ba-LIEKF) algorithms. The filters are compared to the original Frenet-Serret and Bishop algorithms in a tracking scenario featuring a target performing a series of complex manoeuvres. The accelerating forms of the LIEKF are shown to improve velocity estimation during non-constant velocity trajectory segments at the expense of increased noise during simpler manoeuvres.**

***Index Terms*—Frenet-Serret, Bishop frame, Kalman filter, Lie groups**

I. Introduction

Target tracking is the problem of estimating rigid body motions in 3D space that a target undergoes during motion. Traditional nonlinear state estimation algorithms such as the extended (EKF), Unscented (UKF) [1] and cubature Kalman filters (CKF) [2] use models with changes in velocity or acceleration modelled as Gaussian white noise to track manoeuvring targets. Other models such as the Singer acceleration model [3] are common in industrial radar systems with [4] providing a comprehensive review. For manoeuvring targets, a bank of filters are run in a multiple model algorithm such as the interacting multiple model IMM [5] with a Kalman filter running each model before fusing the results. Simpler dynamic models incorporating the kinematics of 3D curves have been proposed to provide a more general dynamic model for target tracking. The Frenet-Serret left-invariant extended Kalman filter (FS-LIEKF) first presented in [6] estimates the pose $\chi_t \in SE(3)$ of a target along with scalar parameters describing the shape and motion of the trajectory. The Frenet-Serret formulae are used to propagate the target pose since they provide a concise means of characterising smooth curves γ, in this case the target trajectory, in 3D space ($\gamma \in \mathbb{R}^3$) through the formulae in equation (1). The Frenet-Serret equations are an elegant framework for tracking as they, by definition, describe the motion of curves. This is beneficial for tracking scenarios where the observer is attempting to reconstruct or predict a curved trajectory by propagating a set of equations. With a set of scalar Frenet parameters, a wide range of curves can be extrapolated, from simple straight segments to helices and spirals.

$$\begin{bmatrix}\dot{\mathbf{T}}\\ \dot{\mathbf{N}}\\ \dot{\mathbf{B}}\end{bmatrix} = u\begin{bmatrix}0 & \kappa & 0\\ -\kappa & 0 & \tau\\ 0 & -\tau & 0\end{bmatrix}\begin{bmatrix}\mathbf{T}\\ \mathbf{N}\\ \mathbf{B}\end{bmatrix} \tag{1}$$

Bishop showed that the Frenet-Serret frame is the not the only frame that can be readily applied to curves, extending the Frenet equations to be globally defined [7] with two signed curvatures rather than a single curvature and torsion. While the Frenet frame defines the true geometry of the space curve, with the unit normal vector N pointing towards the centre of curvature in the osculating plane, the Bishop formulae, shown in (2), enable us to initialise any starting attitude with the development equations valid for any frame. This is the case as the Bishop frame is not unique for a given curve [7].

$$\begin{bmatrix}\dot{\mathbf{T}}\\ \dot{\mathbf{M}}_1\\ \dot{\mathbf{M}}_2\end{bmatrix} = u\begin{bmatrix}0 & \kappa_1 & \kappa_2\\ -\kappa_1 & 0 & 0\\ -\kappa_2 & 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{T}\\ \mathbf{M}_1\\ \mathbf{M}_2\end{bmatrix} \tag{2}$$

The Bishop or parallel transport frame has previously been used to define tracking problems and has been implemented within an invariant extended Kalman filter for tracking a manoeuvring target with radar measurements [8], using the framework laid out by Pilte et al. [6], [9]. Both approaches are well suited to tracking problems given the ability to define complex curves using slow changing or even constant

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

parameters. While the curvature $\hat{\kappa}_t$ and torsion $\hat{\tau}_t$ parameters of the Frenet-Serret apparatus in the FS-LIEKF of [6] are able to account for the twisting motion of trajectories, tangential accelerations cannot be estimated and the filter relies upon process noise on the norm velocity $\hat{u}_t$ and unit tangent vector T to estimate the magnitude and direction. The same is true for the Bishop frame implementation or B-LIEKF, albeit with the replacement of curvature and torsion with the two Bishop curvatures $\hat{\kappa}_1$, $\hat{\kappa}_2$. This extension was originally noted by Pilte [10] with the warning that the acceleration would degrade performance on trajectories with constant velocity segments, similar to the results seen when comparing simple CV and CA EKFs.

With more modern targets able to manoeuvre with high accelerations it is critical to have a kinematic model that can adapt well to large changes in velocity. This paper presents the extension to the Frenet-Serret and Bishop IEKF algorithms to account for accelerating targets. The state error propagation matrix for the Bishop implementation is derived and a short simulation is produced to highlight the improved performance during components of trajectories with non-constant velocity.

II. Frenet-Serret and Bishop Acceleration LIEKFs

The invariant extended Kalman filter (IEKF) is a recent extension to the Kalman filter that enables the definition of state spaces on matrix Lie groups [11]. The key advantage of the IEKF is that by defining a left or right-invariant estimation error, the linearisation is performed on independent error dynamics. This ensures that the computed Kalman gain is not dependent on the accuracy of the current state estimate and hence convergence can be guaranteed for a wider range of trajectories [12]. Barrau and Bonnabel present a complete introduction to the IEKF in [13] with the Unscented variant covered in [14]. The non-accelerating form of the Frenet-Serret process model can be found in [6], [9]. Here, the attitude of the target is expressed as the Frenet-Serret or Bishop rotation matrix R_t as in [6]. The only change is to assume that an acceleration a_t acts on the target to update the norm velocity u_t. Changes in this acceleration, referred to as jerk, are modelled as Gaussian white noise. The equivalent Bishop frame dynamics are written as (3), substituting the curvature and torsion for the first and second Bishop curvatures κ_1, κ_2

$$\frac{d}{dt}\mathbf{x} = \begin{cases} \frac{d}{dt}R_t = R_t[\omega_{b,t} + w_t^{\omega}]_\times \in SO(3) \\ \frac{d}{dt}x_t = R_t(v_t + w_t^x) \in \mathbb{R}^3 \\ \frac{d}{dt}\kappa_t^1 = w_t^{\kappa^1} \in \mathbb{R}^1 \\ \frac{d}{dt}\kappa_t^2 = w_t^{\kappa^2} \in \mathbb{R}^1 \\ \frac{d}{dt}u_t = a_t + w_t^u \in \mathbb{R}^1 \\ \frac{d}{dt}a_t = w_t^a \in \mathbb{R}^1 \end{cases} \tag{3}$$

The target velocity v_t acts only in the tangential direction $v_t = \begin{bmatrix} u_t & 0 & 0 \end{bmatrix}^T$ and the Bishop Darboux vector is written as $\omega_{b,t} = \begin{bmatrix} 0 & -\kappa_2 & \kappa_1 \end{bmatrix}^T$. Note that since the filter estimates the target attitude, process noise for the position is only added in the tangential direction, that is $w_t^x = \begin{bmatrix} w_t^x & 0 & 0 \end{bmatrix}^T$. The state space is defined as $SE(3) \times \mathbb{R}^4$ which we will refer to as the manifold, noting that, since only part of the state is an element of the special Euclidean Lie group of 3D rigid body motion $SE(3)$, one cannot fully implement the IEKF [6]. The convergence guarantees presented in [12] are not valid for filters defined on mixed Lie group states however the IEKF still provides an elegant method for incorporating group constraints associated with common matrix Lie groups such as $SE(3)$. Additionally, the nature of the Frenet and Bishop formulae means that, in situations where the filter runs at frequency exceeding the measurement availability, the propagation is better suited to a wider range of trajectories.

A. IEKF Algorithm

This paper provides the key stages in deriving the state error propagation matrix for the Ba-LIEKF, but the same method can be easily applied to the Frenet-Serret case. To propagate the state error covariance we must first linearise the error dynamics. From [6], the state errors are defined as (4), a combination of left-invariant state error and linear vector error.

$$\eta = \begin{cases} \chi_t^{-1}\hat{\chi} \\ \hat{\zeta} - \zeta_t \end{cases} = \begin{bmatrix} \eta_t^R \\ \eta_t^x \\ \eta_t^{\kappa_1} \\ \eta_t^{\kappa_2} \\ \eta_t^u \\ \eta_t^a \end{bmatrix} = \begin{bmatrix} R_t^T \hat{R}_t \\ R_t^T(\hat{x}_t - x_t) \\ \hat{\kappa_1}_t - \kappa_{1t} \\ \hat{\kappa_2}_t - \kappa_{2t} \\ \hat{u}_t - u_t \\ \hat{a}_t - a_t \end{bmatrix} \tag{4}$$

With the true trajectory formed from the Bishop formulae in (3) and the noise-free filter models we can derive the error dynamics. Since Pilte et al. present this process for the Frenet-Serret formulae in [6] we proceed with the Bishop case. The time derivative of the error dynamics can be shown to be

$$\frac{d}{dt}\eta_t = \begin{bmatrix} -[\omega_{b,t} + w_t^{\omega}]_\times \eta_t^R + \eta_t^R[\hat{\omega}_{b,t}]_\times \\ -[\omega_{b,t} + w_t^{\omega}]_\times \eta_t^x + v_t + w_t^u - \eta_t^R \hat{v}_t \\ -w_t^{\kappa_1} \\ -w_t^{\kappa_2} \\ \eta_t^a - w_t^u \\ -w_t^a \end{bmatrix} \tag{5}$$

This can then be linearised using a first-order approximation which is shown by Barrau and Bonnabel to be exact [12]. The position and $\mathbb{R}^4$ states are assumed to follow $\xi_t = \eta_t$ while the rotation matrix R_t, an element of the special orthogonal group of 3D rotations $SO(3)$, follows a first order approximation of the exponential map for $SO(3)$, that is $\eta_t^R \approx I_3 + [\xi_t^R]_\times$. Substituting our linearised error definitions into equation (5) gives the linearised error equations shown in (6).

$$\frac{d}{dt}\xi_t = \begin{bmatrix} -[\omega_{b,t} + w_{b,t}^{\omega}]_\times (I_3 + [\xi_t^R]_\times) + (I_3 + [\xi_t^R]_\times)[\hat{\omega}_{b,t}]_\times \\ -[\omega_{b,t} + w_t^{\omega}]_\times \xi_t^x + v_t + w_t^u - (I_3 + [\xi_t^R]_\times)\hat{v}_t \\ -w_t^{\kappa_1} \\ -w_t^{\kappa_2} \\ \xi_t^a - w_t^u \\ -w_t^a \end{bmatrix} \tag{6}$$

By rearranging into the form $\dot{\xi}_t = A\xi_t + w_t$ with $w_t = \begin{bmatrix} w_t^{\omega} & w_t^x & w_t^{\kappa_1} & w_t^{\kappa_2} & w_t^u & w_t^a \end{bmatrix}^T$, the state error propagation matrix A_t for the accelerating Bishop equations can be derived as (7).

$$A_t = -\begin{bmatrix} 0 & -\hat{\kappa}_1 & -\hat{\kappa}_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hat{\kappa}_1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hat{\kappa}_2 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\hat{\kappa}_1 & -\hat{\kappa}_2 & 0 & 0 & -1 & 0 \\ 0 & 0 & -\hat{u}_t & \hat{\kappa}_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \hat{u}_t & 0 & \hat{\kappa}_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

It can be seen that for the state error propagation matrix in equation (7), the only change from the non-accelerating case in [6] is the addition of -1 in the final column. This can be repeated for the equivalent Frenet-Serret model by taking the A matrix from [6] and adding the final row and column from equation (7).

1) *Propagation*

With the state error propagation matrix derived, the complete IEKF algorithm can be implemented by first propagating the state using the Frenet-Serret and Bishop equations in [6] and equation (3). The error covariance can then be propagated using equation (8)

$$P_{k|k-1} = \Phi_k P_{k-1|k-1} \Phi_k^T + \check{Q}_k \tag{8}$$

where $\Phi_k = \exp_m(A_t \Delta t)$ and $\check{Q}_k \approx \Phi_k Q \Phi_k^T \Delta t$.

2) *Update Equations*

The FSa-LIEKF and Ba-LIEKF update step follows as equations (9) to (10)

$$K_k = P_{k|k-1} \tilde{H}_k (\tilde{H}_k P_{k|k-1} \tilde{H}_k^T + N_k^R)^{-1} \tag{9}$$

where $\tilde{H}_k$ is the measurement Jacobian of the spherical to Cartesian transformation rotated into the target frame as per [8], [10]. The error covariance is updated using the standard Kalman equation, although the Joseph form is recommended to avoid numerical issues associated with round-off errors.

$$P_{k|k} = (I_{10} - K_k \tilde{H}_k) P_{k|k-1} \tag{10}$$

Due to the composition of the state as a mixed manifold, the state update uses the boxplus $\oplus$ operator to correct the state.

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} \oplus K_k (Y_n - h(\hat{\mathbf{x}}_{k|k-1})) \tag{11}$$

This box-plus operator refers to the composition of a tangent-space element onto the manifold, with the $\ominus$ performing the opposite operation. These retain the left or right bias and as such we use the left $\oplus$ to update the state. This results in two separate operations for the $SE(3)$ Lie group and $\mathbb{R}^4$ vector components as shown in equation (12).

$$\begin{cases} \hat{\chi}_{k|k} = \hat{\chi}_{k|k-1} \exp_{SE(3)}(K_k^{\chi}(Y_n - h(\hat{\chi}_{k|k-1}))) \\ \hat{\zeta}_{k|k} = \hat{\zeta}_{k|k-1} + K_k^{\zeta}(Y_n - h(\hat{\chi}_{k|k-1})) \end{cases} \tag{12}$$

Here the Lie group state is updated using the exponential map of $SE(3)$ and a linear vector addition can be used for the $\mathbb{R}^4$ state.

III. Experimental Results

The IEKFs with the accelerating form of the Frenet-Serret and Bishop dynamic models are implemented in a radar tracking scenario with a target performing a trajectory comprising constant velocity, accelerating and spiralling segments. The scenario used is presented in Figure 1. The observer is kept stationary for simplicity and receives range and bearing measurements at 5Hz with uncertainties of $0.01 rad$ and $5m$ respectively. The filters update at 25Hz, propagating using the kinematic models when a measurement is not available. The process noises for all filters have been tuned manually.

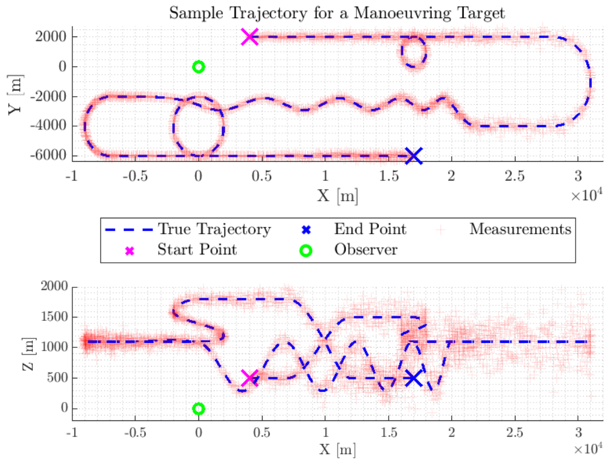


Fig. 1. Sample trajectory for a manoeuvring target used as the tracking scenario.

A. *Single Simulation*

The FSa-LIEKF, Ba-LIEKF are implemented and compared to the FS-LIEKF and B-LIEKF. For comparison to typical algorithms used in industry, a variety of Cartesian CV and CA filters are implemented, along with the CA-CV IMM2. Figure 1 depicts a single simulation comparing the FSa-LIEKF and Ba-LIEKF to their constant velocity counterparts.. All four algorithms perform well on this trajectory but the accelerating forms show slightly reduced tracking error during the decelerating components immediately before and after the spiral manoeuvre. As expected, the FSa-LIEKF and Ba-LIEKF are able to adapt to the changing velocities faster than the constant velocity counterparts but exhibit inferior performance on zero acceleration segments. Performance on accelerating segments could be further improved at the expense of increased noise during constant velocity trajectories. In a multiple-model algorithm the accelerating model could be tuned aggressively to maximise tracking during the accelerating segment before allowing the filter to switch to a FS-LIEKF or B-LIEKF filter. Since the filters are run independently for this simulation a balance is made. Figure 3 shows the FSa-LIEKF and Ba-LIEKF responding to changes in velocity slightly faster than the filters without the norm acceleration state. As the Bishop

Back to Contents

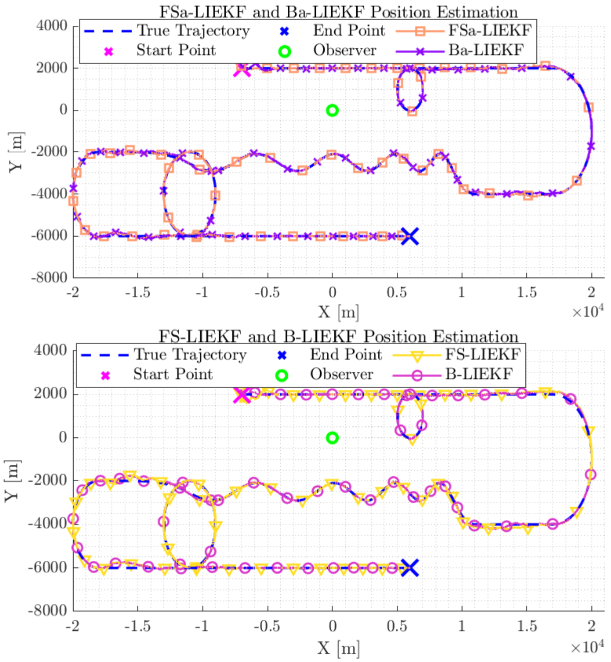


Fig. 2. Position Estimation of the Frenet-Serret and Bishop LIEKFs.

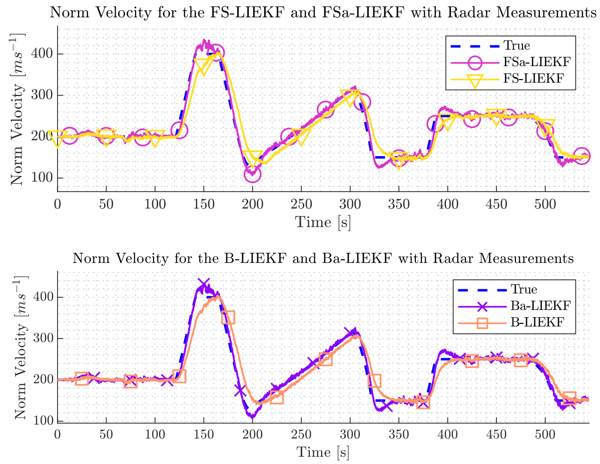


Fig. 3. Norm Velocity Estimation of the Frenet-Serret and Bishop LIEKFs.

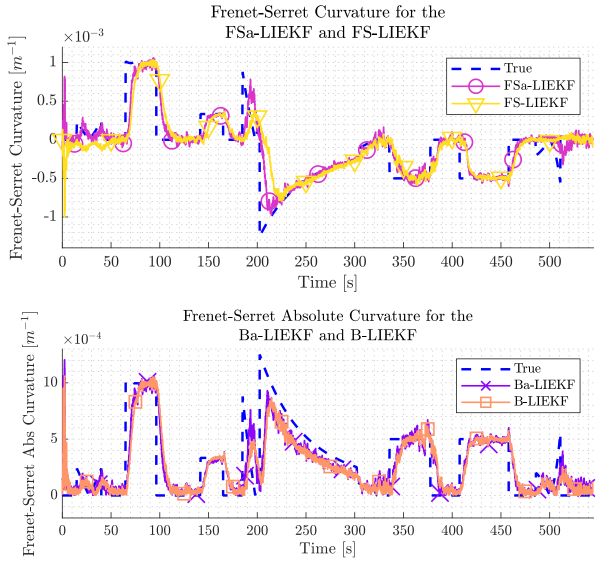


Fig. 4. Frenet-Serret Curvature Estimation of the Frenet-Serret and Bishop LIEKFs.

frame to a curve is not unique, it is hard to verify the accuracy of the estimation process for the two curvatures. This is because the Bishop formulae parallel transport the frame through a minimum rotation and will therefore change dependent on the initial frame. We plot the equivalent absolute Frenet curvature κ through $\kappa = \sqrt{\kappa_1^2 + \kappa_2^2}$. This is presented in Figure 4. It should be noted that we have chosen to define the Frenet-Serret curvature as a signed scalar, with clockwise turns assigned a positive curvature. Additionally, since the filters estimate $\hat{\kappa} = u\kappa$, the state estimate is divided by the estimated norm velocity for plotting. Both Bishop filters track the equivalent Frenet-Serret curvature well and, since the tracking of the curvature is dependent on accurate estimation of both curvatures, it suggests that the Bishop frame is able to estimate both scalars more effectively than the Frenet-Serret counterparts. Norm acceleration estimation of the Ba-LIEKF and FSa-LIEKF is depicted in Figure 5 with both filters performing well, although results could be improved with more aggressive process noise at the expensive of smoother velocity estimation.

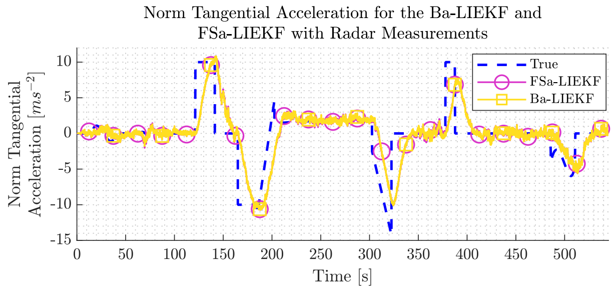


Fig. 5. Norm Tangential Acceleration Estimation of the Frenet-Serret and Bishop LIEKFs.

B. *Monte Carlo Simulation*

The results from a Monte Carlo analysis with 50 simulations provide a performance comparison for the IEKFs along with some basic but common industry algorithms. The root-mean-squared-errors (RMSE) of the position and norm velocity for each filter during the simulation are presented below in Tables I and II. The largest improvement in performance comes during manoeuvres not currently defined by the Frenet-Serret scalars, that is non-constant velocities, and it is here where both the Ba-LIEKF and FSa-LIEKF show their merits. The

TABLE I
LIEKF RMSEs for Simulation

State	B-LIEKF	Ba-LIEKF	FS-LIEKF	FSa-LIEKF
x	34.71	34.52	34.02	33.50
y	42.23	39.43	43.31	40.60
z	38.95	36.71	39.64	43.04
u	9.82	9.09	10.09	9.42

Back to Contents

TABLE II
EKF RMSEs For Simulation

State	CV-EKF	CA-EKF	IMM2-EKF
x	36.48	33.66	35.02
y	60.17	43.09	42.34
z	52.93	43.77	43.35
u	15.72	14.29	13.34

FSa-LIEKF and Ba-LIEKF show marginally improved norm velocity estimation, shown in Table I, although the trajectory presented has six segments with non-zero acceleration, so it is purposely well-suited to the FSa-LIEKF and Ba-LIEKF. Increased noise is seen during trajectory elements that do not require an acceleration term. Since the Frenet-Serret and Bishop formulae already allow for a broad range of motion, the use cases for the FSa-LIEKF and Ba-LIEKF are diminished. It is therefore recommended that the accelerating forms should only be used over the B-LIEKF and FS-LIEKF when a target is known to perform a large number of accelerating manoeuvres. The CV and CA filters are not ideally suited to some of the trajectory segments that would be best tracked by a coordinated-turn (CT) model, but Table II shows the CA-EKF performing better which, given the number of manoeuvres is reasonable. The IMM2 algorithm provides robust performance using simple Cartesian models but would benefit from an additional CT or Frenet-based model.

IV. Conclusion

This paper has presented an extension to the Frenet-Serret and Bishop target models to account for tangential accelerations in the target kinematics. The left-invariant state error propagation matrices have been derived and implemented in LIEKF algorithms to track a manoeuvring target. The FSa-LIEKF and Ba-LIEKF are shown to be more accommodating to trajectories with accelerating components, closely tracking the changes in velocity with the detriment of increased noise during non-accelerating segments. This demonstrates that the addition of the acceleration term only improves small parts of the trajectory and the improvement on the original filters is marginal as the FS-LIEKF and B-LIEKF provide robust, single-model performance. The acceleration term also adds complexity in the tuning process and additional care is required to optimise the filter performance. Based on the simulation undertaken, the original B-LIEKF and FS-LIEKF are more than well equipped to estimate complex trajectories, and the accelerating forms would be complementary extensions in a multiple-model algorithm.

A. Future Work

With two kinematic models available for each filter, we plan on developing an invariant-IMM based on [15] or multiple-model particle filter to embed the geometric models into more complex tracking algorithms.

Acknowledgment

This work is partly supported by PhD funding from Leonardo UK.

References

[1] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on automatic control*, vol. 45, no. 3, pp. 477–482, 2000.

[2] I. Arasaratnam and S. Haykin, "Cubature kalman filters," *IEEE Transactions on automatic control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[3] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, pp. 473–483, 1970.

[4] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Transactions on aerospace and electronic systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[5] X. R. Li and Y. Bar-Shalom, "Performance prediction of the interacting multiple model algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, pp. 755–771, 1993.

[6] M. Pilté, S. Bonnabel, and F. Barbaresco, "Tracking the frenet-serret frame associated to a highly maneuvering target in 3d," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 1969–1974.

[7] R. L. Bishop, "There is more than one way to frame a curve," *The American Mathematical Monthly*, vol. 82, no. 3, pp. 246–251, 1975.

[8] J. Gibbs, D. Anderson, M. Macdonald, J. Russell, "Invariant extended kalman filter for tracking the bishop frame using radar measurements," in *International Conference on Radar Systems*. IET, to be published 2022.

[9] P. Marion, J. Sami, B. Silvère, B. Frédéric, F. Marc, and H. Nicolas, "Invariant extended kalman filter applied to tracking for air traffic control," in *2019 International Radar Conference (RADAR)*. IEEE, 2019, pp. 1–6.

[10] M. Pilte, "Dynamic management of tracking ressources for hyper-manoeuvring targets," Ph.D. dissertation.

[11] S. Bonnabel, "Left-invariant extended kalman filter and attitude estimation," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 1027–1032.

[12] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2016.

[13] A. Barrau and S. Bonnabel, "Invariant kalman filtering," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 237–257, 2018.

[14] M. Brossard, A. Barrau, and S. Bonnabel, "A code for unscented kalman filtering on manifolds (ukf-m)," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5701–5708.

[15] T. L. Koller and U. Frese, "The interacting multiple model filter and smoother on boxplus-manifolds," *Sensors*, vol. 21, 6 2021.

Back to Contents

Joint Undervolting and Overclocking Power Scaling Approximation on FPGAs

Yun Wu, João F. C. Mota, Andrew M. Wallace
The School of Engineering and Physical Sciences
Heriot-Watt University, Edinburgh, UK
{y.wu, j.mota, a.m.wallace}@hw.ac.uk

***Abstract*—For applications in signal processing, Field Programmable Gate Arrays (FPGAs) are more flexible than Application Specific Integrated Circuits (ASICs), yet reconfigurable and still power and energy efficient to a degree. Undervolting and overclocking are approximate computing techniques that can further save power and energy, closing the efficiency gap by reducing the static/dynamic power and potentially speeding up the computation. However, these techniques may introduce bit level faults, which affect not only the computational correctness but also the security of the hardware. Understanding these fault behaviors provides necessary information for approximate implementation in low-power and secure design.**

In this work, we investigate joint undervolting and overclocking of AXI peripherals, specifically on-chip AXI memory access, using different commercial Xilinx Ultrascale+ heterogeneous MPSoCs with practical data movement between the ARM processor and the FPGA. Through experimental study we have observed fine-grained bit-flipping patterns when the voltage and clock are tuned beyond certain thresholds. By judging the probability of bit-flipping in terms of bit error rate, we propose a guideline for a balanced choice of voltage and frequency.

***Index Terms*—Approximate Computing, FPGA, Undervolting, Overclock**

I. Introduction

Field Programmable Gate Arrays (FPGAs) can accelerate performance in a range of applications more effectively than software implementations [1], especially in the signal processing domain [2], where lower power and real-time performance are highly demanded [3]. While they are still less energy-efficient than Application Specific Integrated Circuits (ASICs), undervolting and overclocking are effective approximate computing techniques which can potentially bridge this gap by reducing the power/energy consumption and improving the efficiency in terms of performance per Watt [4].

On the one hand, tuning the voltage below the nominal level of the factory setting, i.e. undervolting, can significantly improve the energy efficiency of hardware [5]. This is true for processing units, such as Central Processing Units (CPUs) [6], Graphics Processing Units (GPUs) [7], FPGAs [8], and ASICs [9], and data storage units, such as Dynamic RAMs (DRAMs) [10] and Static RAMs (SRAMs) [11]. Dynamic voltage and frequency scaling (DVFS) [12] adjusts the voltage and frequency accordingly and saves power/energy dramatically while maintaining the default operating frequency. On the other hand, by surpassing the nominal clock rate or factory setting, overclocking can break through the limitation of processing performance, especially for FPGA accelerators running at lower clock frequencies than the CPU or GPU [13].

Either undervolting or overclocking leads to timing faults, which can cause a system crash or a termination of applications with processing errors. Such errors are characterized as coarse-grained [4], [14]. However, to further understand the vulnerability of applications on FPGAs, as well as how to mitigate the impact of errors, there has been little emphasis on fine-grained error characterization. In this work, we study the combined effects of undervolting and overclocking and characterize fine-grained, bit-flipping errors.

Contributions. We summarize our contributions as follows:

- We develop an infrastructure for joint undervolting and overclocking on an FPGA
- We characterize the fine-grained bit-flipping errors
- We investigate the probability of bit-flipping errors and provide a guideline for design space exploration

The rest of the paper is organized as follows. In Section II, we describe related work on undervolting and overclocking on FPGAs, contrasting it with our approach. In Section III, the hardware/software deployment is illustrated. The vulnerability analysis is presented in Section IV. A conclusion is given in Section V.

II. Background

The power consumption of modern digital signal processing circuits, such as FPGAs, is directly related to their supply voltage level and operating frequency. Total power can be decomposed of dynamic and static power, as shown in Eq. (1).

$$P_{\text{total}} = P_{\text{dynamic}} + P_{\text{static}}. \tag{1}$$

Dynamic and static power are modeled by

$$P_{\text{dynamic}} = \alpha \cdot C \cdot F \cdot V^2, \tag{2}$$

$$P_{\text{static}} = \sum^{C} I_{\text{leakage}} \cdot V, \tag{3}$$

where α is a constant depending on the process technology, C is the capacitance of resource utilization, F is the operating frequency, V is the supply voltage, and I_{leakage} is the leakage current [2]. Hence, from (2) and (3), power savings can be achieved by decreasing α, C, F , V, and I_{leakage}, and throughput can be improved by increasing F.

This work was supported by EPSRC Grant number EP/S000631/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

Many techniques have been explored to minimize the power consumption of FPGAs based on Eqs. (1), (2) and (3), e.g. architectural improvements to reduce α [15], clock or voltage gating [16], [17] to decrease C and P_{static}, DVFS [12] to reduce both F and V. Since both dynamic and static power are directly related to the supply voltage V, undervolting can deliver further power savings and improve energy efficiency [4].

Besides power, if an FPGA has an operating frequency lower than a connected CPU or GPU, this limits the performance of the FPGA accelerator and hence the overall system. By overclocking the operating frequency F, safety margins of error-free computation can be explored. Shi [13] combines overclocking with approximate circuits on an FPGA, and Rowlings [14] further demonstrates the benefits of overclocking in fault-tolerant application.

Both undervolting and overclocking improve energy efficiency. However, they also increase the vulnerability of FPGAs leading to timing violations. Salami [4] has characterized the voltage-to-errors of memory units on FPGAs. Rowlings [14] developed a close-loop overclocking method for a spiking neural network accelerator and investigated frequency-to-errors at the application level. However, both are coarse-grained characterizations of the relation between tuning parameters and error probability, and an investigation of fine-grained bit level error probability is missing.

Given that the Advanced eXtensible Interface (AXI) is the most common data transfer protocol on FPGAs, we concentrate on the impact of combined undervolting and overclocking on the AXI data path to characterize the fine-grained bit error.

III. System Deployment

To characterize the vulnerability of the AXI data path, we have designed hardware with different sizes of data path, and a software instrument for tuning and measuring the voltage/power levels, as well as a data stream driver for the target data path.

A. Hardware Deployment

Figure 1 shows the hardware architecture of the Processing Systems (PS) and Programmable Logic (PL) on a Xilinx Ultrascale+ FPGA, with AXI datapath access to BRAM and its controller on the PL side.

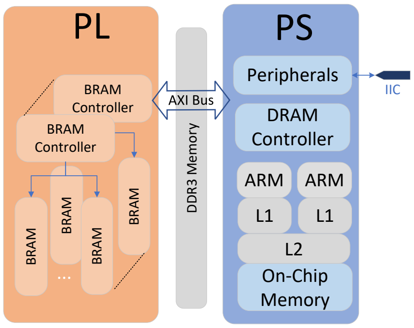


Fig. 1: Hardware Infrastructure

On the PS side, the AXI Master is configured for writing data from the PS to the PL. The AXI Slave is configured to receive data from the PL. The I^2C peripheral I/O is configured to enable PS PMBus access, while the SD peripheral I/O is configured to allow Linux booting and data preservation through an SD card. Using the I^2C interface, the output voltage of the power rails controlled by on-chip power management integrated chip (PMIC) can be modified through a PMBus protocol command by running software on ARM processor.

On the PL, BRAMs are linked to the AXI interface controller to form a single memory block. This is achieved by aggregating multiple instances of BRAM with the AXI interface through the Xilinx tool. We can generate the datapath using varying numbers of BRAMs, configured to perform data movement through a developed device driver. More specifically, we produced a BRAM controller with varying memory addressing sizes (e.g. 16, 64 and 128 Kbytes) and with varying BRAM location through post-synthesis allocation using the LOC semantic in-design constraint.

The bit level faults are investigated at various datapath locations, without exhaustively optimize design but separated elements on the FPGA. We use the generic BRAM IP on Zynq PL through Xilinx Vivado, which is a hybrid of BRAM, registers, and Look-Up-Tables. Note that all the generated designs are processed at compile-time, producing bit files for the PL configuration. These bit files are saved on an SD card, identified by case name for later run-time reconfiguration and evaluation. The resource utilization data from Table I are also recorded for different sizes of the datapath.

TABLE I: AXI Datapath Cost

Datapath. Size (KB)	Reg.	LUT	BRAM	Address	Frequency (MHz)
4	3954	3160	1	0x0-0xFFF	100
16	3974	3171	4	0x0-0x3FFF	100
32	3984	3177	8	0x0-0x7FFF	100
64	3994	3175	16	0x0-0xFFFF	100
128	4004	3191	32	0x0-0x1FFFF	100

As shown, every AXI datapath uses 1 BRAM with 4KB size. The address range increases as the data path size grows, which is used for later data partitioning to access specific BRAMs. Using 11 AXI controllers in Fig. 1, all 312 ($32 \times 9 + 16 + 8$) BRAMs on ZCU104/106 boards can be traversed by addressing.

B. Software Deployment

We developed the device driver to access the AXI datapath by mapping the address of the BRAM controller to main memory while the data is transmitted through the AXI bus protocol between the PS and PL. A specific data value switching behavior is designed, from blocked to interleaved bit-flipping patterns, when moving the data to and from the AXI datapath. Table II lists the designed 32 bit data patterns, where the paired data patterns are exchanged in turn during the transfer. This ensures active bit alternation, which maximizes the chance of capturing any bit-flipping errors when undervolting and overclocking.

TABLE II: Data Switching Patterns

Case No.	Patterns (64 Bits)
c1	0x00000000 / 0xFFFFFFFF
c2	0xFFFF0000 / 0x0000FFFF
c3	0xFF00FF00 / 0x00FF00FF
c4	0xF0F0F0F0 / 0x0F0F0F0F
c5	0x33333333 / 0xCCCCCCCC
c6	0x55555555 / 0xAAAAAAAA

To traverse all the BRAM in the AXI data path, the transfer data is partitioned according to the specific addresses in Table I, with a 4 Bytes addressing interval for 32 bit data patterns. Hence, given a 4KB range for a single BRAM, all transferred data within every 0xFFF address space belongs to a distinct BRAM. By constraining the location of BRAM during the place and route implementation for each AXI controller in sequential order, the location of each BRAM in the AXI data path is known before the data transfer, which is bound to the partitioned data pattern to characterize the bit-flip errors. Fig 2 shows an example of a pre-located AXI data path at the tiles of chip layout.

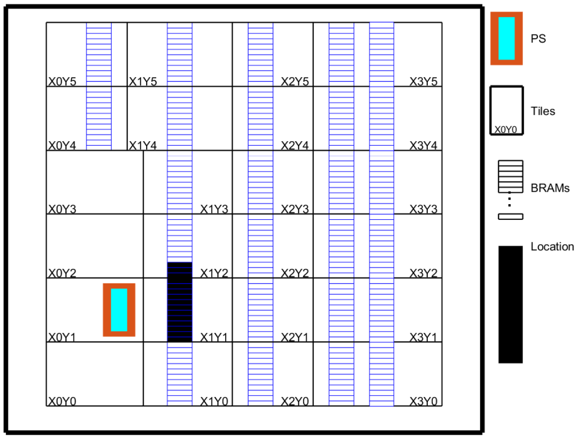


Fig. 2: Post-Synthesis Localization of BRAM

Besides the device driver, to tune the voltage and clock, we also developed a power management infrastructure that can monitor and scale the voltage through the I^2C interface, as well as clock scaling through a Phase Locked Loop (PLL) unit through the PL clock tree Linux device driver (LDD). By testing the voltage and clock scaling on the device, the lowest voltage and highest clock before the device halts are recorded as prior knowledge.

All data patterns are written firstly to BRAM through the AXI data path using the nominal voltage and frequency. Undervolting and overclocking are only performed when reading those pre-written data patterns. Before testing each pair of patterns, the nominal voltage and frequency are reset. Fig. 3 shows the entire software infrastructure for iterative testing of the various data patterns and recording of bit-flipping errors.

The voltage and frequency are set through I^2C and PLL LDD to initialize the system and later undervolting and overscaling. The written data pattern is transferred through the AXI driver between the PS and PL by the AXI data path. With the recorded error values and the pre-setup BRAM location in the AXI data path, the corresponding bit-flipping errors can be derived by masking the read pattern with the written patterns. The whole software infrastructure is iterated for different data patterns, as well as the various voltage and frequency scalings.

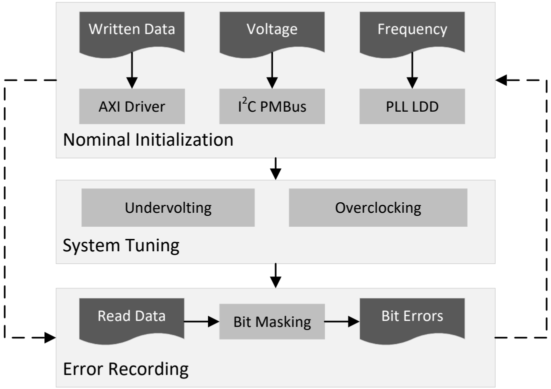


Fig. 3: Software Infrastructure

IV. Vulnerability Analysis

Power scaling is evaluated on two MPSoC evaluation development boards, the Xilinx Zynq Ultrascale+ and ZCU104/106, using the same FPGA XCZU7EV. The AXI data path IP is deployed on both boards using the Xilinx Vivado tool-set version 2019.1.

A. Undervolting

There are several power rails on these two boards, controlling the power supply to different areas. Power scaling targets only the FPGA related power rails, i.e $VCCINT$, which is the power supply rail for the programmable logic system, and $VCCBRAM$, which is the power supply rail for the BRAM. The nominal voltages of $VCCINT$ and $VCCBRAM$ are 0.85V and 0.9V. The official undervolt margins of $VCCINT$ and $VCCBRAM$ are 0.7282V and 0.7253V. Testing the limits of undervolting, 0.59V and 0.52V were determined as the lowest voltages for $VCCBRAM$ and $VCCINT$ respectively, beyond which the board halts.

Undervolting is evaluated alone by changing $VCCBRAM$ from 0.90V to 0.59V, and $VCCINT$ from 0.85v to 0.52V, keeping the nominal frequency 100MHz. There is a one second interval for data transfer of each data pattern. By comparing the read data to the pre-written data, the AXI data path is robust, where no error is captured. Table III shows the relation between undervolting and total power consumption of each power rail for the AXI data path with all 312 BRAMs on the ZCU106.

N/A indicates that no power is recorded due to the halting of the FPGA with insufficient voltage. In the second column, the

Back to Contents

TABLE III: Undervolting Power Consumption (100MHz)

$VCCINT$ (V)	Power (mW)	$VCCBRAM$ (V)	Power (mW)
0.85	297.33 (173.50)	0.90	44.26
0.80	262.54 (153.15)	0.85	36.51
0.75	231.86 (137.56)	0.80	29.17
0.70	203.21 (122.03)	0.75	22.86
0.65	177.19 (107.65)	0.70	16.81
0.60	153.35 (95.43)	0.65	11.45
0.55	131.21 (82.69)	0.60	6.56
0.52	119.21 (75.19)	0.59	5.59
0.51	N/A	0.58	N/A

number in the parentheses is the baseline power when no bitstream is configured to the programmable logic, which is only applicable to the $VCCINT$ power rail. We observe that no bit-flipping error happen when using undervolting only, where about 60% saving in power is achieved.

B. Overclocking

Next, overclocking is evaluated alone on the ZCU106. The operating frequency is scaled from 100MHz to 499 MHz, keeping the nominal voltage. Again, there are no errors. Table IV shows the total power consumption with increasing operating frequency.

TABLE IV: Overclocking Power Consumption (0.85/0.9 Volt)

$Frequency$	$VCCINT$	$VCCBRAM$
(MHz)	Power (mW)	Power (mW)
100	297.33	44.26
199	420.03	44.26
299	564.54	44.26
399	665.69	44.26
499	826.96	44.26

Overclocking can achieve a 5× times performance boost, but at a cost of about 2.8× times power consumption on $VCCINT$, while keeping $VCCBRAM$ unchanged. From this point of view, overclocking increases the energy efficiency about 1.8× without errors on the AXI data path, since there is more gain in performance than in power consumption. This is because static power is independent of frequency F as shown in Eq. 3.

C. Undervolting with Overclocking

By combining undervolting with overclocking, bit-flipping errors happen when the frequency is above 299Mhz with $VCCBRAM$ at 0.59V and $VCCINT$ below the nominal voltage. Fig. 4 illustrates the normalized bit error ratio of 32 bits, which is evaluated with 1000 samples of each data pattern across 11 AXI BRAM controllers.

As shown in Figs. 4, all bit-flipping errors are most likely to occur in the lower range, which corresponds to the least significant arithmetic bits. We also checked the higher frequenies, for example 0.56V and 0.72V with 399MHz and 499MHz relatively. It is clear that the minimum voltage before the device halts increases, while the lower the voltage, the more bit-flipping errors occur.

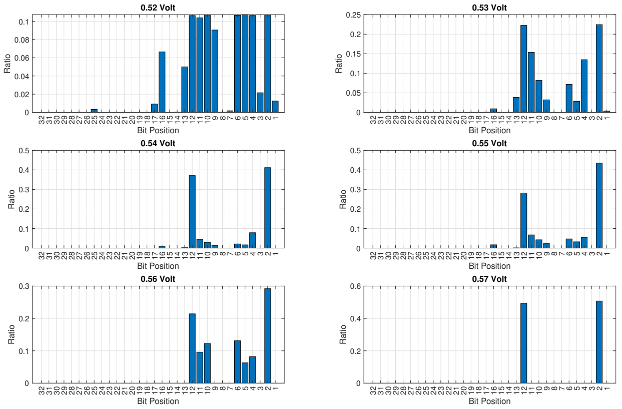


Fig. 4: Bit Flipping Statistics at 299MHz

We further checked bit-flipping on the ZCU104. Not repeating all measurements, 399MHz is chosen for comparison with $VCCBRAM = 0.56V$ and $VCCBRAM = 0.59V$. Every 16 BRAM locations of the AXI data path are rotated across the different tiles on the chip. Fig. 5 shows the bit-flipping errors at different locations on the ZCU104, where x and y are the axes of tile location.

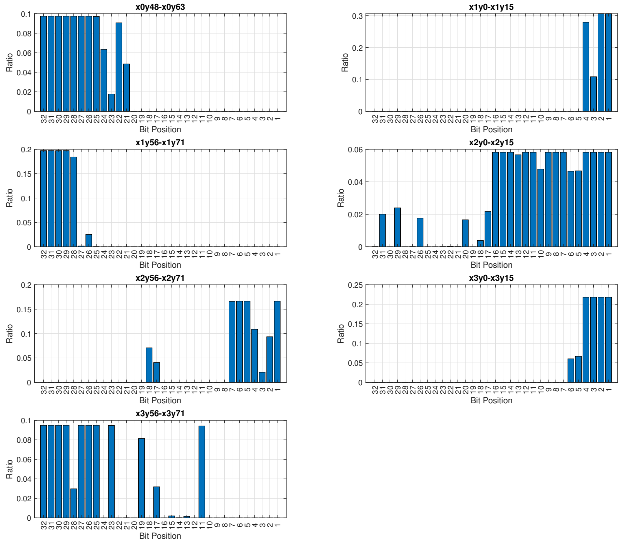
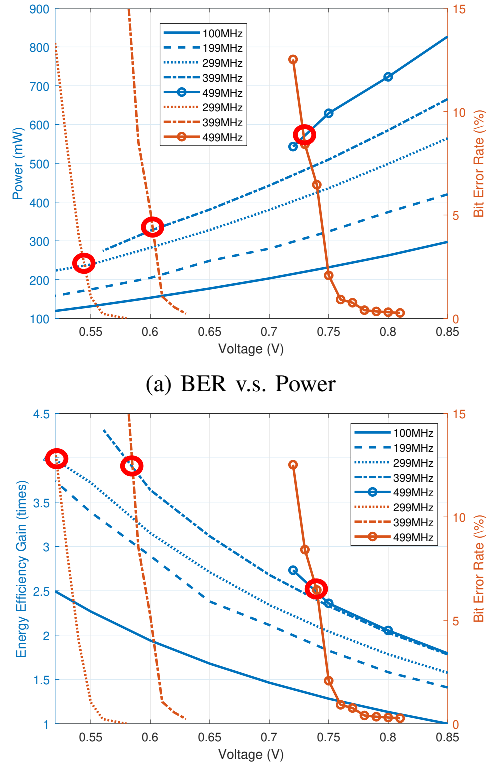


Fig. 5: Bit Flipping Statistics at Different Locations

As shown, the error pattern varies with location of BRAM on the AXI data path. Furthermore, unlike the ZCU106, the bit-flipping position happens at both the most and least significant bits. By defining the ratio of overall bit-flipping in 32 bits for the AXI data path as the bit error rate (BER), Fig. 6 shows both the energy efficiency and power consumption, where the energy efficiency is the ratio of performance gain over normalized power.

As shown in Fig. 6a, power increases from undervolting to

Back to Contents



(a) BER v.s. Power

(b) BER v.s. Efficiency

Fig. 6: BER Performance

the nominal voltage, while BER decreases. There are sweet spots for combined undervolting and overclocking where the BER decrement is across power increment at a specific voltage, e.g. 0.54V, 0.61V, and 0.73V for 299MHz, 399MHz, and 499MHz respectively. In Fig. 6a, the energy efficiency gain decreases as the voltage increases when combining undervolting and overclocking. Such sweet spots also exist when considering the cross point between BER and energy efficiency gain, 0.52V, 0.59V, and 0.73V for the corresponding overclocking frequencies. From a defined energy efficiency or power budget requirement, we can determine the optimal undervolting voltage and overclocking frequency for tolerable BER.

V. Conclusions

We have developed an infrastructure to investigate the fine-grained vulnerability of joint undervolting and overclocking for the AXI data path on an FPGA. Deploying both the hardware and software infrastructure, we are able to scale the voltage and frequency of commercial FPGAs and capture the bit-flipping errors when transferring regular switching data patterns on the AXI data path. Evaluating two distinct devices, we have found that the bit-flipping error patterns are device dependent, even when using the same type of FPGA chip, and also location dependent on the same chip. However, our evaluation shows that a balanced choice of undervolting and overclocking frequency can lead to very significant gains in energy efficiency, for example by a factor of four, and power reduction, for example by a factor of three, while maintaining accurate performance. This provides a meaningful guide for further investigation of efficient signal processing design with different arithmetic and for different applications.

References

[1] J. Fowers, G. Brown, P. Cooke, and G. Stitt, "A performance and energy comparison of fpgas, gpus, and multicores for sliding-window applications," in In Proceedings of the ACM/SIGDA international symposium on Field Programmable Gate Arrays, 2012, pp. 47–56.

[2] R. Woods, J. Mcallister, R. Turner, and et al., FPGA-Based Implementation of Signal Processing Systems. Wiley Publishing, 2008.

[3] V. MADISETTI, The Digital Signal Processing Handbook, 2nd ed. USA: CRC Press, Inc., 2009.

[4] B. Salami, O. Unsal, and A. Cristal, "Fault characterization through fpga undervolting," in 2018 28th International Conference on Field Programmable Logic and Applications (FPL), 2018, pp. 85–853.

[5] U. Kulau, F. Büsching, and L. Wolf, "Undervolting in wsns — a feasibility analysis," in 2014 IEEE World Forum on Internet of Things (WF-IoT), 2014, pp. 553–558.

[6] R. Bertran, P. Bose, D. Brooks, and et al., "Very low voltage (vlv) design," in 2017 IEEE International Conference on Computer Design (ICCD), 2017, pp. 601–604.

[7] J. Leng, A. Buyuktosunoglu, R. Bertran, and et al., "Safe limits on voltage reduction efficiency in gpus: A direct measurement approach," in 2015 48th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2015, pp. 294–307.

[8] Y. Wu, J. Nunez-Yanez, R. Woods, and D. S. Nikolopoulos, "Power modelling and capping for heterogeneous arm/fpga socs," in 2014 International Conference on Field-Programmable Technology (FPT), 2014, pp. 231–234.

[9] P. N. Whatmough, S. K. Lee, H. Lee, and et al., "14.3 a 28nm soc with a 1.2ghz 568nj/prediction sparse deep-neural-network engine with gt;0.1 timing error rate tolerance for iot applications," in 2017 IEEE International Solid-State Circuits Conference (ISSCC), 2017, pp. 242–243.

[10] K. K. Chang, A. G. Yağlıkçı, S. Ghose, and et al., "Understanding reduced-voltage operation in modern dram devices: Experimental characterization, analysis, and mechanisms," Proc. ACM Meas. Anal. Comput. Syst., vol. 1, no. 1, jun 2017. [Online]. Available: https://doi.org/10.1145/3084447

[11] L. Yang and B. Murmann, "Sram voltage scaling for energy-efficient convolutional neural networks," in 2017 18th International Symposium on Quality Electronic Design (ISQED), 2017, pp. 7–12.

[12] Y. Wu, D. S. Nikolopoulos, and R. Woods, "Runtime support for adaptive power capping on heterogeneous socs," in 2016 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS), 2016, pp. 71–78.

[13] K. Shi, D. Boland, and G. A. Constantinides, "Accuracy-performance tradeoffs on an fpga through overclocking," in 2013 IEEE 21st Annual International Symposium on Field-Programmable Custom Computing Machines, 2013, pp. 29–36.

[14] M. Rowlings, A. M. Tyrrell, and M. A. Trefzer, "Operating beyond fpga tool limitations: Nervous systems for embedded runtime management," in 2021 Design, Automation Test in Europe Conference Exhibition (DATE), 2021, pp. 68–71.

[15] Xilinx, Power Analysis and Optimization, accessed on March 1, 2022. [Online]. Available: https://docs.xilinx.com/v/u/2020.1-English/ug907-vivado-power-analysis-optimization

[16] B. Pandey, J. Yadav, M. Pattanaik, and N. Rajoria, "Clock gating based energy efficient alu design and implementation on fpga," in 2013 International Conference on Energy Efficient Technologies for Sustainability, 2013, pp. 93–97.

[17] T. Tuan, S. Kao, A. Rahman, and et al., "A 90nm low-power fpga for battery-powered applications," in Proceedings of the 2006 ACM/SIGDA 14th International Symposium on Field Programmable Gate Arrays, ser. FPGA '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 3–11. [Online]. Available: https://doi.org/10.1145/1117201.1117203

State Estimation of the Spread of COVID-19 in Saudi Arabia using Extended Kalman Filter

Lamia Alyami
Department of Mathematics, College of Engineering,
Mathematics and Physical Sciences, University of Exeter,
Penryn Campus TR10 9FE, United Kingdom.
Email: la424@exeter.ac.uk

Saptarshi Das, *Member, IEEE*
Department of Mathematics, College of Engineering,
Mathematics and Physical Sciences, University of Exeter,
Penryn Campus TR10 9FE, United Kingdom.
Email: S.Das3@exeter.ac.uk, saptarshi.das@ieee.org

***Abstract*—COVID-19 has caused global concern as the World Health Organization (WHO) considered it a global pandemic that has affected all countries to different extent. Numerous studies have examined the behaviour of the pandemic using a wide variety of mathematical models. In this paper, we consider the nonlinear compartmental epidemiological dynamical system model in the Susceptible-Exposed-Infected-Quarantined-Recovered-Deceased (SEIQRD) form based on the recursive estimator known as the extended Kalman filter (EKF) to predict the evolution of the COVID-19 pandemic in Saudi Arabia. We adopt the nested sampling algorithm for parameter estimation and uncertainty quantification of the SEIQRD model parameters using real data. Our simulation results show that the EKF can not only predict the evolution of the directly measured variables i.e. the total death (*D*) and active case (*I*) but can also be useful in the estimation of the unmeasurable state variables and help predicting their future trends.**

***Index Terms*—Extended Kalman Filter (EKF), SEIQRD model**

I. Introduction

Coronaviruses SARS-CoV-2 (Severe acute respiratory syndrome coronavirus 2) or the COVID-19 are a group of viruses that has become a global pandemic recognised by the WHO, affecting every country. The COVID-19 virus has high transmissibility which has caused infections and increased the burden on the public health, especially for older people. The WHO declared the COVID-19 as a global pandemic in March 2020 [1] which started a rapid increase in research in epidemiological modeling using different countries' data. Now, with more mutated variants of this virus, it is crucial to investigate and predict the long-term behaviour of such infectious disease models from an early stage until the end of the pandemic. In March 2022 there were more than 400 million confirmed cases of the coronavirus and the death cases have exceeded 6 million. The fundamental epidemiological model called SIR *(Susceptible-Infected-Recovered)* introduced by [2] has been utilized widely to predict the COVID-19 outbreak. SIR and its modifications such as SEIR *(Susceptible-Exposed-Infected-Recovered)* etc., have been discuss in details in [3]. Developing the SEI model within Bayesian framework and estimate the posterior probability distributions for parameters of interest using Markov chain Monte Carlo (MCMC) method was reported in [4]. The SEIR model in [5] demonstrated a consistent prediction between the model outputs and real COVID-19 data in Saudi Arabia. Later in [6] the SEIQR *(Susceptible-Exposed-Infected-Quarantine-Recovered)* model has been studied for stability analysis of the model based on Saudi Arabia infection daily data. The SEIQRD *(Susceptible-Exposed-Infected-Quarantine-Recovered-Dead)* model was used in [7] for risk management to forecast the spread of COVID-19 pandemic in different countries.

As opposed to these existing works, the aim of this paper is to construct a new epidemiological model SEIQRD with reinfection to explain the long-term COVID-19 spread in Saudi Arabia. This will then enhance the prediction using Kalman filtering algorithms and help understand the behaviour of the underlying dynamic process and track hidden or unmeasurable variables of the compartmental model. With this aim, we have investigated the utility of the KF family of algorithms in COVID-19 spread. The KF algorithm is a recursive technique to generate estimates of the state variables based on measured time-series data of a fewer variables (e.g. active case and total death) often corrupted with noise and bias. The KF algorithm is well-known as an optimal recursive solution of the discrete-time linear observer or state estimation problem [8]. The optimality of the KF is in the sense of minimizing the mean squared error (MMSE). The KF algorithm operates in two steps: 1) predicting the current state estimates from the previous estimates; 2) updating the estimates of the filter using the error covariance matrices with the feedback mechanism. Hence, the KF is referred as a predictor-corrector algorithm.

The drawback of the classical KF algorithm is strictly applicable to the linear systems and Gaussian noise. For this reason KF has been extended to other different versions. One of its variant solves the model nonlinearity problem, known as the Extended Kalman Filter (EKF). The EKF is an approximation filter for the nonlinear systems based on first-order Taylor series expansion, evaluated at each time step around the current state. More information on EKF can be found in [9], [10]. The EKF can be applied in tracking the epidemiological processes but there has been limited amount of literature available, addressing the use of EKF in tracking the spread of other types of infectious diseases e.g. [11], [12]. For using the EKF to estimate the COVID-19 spread, there are only few literature addressing the COVID-19 pandemic. In [13], the

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

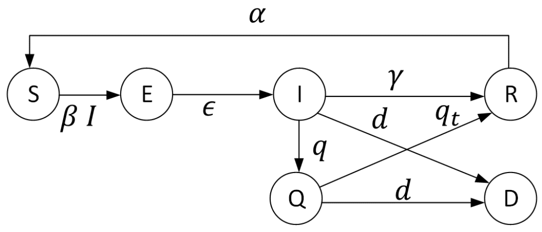


Fig. 1: The proposed SEIQRD model with reinfection.

authors used the maximum likelihood estimation (MLE) in the EKF to predict the COVID-19 transmission in China and USA. Also, [14] proposed the use of EKF to simulate stochastic and deterministic models with greater accuracy for the prediction of COVID-19 behaviour. In [15] the authors have implemented the EKF to predict the evolution of the COVID-19 pandemic, within a short time span. Since the rapid outbreak of the COVID-19, the accuracies of different dynamic models play an important role in prediction and decision-making. Here, the COVID-19 pandemic data has been used to fit a newly proposed model of SEIQRD form with reinfection term along with quantifying parametric uncertainties. We have considered a reinfection term in the model since the reinfections cases are reported in different sources e.g. [16]. Based on this model, the EKF has been used to analyze the COVID-19 behaviour over long-term. The simulation of the proposed method is applied for the COVID-19 data outbreak in Saudi Arabia from 15 February 2020 to 17 March 2022. Moreover, the real data has been compared with the EKF predictions on the measured states beside estimating the trends of the unmeasurable states of this SEIQRD compartmental model.

II. Description of the Proposed SEIQRD Model

The SEIQRD model divides the population into six classes: susceptible $S(t)$, exposed $E(t)$, infected $I(t)$, quarantined $Q(t)$, recovered $R(t)$ and dead $D(t)$. The compartments were modelled using the system illustrated in Fig. 1 as the nonlinear differential equation or nonlinear state space model as follows:

$$
\begin{aligned}
\frac{dS}{dt} &= -\beta IS + \alpha R,\\
\frac{dE}{dt} &= \beta IS - \epsilon E,\\
\frac{dI}{dt} &= \epsilon E - \gamma I - qI - dI,\\
\frac{dQ}{dt} &= qI - q_t Q - dQ,\\
\frac{dR}{dt} &= \gamma I + q_t Q - \alpha R,\\
\frac{dD}{dt} &= dI + dQ.
\end{aligned}
\tag{1}
$$

The model parameters $\{\beta,\ \gamma,\ \epsilon,\ q,\ q_t,\ \alpha,\ d\}$ are non-negative and defined as the infection rate, recovery rate, incubation rate, quarantine rate, quarantine period, reinfection rate, and death rate respectively. Since the length of the protective immunity is unknown, we consider the possibility of reinfection after recovery where a fraction (α) of the recovered population returns to the susceptible compartment and when $\alpha = 0$ our proposed model coincides with the SEIQRD model proposed in [17]. It is a closed compartmental model (2) defined as:

$$S + E + I + Q + R + D = N, \tag{2}$$

defining N as the total size of the population of a country under study. The basic reproduction number R_0 for the proposed model can be defined based on the next generation matrix proposed in [18] as:

$$R_0 = \frac{N\beta}{d + \gamma + q}. \tag{3}$$

III. Methodology for Model Parameter and Uncertainty Estimation

The official epidemic data is frequently published on a daily or weekly basis. The reported measurements are usually in discrete time domain (cases per day) whereas the epidemiological model is in continuous time. Therefore, the model under local linearization needs to be discretized to match with the dataset collected on a daily basis using the hybrid extended Kalman filter (discrete-continuous EKF). The first step of this is estimating the parameters of the SEIQRD model using a Bayesian uncertainty quantification method called the nested sampling algorithm that draws samples from the posterior distribution of the unknown dynamic model parameters as a by-product while calculating the Bayesian evidence or the marginal likelihood [19]. Using the mean estimate of the posterior distribution of the unknown model parameters, we carry out the state estimation using the EKF approach. We consider the prediction of COVID-19 spread using the locally linearized hybrid EKF using the mean posterior of the new SEIQRD model parameters to estimate the unmeasurable states with a given initial state vector X_0 and covariance matrix P_0.

IV. Extended Kalman Filter Applied to the SEIQRD Model for State Estimation

The system state vector X for the SEIQRD model with reinfection (1) is defined as:

$$X = \begin{bmatrix} S & E & I & Q & R & D \end{bmatrix}^T. \tag{4}$$

Starting with the non-negative initial conditions:

$$X(t_0) = [S_0, E_0, I_0, Q_0, R_0, D_0], \tag{5}$$

which yields the state space model described as:

$$X_{t+1} = f(X_t) + \xi_t, \tag{6}$$

where f is the nonlinear function, ξ_t is the process noise that is assumed to be Gaussian with zero mean and covariance matrix Ξ. Now, the discrete time nonlinear function $f(X_t)$ can be represented as:

$$f(X_t) = \begin{bmatrix} -\beta IS + \alpha R \\ \beta IS - \epsilon E \\ \epsilon E - \gamma I - qI - dI \\ qI - q_t Q - dQ \\ \gamma I + q_t Q - \alpha R \\ d(N - (S + E + R + D)) \end{bmatrix}. \tag{7}$$

Back to Contents

Using the Taylor series approximation to linearize the nonlinear discrete time system in (7) as a linear system we get:

$$F_t = \begin{bmatrix} -\beta I & 0 & -\beta S & 0 & \alpha & 0 \\ \beta I & -\epsilon & -\beta S & 0 & 0 & 0 \\ 0 & \epsilon & (-\gamma - q - d) & 0 & 0 & 0 \\ 0 & 0 & q & (-q_t - d) & 0 & 0 \\ 0 & 0 & \gamma & q_t & -\alpha & 0 \\ -d & -d & 0 & 0 & -d & -d \end{bmatrix}, \tag{8}$$

where F_t is the Jacobian matrix. In the available dataset of reported cases in Saudi Arabia, we use the measurements of the active cases (I) and cumulative death (D) by incorporating them within the model using the measurement equation as:

$$y_t = HX_t + \omega_t, \tag{9}$$

where, y_t is the measurement vector of the observed data and H is the observation matrix structured as:

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{10}$$

and ω_t is the measurement noise assumed to be Gaussian distributed with zero mean and covariance matrix Ω. The EKF algorithm for state estimation involving both measured and unmeasured states can be described as the following steps:

1) Start with initializing the state vector X_0 and the covariance matrix P_0 as:

$$\begin{aligned} \hat{X_0}^+ &= E[X_0], \\ \hat{P_0}^+ &= E[(X_0 - \hat{X_0}^+)(X_0 - \hat{X_0}^+)^T]. \end{aligned} \tag{11}$$

2) Perform the prediction of state estimates and error covariance as:

$$\begin{aligned} \hat{X}_t^- &= f(\hat{X}_t), \\ P_t^- &= FP_t^+F^T + \Xi_t. \end{aligned} \tag{12}$$

3) Perform the measurement update of the state estimate and estimation error covariance as:

$$\hat{X}_t^+ = \hat{X}_t^- + K_t\left(y_t - H_t\left(\hat{X}_t^-\right)\right), \tag{13}$$

$$K_t = P_t^- H_t^T \left(H_t P_t^- H_t^T + \Omega_t\right)^{-1}, \tag{14}$$

$$P_t^+ = \left(I - K_t H_t\right) P_t^-, \tag{15}$$

where K_t is the Kalman gain and $\hat{+}$ denotes the estimate after processing the measurement whereas $^-$ denotes the process before the correction step.

4) The steps in 2 and 3 can be repeated until getting a better estimate of X_t.

V. Parameter Estimation and Simulation of COVID-19 Spread in Saudi Arabia

We analysed the Saudi Arabia COVID-19 data with active cases (I) and death cases (D) based on the openly available dataset in [20]. The dataset contains daily measurements of these two state variables between the dates 15 February 2020 and 17 March 2022. This dataset contains data about the numbers of tests, cases, deaths, critical cases, active cases and recovered cases in each country. We next used the nested sampling algorithm with Markov Chain Monte Carlo (MCMC) random walk for the live points (N_{live}) to draw samples from the likelihood surface as presented in [19]. The nested sampling is a generic Bayesian inference framework to estimate unknown model parameters along with uncertainty bounds from their posterior probability distribution while also calculating the marginal likelihood or Bayesian evidence (log$Z$) of the model showing the degree of agreement between the model and the measured data. The uncertainty information on the SEIQRD model parameters within the one standard deviation around the mean ($\mu \pm \sigma$) confidence interval (CI) has been shown in Fig. 2. Using the Saudi Arabia COVID-19 data, the estimated model parameters as the mean of the posterior are presented in Table I. The tuning parameters in the nested sampling algorithm to fit the SEIQRD model include: (a) the number of live points $N_{live} = 90$ for exploring the 9D joint posterior distribution of the unknown model parameters, (b) stopping criterion for log-evidence calculation Δlog$Z = 0.01$. As an output we get: (a) total number of likelihood evaluations $N_{like} = 9792$ proportional to N_{live}, (b) model evidence log Z = -7236.54. In the Bayesian inference engine, we used an uninformative prior as a uniform distribution over a specified range of unknown model parameters i.e. $\{\beta, \epsilon, \gamma, d, q, q_t, S_0, E_0, \alpha\}$ as:

$$\begin{gathered} \pi(\beta) \sim \mathcal{U}\left[0, 10^{-4}\right], \quad \pi(\epsilon) \sim \mathcal{U}[0,1], \\ \pi(\gamma) \sim \mathcal{U}[0,1], \quad \pi(d) \sim \mathcal{U}\left[0,1\right], \\ \pi(q) \sim \mathcal{U}\left[0,1\right], \quad \pi(q_t) \sim \mathcal{U}\left[0,1\right], \\ \pi(S_0) \sim \mathcal{U}\left[1 \times 10^5, 1.5 \times 10^9\right], \\ \pi(E_0) \sim \mathcal{U}\left[1 \times 10^5, 1.5 \times 10^9\right], \\ \pi(\alpha) \sim \mathcal{U}\left[0,1\right]. \end{gathered} \tag{16}$$

Using the above prior and a multivariate Gaussian likelihood function assuming the temporal data-points are independent and identically distributed (i.i.d) samples, the nested sampling algorithm draws random samples from the posterior of the unknown SEIQRD model parameters. Fig. 3 shows the posterior distribution of all the parameters of the proposed SEIQRD model with reinfection term using the univariate marginal histograms in the principal diagonal and the bivariate kernel density estimates (KDE) along with scatterplots of the posterior samples in the off-diagonals. Simulation results was conducted using the mean of the posterior prediction to predict the COVID-19 spread in Saudi Arabia as shown in Fig 4. It can be seen from Fig. 4(a) around 150th day of the simulation the infected reported cases reached it peak with 63000 infections

Back to Contents

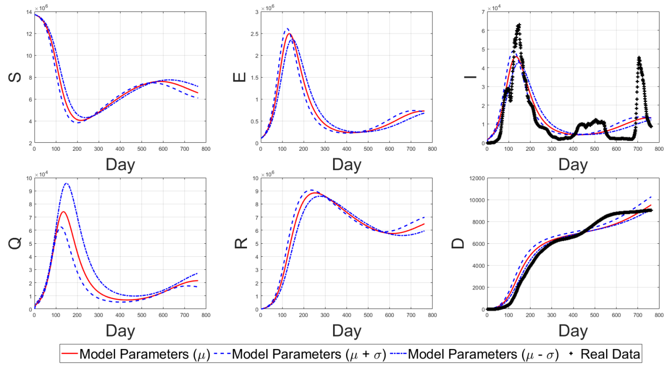


Fig. 2: The estimated uncertainty bounds of the posterior distribution of the model parameters.

TABLE I: Mean Posterior of the proposed SEIQRD Model Parameters for the Saudi Arabia COVID-19 Data

Parameter	Value	Description
N	34 $\times 10^6$	population number
β	2.92 $\times 10^{-7}$	infection rate
α	0.0028	reinfection rate
ϵ	0.0353	1/incubation period
q	0.9593	quarantine rate
qt	0.5939	time period of quarantine
d	3.5853$\times 10^{-4}$	death rate
γ	0.9586	recovery rate
R_0	5.44	Reproductive number

TABLE II: Root Mean Square Error of the EKF-based on SEIQRD Model

Covariance Matrices	Infected Error	Death Error
Ξ = 1 , Ω = diag[10, 10]	196.442	0.74698
Ξ = 0.01 , Ω = diag[10 10000]	212.3171	42.2808
Ξ = 500 , Ω = diag[100, 1000]	56.3065	0.27098

and it is noted that the total death in 4(b) in the same period is gradually increasing over time. After the first peak,the number of active cases decreased rapidly which clearly indicates that the Saudi Government has effectively controlled the pandemic using various measures such as lockdown, self-isolation and social distancing which were strictly applied.

For the full period in this study the basic reproduction number R_0 is calculated according to the observed data and estimated around 5.44 for the whole size of the epidemic since the outbreak. In the EKF simulations, several values of the covariances Ξ and Ω were tested and each values has a different performance where we observe that if the Ξ and Ω are small we get a large error in the EKF estimates. This helped deciding to use the process and measurement noise covariance matrices as: $\Xi = 500 \times I_{6\times 6}$, $\Omega = diag([100, 1000])$ where we obtained small error in both the infected and death situations which is the best estimate in these cases. Thus, as shown in Fig. 4(a), the measured data of active cases lie very close to the EKF predictions where the mean posterior numerical simulation as the smooth output of the SEIQRD model alone does not perform so well in explaining the non-smooth changes in the active cases. Fig. 4(b) shows the estimated cumulative number of death cases which is around 9000 in Saudi Arabia since the early intervention helped to reduce the mortality rate. The EKF predictions for I and D are very close to the reported data and better than the posterior mean simulations of the SEIQRD model. Each subplot in Fig. 5 corresponds to the simulation of the proposed SEIQRD model and the predictions based on the EKF for the unmeasurable states. Due to the unavailability of the ground truth data for these 4 unmeasurable states, the EKF prediction is more reliable than the smooth dynamical model simulations. However, we notice that the dynamical system simulation model is close to the EKF prediction results in the susceptible cases and recovered cases while for the remaining variables viz. exposed and quarantined cases, the model differs slightly in terms of when they reach their peaks. Thus, the use of the EKF helps in estimating the unmeasurable states such as susceptible, exposed, quarantined and recovered more accurately than the nonlinear system model simulations since it can predict sharp changes while the ODE simulation alone is mostly smooth in nature. In order to quantify the EKF prediction results, we use the root mean square error (RMSE) of the active and deaths cases defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{N}(X_{\text{reported } I,D} - X_{\text{EKF},I,D})^2}, \tag{17}$$

where, n is the number of measurement points. The RMSE values were compared in Table II, for the two state variables I and D using different assumptions of the noise covariance matrices. This demonstrates the validity and efficacy of the proposed nonlinear state estimation method beside visual comparisons of the real COVID-19 data with the EKF-based predictions.

VI. Conclusion

This paper presents a new epidemiological model of the SEIQRD form with reinfection to understand the impact of COVID-19 based on active and death cases data in Saudi Arabia. Nested sampling algorithm based posterior mean parameters were used in the SEIQRD model for dynamic simulations. The fitted dynamic model can be useful to predict the spread of infectious disease and can be further used to help the Saudi Government to monitor the COVID-19 pandemic since different scenarios of unknown bias/noise covariance have been considered. The EKF was applied with the linearized version of the SEIQRD model to estimate the dynamics of COVID-19 unmeasurable states while also validating the predictions with the actual measurements of the active cases and the cumulative deaths. Our results show that the EKF is capable of estimating the evolution of the pandemic in the long term which yields more accurate estimation than the fitted nonlinear dynamical system model. In the future, we shall consider more complex epidemiological models and other families of the nonlinear Kalman filters with different assumptions of the noise distribution beside the normal case.

Back to Contents

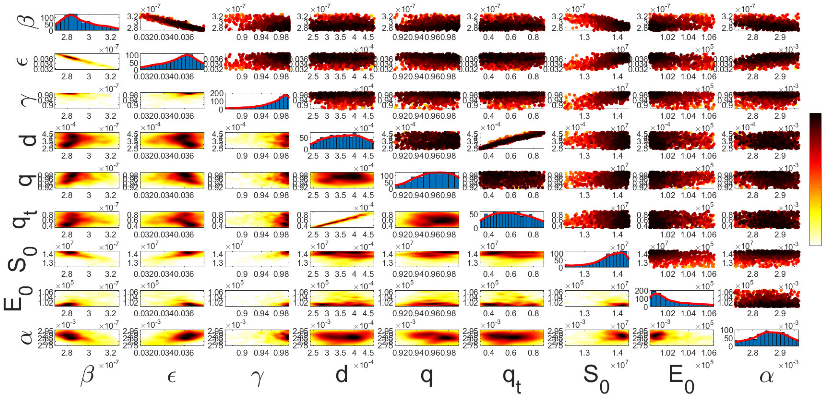


Fig. 3: Posterior distribution of the proposed SEIQRD model parameters with the reinfection term.

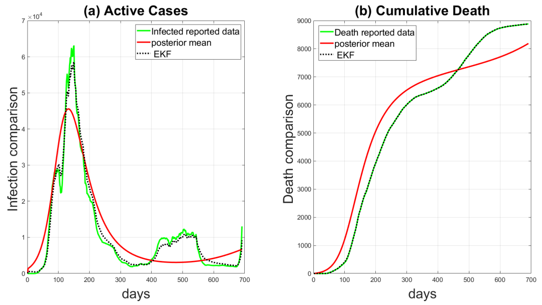


Fig. 4: Comparison of the state estimation based on EKF with real data in Saudi Arabia: (a) active cases (b) cumulative death.

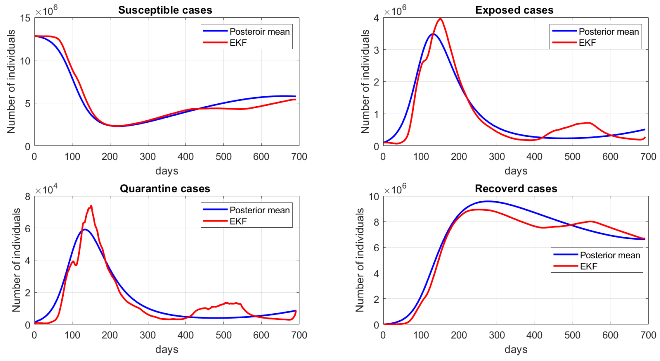


Fig. 5: Comparing estimated variables of the Susceptible, Exposed, Quarantined and Recovered cases in Saudi Arabia.

References

[1] C. Sohrabi, Z. Alsafi, N. O'neill, M. Khan, A. Kerwan, A. Al-Jabir, C. Iosifidis, and R. Agha, "World health organization declares global emergency: A review of the 2019 novel coronavirus (covid-19)," *International Journal of Surgery*, vol. 76, pp. 71–76, 2020.

[2] W. O. Kermack and A. G. McKendrick, "A contribution to the mathematical theory of epidemics," *Proceedings of the Royal Society of London. Series A*, vol. 115, no. 772, pp. 700–721, 1927.

[3] J. Guan, Y. Zhao, Y. Wei, S. Shen, D. You, R. Zhang, T. Lange, and F. Chen, "Transmission dynamics model and the coronavirus disease 2019 epidemic: applications and challenges," *Medical Review*, vol. 2, no. 1, pp. 89–109, 2022.

[4] P. Birrell, J. Blake, E. Van Leeuwen, N. Gent, and D. De Angelis, "Real-time nowcasting and forecasting of covid-19 dynamics in england: the first wave," *Philosophical Transactions of the Royal Society B*, vol. 376, no. 1829, p. 20200279, 2021.

[5] H. M. Youssef, N. A. Alghamdi, M. A. Ezzat, A. A. El-Bary, and A. M. Shawky, "A modified seir model applied to the data of covid-19 spread in saudi arabia," *AIP advances*, vol. 10, no. 12, p. 125210, 2020.

[6] H. M. Youssef, N. Alghamdi, M. A. Ezzat, A. A. El-Bary, and A. M. Shawky, "A proposed modified seiqr epidemic model to analyze the covid-19 spreading in saudi arabia," *Alexandria Engineering Journal*, vol. 61, no. 3, pp. 2456–2470, 2022.

[7] M. Alauddin, M. A. I. Khan, F. Khan, S. Imtiaz, S. Ahmed, and P. Amyotte, "How can process safety and a risk management approach guide pandemic risk management?" *Journal of Loss Prevention in the Process Industries*, vol. 68, p. 104310, 2020.

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[9] A. Jazwinski, "Stochastic processes and filtering theory. vol. 64 of math," *Science and Engineering. Academic Press. New York*, 1970.

[10] D. Simon, *Optimal state estimation: Kalman, H_∞, and nonlinear approaches*. John Wiley & Sons, 2006.

[11] B. Cazelles and N. Chau, "Using the kalman filter and dynamic models to assess the changing hiv/aids epidemic," *Mathematical Biosciences*, vol. 140, no. 2, pp. 131–154, 1997.

[12] D. Ndanguza, I. S. Mbalawata, H. Haario, and J. M. Tchuenche, "Analysis of bias in an ebola epidemic model by extended kalman filter approach," *Mathematics and Computers in Simulation*, vol. 142, pp. 113–129, 2017.

[13] J. Song, H. Xie, B. Gao, Y. Zhong, C. Gu, and K.-S. Choi, "Maximum likelihood-based extended kalman filter for covid-19 prediction," *Chaos, Solitons & Fractals*, vol. 146, p. 110922, 2021.

[14] X. Zhu, B. Gao, Y. Zhong, C. Gu, and K.-S. Choi, "Extended kalman filter based on stochastic epidemiological model for covid-19 modelling," *Computers in Biology and Medicine*, vol. 137, p. 104810, 2021.

[15] K. K. Singh, S. Kumar, P. Dixit, and M. K. Bajpai, "Kalman filter based short term prediction model for covid-19 spread," *Applied Intelligence*, vol. 51, no. 5, pp. 2714–2726, 2021.

[16] A. S. Breathnach, P. A. Riley, M. P. Cotter, A. C. Houston, M. S. Habibi, and T. D. Planche, "Prior covid-19 significantly reduces the risk of subsequent infection, but reinfections are seen after eight months," *Journal of Infection*, vol. 82, no. 4, pp. e11–e12, 2021.

[17] K. Chatterjee, K. Chatterjee, A. Kumar, and S. Shankar, "Healthcare impact of covid-19 epidemic in india: A stochastic mathematical model," *Medical Journal Armed Forces India*, vol. 76, no. 2, pp. 147–155, 2020.

[18] O. Diekmann, J. Heesterbeek, and M. G. Roberts, "The construction of next-generation matrices for compartmental epidemic models," *Journal of the Royal Society Interface*, vol. 7, no. 47, pp. 873–885, 2010.

[19] J. Skilling, "Nested sampling for general bayesian computation," *Bayesian Analysis*, vol. 1, no. 4, pp. 833–859, 2006.

[20] Worldometer. (2022) Covid-19 worldometer daily snapshots. [Online]. Available: https://www.kaggle.com/datasets/selfishgene/covid19-worldometer-snapshots-since-april-18

Optimal Bernoulli point estimation with applications

Alexey Narykov, Murat Üney, Jason F. Ralph
Dept. of Electrical Engineering and Electronics
University of Liverpool, L69 3BX, Liverpool, UK
Emails: {a.narykov, m.uney, jfralph}@liverpool.ac.uk

***Abstract*—This paper develops optimal procedures for point estimation with Bernoulli filters. These filters are of interest to radar and sonar surveillance because they are designed for stochastic targets that can enter and exit the surveillance region at random instances. Because of this property they are not served by the minimum mean square estimator, which is the most widely used approach to optimal point estimation. Instead of the squared error loss, this paper proposes an application-oriented loss function that is compatible with Bernoulli filters, and it develops two significant practical estimators: the minimum probability of error estimate (which is based on the rule of ideal observer), and the minimum mean operational loss estimate (which models a simple defence scenario).**

I. Introduction

Radar and sonar processing chains often use a Bayesian filter that outputs a probability distribution describing the state of a time-varying stochastic world. Such a probabilistic representation is unintelligible in many practical applications and to human decision makers alike. More interpretable results are obtained by collapsing the full distribution into the best possible estimate (called an *optimal point estimate*), which is then used by the dependent application as if it were the true state of the world. The best estimate, from the perspective of Bayesian decision theory, is the one which minimises the expected amount of loss in the application. This loss emerges due to the discrepancy between the revealed true state of the world and its estimate, and is typically quantified by the squared error (SE) (loss) function (shown on Fig. 1). This function leads to the minimum mean SE (MMSE) estimate, which happens to coincide with the expected value of the random variable, and is often easy to compute.

This paper studies optimal point estimation for Bernoulli filters [1]. These filters are designed for stochastic dynamic systems that randomly switch *on* and *off* and of interest to radar and sonar surveillance [2], [3] where the target of interest may not always exist in the surveillance region. The MMSE estimator is known to be incompatible with such random finite set filters [4] since in the SE loss the underlying definition of error is based on the Euclidean distance, which does not extend to the cardinality errors, i.e., errors in the number of targets. In Bernoulli filtering, this latter errors are the equivalents of false alarm and missed detections.

Nevertheless, there have been efforts to adopt the SE loss regardless. Some authors have proposed using alternative set distance definitions (such as the optimal subpattern assignment (OSPA) distance), which combine errors in location and cardinality after redefining the SE loss [5], [6] (see also [7]). To the best of our knowledge, these approaches have not reached widespread use (see, e.g., [8], [9] for relevant developments). One difficulty for their use is that the resulting estimates are not as easy to compute as other sub-optimal ones. For example, a typical alternative is to test the target's probability of existence against a pre-defined threshold, and, only if it exceeds, use the SE loss to extract an estimate of the kinematic state from the localisation density [10], [11], [12]. This approach is ad hoc in its nature as there is no criterion to uniquely select the threshold, and the resulting estimate is not endowed with properties of optimality in some prespecified sense.

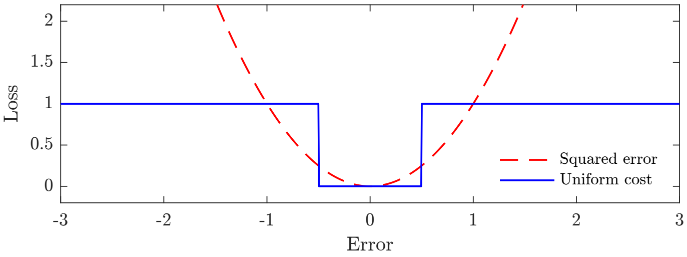


Fig. 1. The SE loss (solid) and the UC loss [13], [14] (dashed) with tolerance r_0 ($r_0 = 0.5$ shown). These losses are symmetric, i.e., they equally penalise errors of over- and underestimation. Note that only the UC loss is bounded.

This paper proposes a loss function that is directly compatible with Bernoulli filters. The proposed approach can be configured to model losses in different applications. In particular, the loss function is constructed to integrate the loss resulting from the error in the target's kinematic state (quantified with the uniform cost (UC) loss shown on Fig. 1) and the loss due to the error in cardinality. The approach is validated with two examples, which are irreducible to each other, and yield practical optimal point estimates:

- the minimum probability of error (MPE) estimate, and
- the minimum mean operational loss (MMOL) estimate.

To the best of our knowledge, precursors of our approach combine other loss functions, and do that in different contexts, such as joint signal detection and estimation [15, Ch.6], [16], or joint tracking of a target and its classification [17].

The developments in this paper are for a single Bernoulli variable. Technically, the resulting expressions can be applied to individual Bernoulli components of a multi-Bernoulli process output by a multi-object filter (e.g., [12], [18], [19]). However, investigation of optimality in, and extension of, our optimal estimation approach to multi-Bernoulli systems is the subject of future work.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

The paper is organised as follows. Section II introduces a Bernoulli point process, which models a stochastic target, and outlines the procedure of Bayes-optimal point estimation. Section III proposes an application-oriented loss function, and develops an optimal point estimator that thresholds the probability of existence to declare a target. Section IV develops two practical estimators and studies their thresholds.

II. Background

A. Bernoulli point process

In this article, the objects of interest, i.e., the targets, have individual states x in some d_x-dimensional state space $\mathcal{X} \subset \mathbb{R}^{d_x}$, typically consisting of position, velocity and class variables. A point process (p.p.) Φ on $\mathcal{X}$ is a random variable on the process space $\mathfrak{X} = \bigcup_{n=0}^{\infty} \mathcal{X}^n$, i.e. the space of all finite sequences of points in $\mathcal{X}$, whose number of elements *and* element states are unknown and (possibly) time-varying. A realisation of Φ is a sequence $x_{1:n} \in \mathcal{X}^n$, representing a *population* of n objects with states $x_i \in \mathcal{X}$, $1 \leq i \leq n$, where $n \in \mathbb{N}$. A more formal definition can be found in [20]. In the context of Bayesian filtering, this sequence depicts a specific multi-object configuration.

As for regular real-valued random variables, a p.p. is described by its probability distribution P_Φ on $\mathfrak{X}$; the projection measure $P_\Phi^{(n)}$ describes the realisations of Φ with n elements, $n \geq 0$. The projection measures are assumed to be symmetrical functions, so that the order of points in a realisation is irrelevant for statistical purposes and the permutations of a realization of the p.p.—such as (x_1, x_2) and (x_2, x_1)—are equally probable. In addition, a p.p. is called *simple* if the probability distribution is such that realisations are sequences of points that are pairwise distinct almost surely, i.e., a realization does not contain repetitions. For the rest of the paper, all of the point processes are assumed to be simple. The density of the projection measure $P_\Phi^{(n)}$, $n \geq 0$, is then denoted by $p_\Phi^{(n)}$.

Definition II.1 (Bernoulli point process [21], [12])**.** *A Bernoulli point process Φ on $\mathcal{X}$ with parameter $0 \leq p \leq 1$ and spatial distribution s is an i.i.d. cluster process with spatial distribution s, whose size is 1 with probability p and 0 with probability $1-p$. Its probability density is given by:*

$$p_\Phi^{(n)}(\varphi) = \begin{cases} 1-p, & \textit{if } \varphi = \emptyset, \\ p \cdot s(x), & \textit{if } \varphi = \{x\}, , \\ 0, & \textit{otherwise}. \end{cases} \tag{1}$$

where $n = |\varphi|$ is the set cardinality, and $\emptyset$ is the empty set. Its probability generating functional (p.g.fl.) is given by[1]

$$\mathcal{G}_\Phi[h] = 1 - p + p \int h(x) s(\mathrm{d}x), \tag{2}$$

where $h : \mathcal{X} \to [0, 1]$ is a test function.

In the context of target tracking, the parameter p is typically referred to as the target's *probability of existence*.

B. Bayes-optimal point estimation

In the Bayesian framework, the optimal solution to a point estimation problem is obtained following the minimum expected loss principle [22], [23], where a loss function

$$L : \mathfrak{X} \times \mathfrak{X} \to \mathbb{R}_0^+ \tag{3}$$

assigns a non-negative real number to every possible pair of an estimate and the true state on the state space $\mathfrak{X}$.

Proposition II.2 (Optimal Bernoulli point estimation)**.** *For a Bernoulli p.p. Φ from Definition II.1, the solution to the optimal point estimation problem is a pair $(\alpha_\Phi^*, \rho_\Phi^*)$ of, respectively, the optimal estimate and associated expected loss*

$$\alpha_\Phi^* = \underset{\alpha \in \mathfrak{X}}{\arg\min}\, \mathbb{E}\left[L(\alpha, \Phi)\right], \tag{4}$$

$$\rho_\Phi^* = \mathbb{E}\left[L(\alpha_\Phi^*, \Phi)\right], \tag{5}$$

where L is in (3)*, and its expected value for some $\alpha \in \mathfrak{X}$ is*

$$\mathbb{E}\left[L(\alpha, \Phi)\right] = \sum_{n \geq 0} \int L(\alpha, \varphi) P_\Phi^{(n)}(\mathrm{d}x_{1:n}) \tag{6a}$$

$$= p_\Phi^{(0)}(\emptyset) L(\alpha, \emptyset) + \int L(\alpha, \{x\}) p_\Phi^{(1)}(x) \mathrm{d}x + \sum_{n \geq 2} \int L(\alpha, \{x_{1:n}\}) p_\Phi^{(n)}(x_{1:n}) \mathrm{d}x_{1:n} \tag{6b}$$

$$= (1-p) L(\alpha, \emptyset) + p \int L(\alpha, \{x\}) s(\mathrm{d}x). \tag{6c}$$

III. Application-oriented point estimation

A. Proposed loss function

We propose an estimation loss compatible with Bernoulli p.p. that, as will be shown in Section IV, can be configured to model loss in particular applications.

Definition III.1 (Application loss)**.** *The loss function is*

$$L(\alpha, \varphi) := \begin{cases} c_{00}, & \textit{if } \alpha = \emptyset, \varphi = \emptyset, \\ c_{01}, & \textit{if } \alpha = \emptyset, \varphi = \{x\}, \\ c_{10}, & \textit{if } \alpha = \{a\}, \varphi = \emptyset, \\ c_{11} + c \cdot \mathbb{1}_{B_a}(x), & \textit{if } \alpha = \{a\}, \varphi = \{x\}. \end{cases}, \tag{7}$$

where $c_{00}, c_{01}, c_{10}, c_{11}, c \in \mathbb{R}^+$, and $\mathbb{1}_{B_a}$ is the indicator function on a region $B_a \subset \mathcal{X}$ such that

$$\mathbb{1}_{B_a}(x) := \begin{cases} 1, & \textit{if } x \in B_a, \\ 0, & \textit{if } x \notin B_a. \end{cases}, \tag{8}$$

where B_a is the rejection *region*[2] *(for Euclidean distance d):*

$$B_a := \{x \mid d(a, x) > r_0\}, \; \forall\, x \in \mathcal{X}. \tag{9}$$

The proposed loss function is a combination of a set of coefficients $\{c_{00}, c_{01}, c_{10}, c_{11}\}$ and the loss in (8). The set encodes a cost matrix (hence the subscripts), which is essentially a loss function on the state space comprising just two points [24, Ch. 8.11]; (8) is effectively the UC loss function (Fig. 1) [13],

[1] Here and in the following, notation $s(\mathrm{d}x) = s(x)\mathrm{d}x$ is used for the sake of compactness.

[2] We note that here, B_a does *not* denote the ball of radius r_0 around a, but rather it indicates the complement of the ball in $\mathcal{X}$.

Back to Contents

TABLE I
COST ASSIGNMENT IN (7) FOR THE MPE AND MMOL ESTIMATORS.

Estimator	c_{00}	c_{01}	c_{10}	c_{11}	c
MPE	0	1	1	0	1
MMOL	0	c_A	c_M	c_M	c_A

[14]. The UC loss assigns cost 1 to every pair of a and x with distance between them higher than the tolerance parameter r_0, and 0 otherwise. In principle, the UC loss models effectors with a limited impact region, e.g., pencil-beam radars, low-power sensing nodes in a network [25], [26], [27], precise defensive countermeasures [28], [29], or rescue supplies [30, p. 40].

Remark III.2. *The loss in* (7) *and the squared OSPA error loss (discussed in the introduction) are distinct and not reducible to each other as, in the latter,* c_{00} *and* c_{11} *are set to* 0*,* c_{01} *and* c_{10} *to* c^2*, and,* $c \cdot \mathbb{1}_{B_a}(x)$ *is replaced by* $\min(c, d(a,x))^2$ *for each* $a, x \in \mathcal{X}$*, where* $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_0^+$ *is the Euclidean or other distance metric.*

B. Bayes-optimal point estimation

Theorem III.3 (Bayes-optimal Bernoulli point estimation)**.** *For a Bernoulli p.p.* Φ *with parameter* p *and spatial distribution* s*, a Bayes-optimal solution to the point estimation problem under loss* (7) *is a pair* (α^*, ρ^*)*, respectively, of the optimal point estimate and associated expected loss given by*

$$(\alpha^*, \rho^*) = \begin{cases} (\{a^*_{\text{MMUC}}\}, \rho_{\{a^*_{\text{MMUC}}\}}), & \text{if } p > \Gamma, \\ (\emptyset, \rho_\emptyset), & \text{if } p < \Gamma. \end{cases}, \tag{10}$$

where Γ *is the reporting threshold obtained as*

$$\Gamma = \frac{c_{00} - c_{10}}{c_{00} - c_{01} - c_{10} + c_{11} + c \int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x) s(\mathrm{d}x)}, \tag{11}$$

a^*_{MMUC} *is the minimum mean uniform cost (MMUC) estimate*

$$a^*_{\text{MMUC}} = \arg\min_{a \in \mathcal{X}} \int \mathbb{1}_{B_a}(x) s(\mathrm{d}x), \tag{12}$$

with its corresponding expected loss

$$\rho_{\{a^*_{\text{MMUC}}\}} = (1-p)c_{10} + p\left[c_{11} + c\int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x) s(\mathrm{d}x)\right]. \tag{13}$$

In (10)*, the expected loss of the empty set* $\emptyset$ *is given by*

$$\rho_\emptyset = (1-p)c_{00} + pc_{01}. \tag{14}$$

The proof is given in Appendix. This estimator is a test over the Bernoulli parameter p in (1) against a threshold Γ. There are sub-optimal procedures [10], [11], [12] with a similar structure. Our approach differs in that the threshold Γ optimally adapts to the spatial distribution s and the loss function (7) that models the application at hand (cf. the estimator in [31, Ch. 14.7.5.2]).

Estimator in (10) requires solving (12) which can be computationally expensive. From Sherman's theorem [32], if s is unimodal and symmetric around its mean, the optimal estimate in (12) is the mean of s, i.e., $\int xs(\mathrm{d}x)$, which is easier to compute. If s is multimodal, using the SE loss, i.e., the mean estimate, is discouraged in practice [33], [34], and an estimator based on a bounded loss function, e.g., (8), is preferable [29].

Finally, since s and p are common both to Bernoulli filters and the integrated probabilistic data association (IPDA) filter [35], [36], this estimator is compatible with both algorithms.

IV. APPLICATION-SPECIFIC POINT ESTIMATES

This section develops two examples of application-specific estimators that are based on the proposed loss function in (7) and use the cost relations in Table I. We study the estimators for Bernoulli-Gaussian processes, i.e., Bernoullis with $s(\cdot) = \mathcal{N}(\cdot; \mu, \sigma^2)$, where μ and σ are, respectively, the mean and standard deviation. The focus is primarily on the behaviour of Γ, which tests p to determine whether the empty set $\emptyset$ or a singleton $\{\mu\}$ should be reported.

A. Minimum probability of error estimate

Cost assignment in this estimator is inspired by the MPE decision rule in detection theory [37, p. 8], which is sometimes called *the rule of ideal observer* [15, p. 51] or *the Siegert-Kotelnikov rule* [38, p. 65]. It assigns the costs such that correct decisions incur no penalties, and incorrect decisions are penalised with the unit cost. Such assignment is compatible with the UC loss function within loss (7) when costs from Table I are used, and provided that correct detection is penalised if the true target kinematic state falls inside the rejection region.

Corollary IV.1 (Minimum probability of error estimation)**.** *Under the MPE cost assignment from Table I, the MPE estimator is a pair* $(\alpha^*_{\text{MPE}}, \rho^*_{\text{MPE}})$ *that is obtained from* $(\alpha^*_\Phi, \rho^*_\Phi)$ *in Theorem III.3 with*

$$\Gamma = \left[2 - \int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x) s(\mathrm{d}x)\right]^{-1}, \tag{15}$$

$$\rho_{\{a^*_{\text{MMUC}}\}} = \mathcal{G}_\Phi\left[\mathbb{1}_{B_{a^*_{\text{MMUC}}}}\right], \tag{16}$$

$$\rho_\emptyset = p, \tag{17}$$

where $\mathcal{G}_\Phi[\cdot]$ *is defined in* (2)*.*

Proof. The result is obtained by substituting the MPE costs from Table I into Theorem III.3. For (13), this leads to

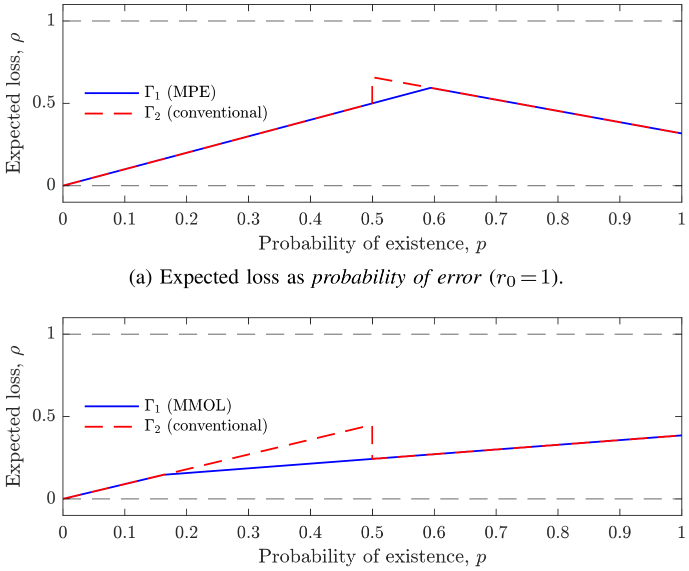
$$\rho_{\{a^*_{\text{MMUC}}\}} = 1 - p + p\int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x) s(\mathrm{d}x), \tag{18}$$

which is equivalent to (16) when notations (2) are used. □

The result in (16) highlights the utility of p.g.fl.s in practical applications, beyond filtering derivations (e.g., [21]). Another example is the statistics of adversarial risk in [39, Thm. IV.2].

Fig. 2a compares the quality of MPE and conventional estimates that are produced, respectively, using $\Gamma_1 = 0.5943$ and $\Gamma_2 = 0.5$. The MPE threshold yields estimates with lower probability of error for Bernoullis with $p \in [\Gamma_2, \Gamma_1]$. The MPE threshold is further studied on Fig. 3a: Bernoullis with higher spatial uncertainty require higher thresholds for a target to be declared. A Bernoulli with $p < 0.5$ is never declared as a target (i.e., the threshold values are bounded from below), whereas when $p > 0.5$ it may be estimated as no target in case the uncertainty is high with respect to r_0.

Back to Contents



(a) Expected loss as *probability of error* ($r_0 = 1$).

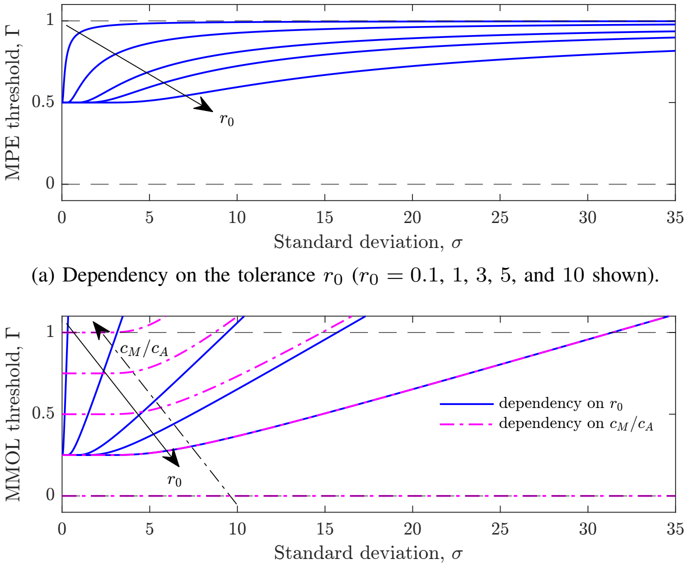
(b) Expected loss as *mean operational loss* ($r_0 = 1$, $c_A = 0.9$, $c_M = 0.1$).

Fig. 2. The quality of point estimates as perceived by the respective application, which are produced with optimal (Γ_1) and conventional ($\Gamma_2 = 0.5$) thresholds. The estimates are of Bernoulli-Gaussians with distinct p values, $0 \leq p \leq 1$, and spatial distributions with the same $\mu = 0$ and $\sigma = 1$. The quality is naturally quantified by the expected loss in the application.

B. Minimum mean operational loss estimate

Cost assignment in the MMOL estimator is inspired by a textbook example of decision making under uncertainty, typically called the umbrella problem [40, p. 24] or cost-loss model [41]. This model is used to determine how the probability of adverse events affects the decision of whether to take a costly precautionary measure for protection against losses from that event. We consider an operational scenario with one potential target aiming to destroy the asset of cost c_A (see, e.g., [39]), when we control a countermeasure of cost c_M. What distinguishes it from the rule of ideal observer is that cardinality errors are penalised in different ways (see Table I): $c_{01} \neq c_{10}$ (losing the asset is commonly more damaging than wasting the countermeasure), and $c_{11} > 0$ (countermeasure is committed to prevent losing the asset). We extend this model within loss (7) by considering that the countermeasure has limited impact around the point of its application with radius r_0: failure to apply it sufficiently close to the target is then modelled by the UC loss in (8), and is penalised by both c_A and c_M.

Although the original model is designed for studying decisions about what course of action is to implement, it permits another interpretation, see e.g. [42], that communicates a statement about the state of stochastic world. For example, if the optimal action is to preserve the countermeasure, it is equivalent to acting as if *there were no target*. And similarly, applying the countermeasure in a certain location is equivalent to acting as if *there were a target in that point*.



(a) Dependency on the tolerance r_0 ($r_0 = 0.1$, 1, 3, 5, and 10 shown).

(b) Dependencies on r_0 for $c_M/c_A = 0.25$ ($r_0 = 0.1$, 1, 3, 5, and 10 shown), and on c_M/c_A for $r_0 = 10$ ($c_M/c_A = 0$, 0.25, 0.5, 0.75, and 1 shown).

Fig. 3. Reporting threshold Γ as a function of spatial uncertainty, which is characterized by the standard deviation σ in the Bernoulli-Gaussian case.

Corollary IV.2 (Minimum mean operational loss estimation)**.** *Under the MMOL cost assignment from Table I, the MMOL estimator is a pair $(\alpha^*_{\text{MMOL}}, \rho^*_{\text{MMOL}})$ that is obtained from $(\alpha^*_{\Phi}, \rho^*_{\Phi})$ in Theorem III.3 with*

$$\Gamma = \frac{c_M}{c_A}\left[1 - \int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x)s(\mathrm{d}x)\right]^{-1} \tag{19}$$

$$\rho_{\{a^*_{\text{MMUC}}\}} = c_M + p \cdot c_A \cdot \int \mathbb{1}_{B_{a^*_{\text{MMUC}}}}(x)s(\mathrm{d}x), \tag{20}$$

$$\rho_{\emptyset} = p \cdot c_A. \tag{21}$$

The result is obtained by substituting the MMOL costs from Table I into Theorem III.3. Fig. 2b compares the quality of the MMOL and conventional estimates, which are produced, respectively, with $\Gamma_1 = 0.1628$ and $\Gamma_2 = 0.5$. The MMOL threshold yields lower mean operational loss for Bernoullis with $p \in [\Gamma_1, \Gamma_2]$. The MMOL threshold is studied on Fig. 3b: it is bounded from below by c_M/c_A, which can generally be smaller than 0.5. When the threshold is studied for various c_M/c_A (dash-dotted), it reveals its characteristic behaviour: if $c_M = 0$, the target is always declared as there is no cost of committing the countermeasure; if $c_M \geq c_A$, the target is never declared since it is always better to preserve a costly countermeasure.

V. Conclusion

In this paper we have proposed an application-oriented loss function for Bernoulli filters, and developed two examples of optimal application-specific point estimators. Similar to the conventional estimators, they involve the step of thresholding of the target's probability of existence. However, this threshold is not a constant, but a function of specific parameterization in the loss function as well as certain features of the spatial probability density. A critical difference of the resulting estimators is that a Bernoulli with high probability of existence may still

Back to Contents

be declared as absent if the spatial uncertainty is high, or if committing costly measures brings unjustified expected losses.

Appendix: Proof of Theorem III.3

Proof. Let us first obtain expressions of the expected loss $\rho_\emptyset$ for the empty set and $\rho_{\{a\}}$ for a singleton containing an arbitrary kinematic state a. For $\alpha = \emptyset$, the expected loss $\rho_\emptyset = \mathbb{E}\left[L(\emptyset, \Phi)\right]$ is given by

$$\rho_\emptyset = (1-p)\cdot L(\emptyset, \emptyset) + p\int L(\emptyset, \{x\})s(\mathrm{d}x), \tag{22a}$$

and substituting from (7) in the above equation yields (14).

For $\alpha = \{a\}$, the expected loss is

$$\begin{aligned}\rho_{\{a\}} &= \mathbb{E}\left[L(\{a\}, \Phi)\right] && \text{(23a)}\\ &= (1-p)\cdot L(\{a\}, \emptyset) + p\int L(\{a\}, \{x\})s(\mathrm{d}x) && \text{(23b)}\\ &= (1-p)\cdot c_{01} + p\cdot\left[c_{11} + c\int \mathbb{1}_{B_a}(x)s(\mathrm{d}x)\right]. && \text{(23c)}\end{aligned}$$

The minimum of (23c) is obtained for $a = a^*_{\mathrm{MMUC}}$ given by (12). Substituting (12) into (23c) yields (13), i.e., the minimum expected loss $\rho_{\{a^*_{\mathrm{MMUC}}\}}$ for a singleton. The optimal estimate (and associated minimum expected loss) is then obtained by comparing the resulting values of expected loss as

$$\alpha^* = \underset{\alpha\in\{\emptyset, \{a^*_{\mathrm{MMUC}}\}\}}{\arg\min}\ \rho_\alpha, \tag{24}$$

which is written as a test for p in (10), where the threshold Γ in (11) is obtained by solving $\rho_\emptyset = \rho_{\{a^*_{\mathrm{MMUC}}\}}$ w.r.t. p. □

Acknowledgements

The authors would like to thank the UK Defence Science and Technology Labs (Dstl) grant no. 1000143726 (Project BLUE) for financial support.

References

[1] B. Ristic, B.-T. Vo, B.-N. Vo, and A. Farina, "A tutorial on Bernoulli filters: theory, implementation and applications," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3406–3430, 2013.

[2] D. Cormack and D. Clark, "Tracking small UAVs using a Bernoulli filter," in *2016 SSPD*, pp. 1–5, IEEE, 2016.

[3] A. Gunes and M. B. Guldogan, "Joint underwater target detection and tracking with the Bernoulli filter using an acoustic vector sensor," *Digital Signal Processing*, vol. 48, pp. 246–258, 2016.

[4] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

[5] M. Baum, P. Willett, and U. D. Hanebeck, "Polynomial-time algorithms for the exact MMOSPA estimate of a multi-object probability density represented by particles," *IEEE Transactions on Signal Processing*, vol. 63, no. 10, pp. 2476–2484, 2015.

[6] Á. F. García-Fernández and L. Svensson, "Spooky effect in optimal OSPA estimation and how GOSPA solves it," in *Fusion'19*, pp. 1–8, IEEE, 2019.

[7] T. Vu, "A complete optimal subpattern assignment (COSPA) metric," in *2020 IEEE 23rd FUSION Conference*, pp. 1–8, IEEE, 2020.

[8] M. Rezaeian and B.-N. Vo, "Error bounds for joint detection and estimation of a single object with random finite set observation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1493–1506, 2009.

[9] Á. F. García-Fernández, M. Hernandez, and S. Maskell, "An analysis on metric-driven multi-target sensor management: GOSPA versus OSPA," in *Fusion'21*, pp. 1–8, IEEE, 2021.

[10] B.-T. Vo, D. Clark, B.-N. Vo, and B. Ristic, "Bernoulli forward-backward smoothing for joint target detection and tracking," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4473–4477, 2011.

[11] B. Ristic and S. Arulampalam, "Bernoulli particle filter with observer control for bearings-only tracking in clutter," *IEEE Transactions on Aerospace and Electronic systems*, vol. 48, no. 3, pp. 2405–2415, 2012.

[12] Á. F. García-Fernández, J. L. Williams, K. Granström, and L. Svensson, "Poisson multi-Bernoulli mixture filter: direct derivation and implementation," *IEEE TAES*, vol. 54, no. 4, pp. 1883–1901, 2018.

[13] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Pt. I: Detection, Estimation, and Linear Modulation Theory*. JWS, 2004.

[14] A. H. Jazwinski, *Stochastic processes and filtering theory*. Dover, 2007.

[15] D. Middleton, *Non-Gaussian statistical communication theory*, vol. 22. JWS, 2012.

[16] G. V. Moustakides, G. H. Jajamovich, A. Tajer, and X. Wang, "Joint detection and estimation: Optimum tests and applications," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4215–4229, 2012.

[17] X. R. Li, "Optimal Bayes joint decision and estimation," in *2007 10th International Conference on Information Fusion*, pp. 1–8, IEEE, 2007.

[18] B. Vo, B. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE TSP*, vol. 62, no. 24, pp. 6554–6567, 2014.

[19] D. Musicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE TAES*, vol. 40, no. 3, pp. 1093–1099, 2004.

[20] D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic geometry and its applications*. John Wiley & Sons, 1995.

[21] I. Schlangen, E. D. Delande, J. Houssineau, and D. E. Clark, "A Second-Order PHD Filter With Mean and Variance in Target Number," *IEEE Transactions on Signal Processing*, vol. 66, pp. 48–63, Jan 2018.

[22] J. O. Berger, *Statistical decision Theory and Bayesian Analysis*. Springer Science & Business Media, 1985.

[23] H. Raiffa and R. Schlaifer, *Applied statistical decision theory*. Harvard Univ., 1961.

[24] M. H. DeGroot, *Optimal statistical decisions*, vol. 82. JWS, 2005.

[25] Y. Boers, H. Driessen, and L. Schipper, "Particle filter based sensor selection in binary sensor networks," in *2008 11th International Conference on Information Fusion*, pp. 1–7, IEEE, 2008.

[26] W. Koch, "Adaptive parameter control for phased-array tracking," in *Signal and data Processing of Small Targets 1999*, vol. 3809, pp. 444–455, International Society for Optics and Photonics, 1999.

[27] A. S. Narykov, O. A. Krasnov, and A. Yarovoy, "Effectiveness-based radar resource management for target tracking," in *2014 International Radar Conference*, pp. 1–5, IEEE, 2014.

[28] M. Guerriero, L. Svensson, D. Svensson, and P. Willett, "Shooting two birds with two bullets: How to find minimum mean OSPA estimates," in *2010 13th FUSION Conference*, pp. 1–8, IEEE, 2010.

[29] D. Salmond, N. Everett, and N. Gordon, "Target tracking and guidance using particles," in *Proceedings of the 2001 American Control Conference.*, vol. 6, pp. 4387–4392, IEEE, 2001.

[30] J. L. Williams, *Information theoretic sensor management*. PhD thesis, Massachusetts Institute of Technology, 2007.

[31] R. P. Mahler, *Statistical multisource-multitarget information fusion*. Artech House, Inc., 2007.

[32] S. Sherman, "A theorem on convex sets with applications," *The Annals of Mathematical Statistics*, pp. 763–767, 1955.

[33] D. Dionne, H. Michalska, and C. Rabbath, "Predictive guidance for pursuit-evasion engagements involving multiple decoys," *Journal of guidance, control, and dynamics*, vol. 30, no. 5, pp. 1277–1286, 2007.

[34] S. Saha, Y. Boers, H. Driessen, P. K. Mandal, and A. Bagchi, "Particle based MAP state estimation: A comparison," in *2009 12th FUSION Conference*, pp. 278–283, IEEE, 2009.

[35] D. Musicki, R. Evans, and S. Stankovic, "Integrated probabilistic data association," *IEEE TAC*, vol. 39, no. 6, pp. 1237–1241, 1994.

[36] E. Brekke, O. Hallingstad, and J. Glattetre, "The signal-to-noise ratio of human divers," in *OCEANS'10 IEEE SYDNEY*, pp. 1–10, IEEE, 2010.

[37] H. V. Poor, *An introduction to signal detection and estimation*. Springer Science & Business Media, 2013.

[38] Y. Tsypkin, *Foundations of the Theory of Learning Systems*. New York: Academic Press, 1973.

[39] A. Narykov, E. Delande, and D. E. Clark, "A formulation of the adversarial risk for multiobject filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 4, pp. 2082–2092, 2021.

[40] R. L. Winkler, *An introduction to Bayesian inference and decision*. Probabilistic Publishing, 2003.

[41] A. H. Murphy, "The value of climatological, categorical and probabilistic forecasts in the cost-loss ratio situation," *Monthly Weather Review*, vol. 105, no. 7, pp. 803–816, 1977.

[42] J. W. Tukey, "Conclusions vs decisions," *Technometrics*, vol. 2, no. 4, pp. 423–433, 1960.

Back to Contents

High Resolution DOA Estimation for Contiguous Target with Large Power Difference

1st Murtiza Ali
Department of Electrical Engineering
Indian Institute of Technology
Jammu, India
2020ree2059@iitjammu.ac.in

2nd Karan Nathwani
Department of Electrical Engineering
Indian Institute of Technology
Jammu, India
karan.nathwani@iitjammu.ac.in

***Abstract*—The Direction of arrival (DOA) for contiguous targets with a significant power difference, with fewer sensors and snapshots, is a challenging task. Our earlier work based on Modified Group delay-ℓ_1-SVD [1] can correctly resolve contiguous targets with fewer sensors. The weights in Modified Group delay ℓ_1-SVD were computed using the Hadamard product of MUSIC and Group delay (GD) weights. However, a significant power difference between the targets results in degraded performance. This is due to the incorrect MUSIC weights, which result in poor spatial resolution for contiguous targets with a significant power difference. Re-weighting the ℓ_1-SVD with Capon-MUSIC and Capon-MUSIC GD weights helps estimate DOAs with a high spatial resolution at fewer sensors. Average-Root-Mean-Square-Error (ARMSE) and Resolution Probability are used to evaluate the performance of the proposed methods.**

***Index Terms*—ℓ_1-SVD, Capon-MUSIC, high resolution, power difference**

I. Introduction

Direction-of-Arrival (DOA) estimation plays an essential role in the field of SONAR, RADAR, radio astronomy, seismology, and wireless communications [2]. DOA estimation is perceived as an important task for target localization and tracking. Some common DOA estimation techniques like delay-sum beamformer (DSB) [3], adaptive beamforming methods such as Capon beamformer [4] and subspace techniques like Multiple-Signal-Classification (MUSIC) [5] are widely used. In dynamic environments like oceans employing the passive SONAR, echos from the targets may be several dB's lower than the intentional interference or self-noise from the ship. Masking the weak signal with a strong one leads to a degraded DOA estimation [6]. The performance is degraded further when the targets with large power differences are contiguous or closely spaced [7] [8]. Further, in aerial and oceanic environments, stationarity is observed for a short time, thus limiting the number of snapshots. At low SNR regions with limited snapshots, the spatial spectrum peaks corresponding to the low power target become smaller and difficult to identify.

Estimating the DOAs of targets with a significant power difference involves estimating a weak target after mitigating the strong interference. With prior knowledge of the strong interference, the jamming method is used for DOA estimation [9]. However, the interference jamming method suffers from the reduced array aperture due to the removal of strong interference from the array manifold matrix. With the prior knowledge of the DOAs, constrained MUSIC [10] achieves accurate DOA estimation for the weak target. Additionally, using eigenvectors projection for modified spectrum has proved to be efficient for estimating DOAs provided the targets are known [11]. An estimation technique based on the Capon-MUSIC algorithm was proposed in [7] for closely spaced targets but suffers from reduced peak levels for both strong and weak targets. Using "Eigenbeam *m*Capon" [12], the number of targets and DOA estimation for weak targets can be done at the cost of repeated iterations.

Some other studies use compressive sensing (CS) without prior knowledge to estimate the DOA of strong targets, then remove the strong source effect via orthogonal complements [13]. Following the convex optimization framework, Sparse Recovery algorithms like ℓ_1-SVD offer theoretical guarantees to perform well in noise [14] [15]. A robust sparse asymptotic minimum variance (RSAMV) algorithm was proposed in [16] after analyzing the loss of weak targets in the asymptotic minimum variance (SAMV) algorithm. Also, a computationally expensive method of robust orthogonal projection [17] estimates the target DOAs without prior knowledge. Certain practical constraints are encountered in real-time scenarios, like the number of snapshots, sensors, and region of SNR [18].

The overall performance of the aforementioned algorithms degrades profoundly by reducing the number of snapshots and the number of sensors. In this paper, we have proposed MUSIC Capon-GD and ℓ_1-SVD in two variants for estimating the DOAs for contiguous targets with significant power differences and fewer snapshots and sensors. The proposed method can correctly resolve contiguous targets with a significant power difference. The rest of the paper is organised as follows: In Section II we discuss signal model followed by related works in Section III. Section IV elaborates the two proposed methods in detail. Performance evaluation in Section V and conclusion in Section VI.

II. Signal Model

Let us consider a Uniform Linear Array (ULA) of N sensors on which J narrowband signals with centre frequency f are arriving from directions $\theta_1 \ldots, \theta_J$. The data $\mathbf{y}(t)$ received at the array during time t is given by

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

$$\mathbf{y}(t) \quad = \quad \sum_{j=1}^{J}\mathbf{a}(\theta_j)s_j(t) \; + \; \mathbf{n}(t);\, t \quad = \quad 1,\dots,L \quad (1)$$

Here $\mathbf{a}(\theta_j) = [1,\dots,e^{-j2\pi(d/\lambda)\sin(\theta_j)(N-1)}]^T$ is array steering vector for θ_j direction, λ is the incoming signal wavelength, d is the inter-sensor spacing conserved at $\lambda/2$. Also, $s_j(t)$ is the amplitude of the j^{th} incoming target signal at t^{th} snapshot. $\mathbf{n}(t)$ is the independent and identical (i.i.d) circularly symmetric complex Gaussian noise, with zero mean and diagonal covariance matrix, i.e, $\mathbf{n}(t) \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2\mathbf{I})$ [19]. In matrix form the Eq.1 can be written as:

$$\mathbf{Y} = \mathbf{AS} + \mathbf{N} \quad \in \mathbb{C}^{N\times L} \quad (2)$$

where $\mathbf{S} \in \mathbb{C}^{J\times L}$ is a matrix of amplitudes for incoming signals for L snapshots, $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_J)] \in \mathbb{C}^{N\times J}$ being the array manifold matrix for J narrowband targets. $\mathbf{N} \in \mathbb{C}^{N\times L}$ the noise model matrix [19]. The estimated covariance matrix of $\mathbf{y}(t)$ is given as:

$$\hat{\mathbf{R}} = \frac{1}{L}\sum_{t=1}^{L}\mathbf{y}(t)\mathbf{y}(t)^H = \frac{1}{L}\mathbf{YY}^H \quad (3)$$

III. Some Earlier Related Works

There are a number of DOA estimation techniques pertaining to targets with large power differences. Some of the basic techniques used are mentioned herein:

A. MUSIC and Capon-MUSIC Algorithm

The spatial spectrum for MUSIC is derived from the noise subspace of the sample covariance matrix [5] $\hat{\mathbf{R}}$ as:

$$\mathbf{P_{music}}(\Theta) = \frac{1}{\mathbf{a_\theta}^H\mathbf{U_n}\mathbf{U_n}^H\mathbf{a_\theta}} \quad (4)$$

Thus Capon-Music is defined in [7] as:

$$\mathbf{P_{C-music}}(\Theta) = \frac{\mathbf{a_\theta}^H\hat{\mathbf{R}}^{\mathbf{-1}}\mathbf{a_\theta}}{\mathbf{a_\theta}^H\mathbf{U_n}\mathbf{U_n}^H\mathbf{a_\theta}} \quad (5)$$

The DOAs are found by searching for the peaks in the spatial spectrum obtained in Eq.(4) and Eq.(5).

B. ℓ_1-SVD

Following the convex optimization, the SR algorithms are efficient in terms of snapshots [15]. They exhibit high resolution and are effective at low SNRs. With the knowledge of the over-complete basis $\mathbf{A}_\Omega$ which comprises of the over-complete set of basis vectors $\mathbf{a}_{\theta_i}, ..., \mathbf{a}_{\theta_K}$,where K being the discrete set of spatial search points $\theta \in [-90, 90)$.

Consider the SVD of the data matrix $\mathbf{Y}$ in Eq.2 as:

$$\mathbf{Y} = \mathbf{U}\Lambda\mathbf{V}^H \quad (6)$$

$\mathbf{U}$ and $\mathbf{V}$ being singular vectors. We can define

$$\tilde{\mathbf{Y}} = \mathbf{U}\Lambda\mathbf{E}_J = \mathbf{YVE}_J \in \mathbb{C}^{(N\times J)} \quad (7)$$

where $\mathbf{E}_J = [\mathbf{I}_J \;\; \mathbf{0}]^T \in \mathbb{C}^{(L\times J)}$, $\mathbf{I}_J$ is the $(J\times J)$ identity matrix, and $\mathbf{0}$ is the $J\times(L-J)$ zero-matrix. Rewriting the Eq. (7) as:

$$\tilde{\mathbf{Y}} = \mathbf{A}_\Omega\tilde{\mathbf{S}} + \tilde{\mathbf{N}} \quad (8)$$

where $\tilde{\mathbf{S}} = \hat{\mathbf{S}}\mathbf{VE}_J$ and $\tilde{\mathbf{N}} = \mathbf{NVE}_J$. The reduced data matrix $\tilde{\mathbf{Y}}$ retains most of the signal power. The row-support of $\tilde{\mathbf{S}}$ being identical to $\hat{\mathbf{S}}$. Expressing the columns of $\tilde{\mathbf{Y}}$ as:

$$\tilde{\mathbf{y}}(j) = \mathbf{A}_\Omega\tilde{\mathbf{s}}(j) + \tilde{\mathbf{n}}(j), j = 1,\dots,J \quad (9)$$

Defining $s_k^{(\ell_2)} = \sqrt{\sum_{j=1}^{J}|\tilde{s}_k(j)|^2}, k = 1,\dots,K$, where $\tilde{s}_k(j)$ represents the k^{th} element of $\tilde{\mathbf{s}}(j)$. Collectively, a row vector is created by $s_k^{(\ell_2)}$ as:

$$\bar{\mathbf{s}}^{(\ell_2)} = [s_1^{(\ell_2)}, \dots, s_K^{(\ell_2)}]^T \in \mathbb{R}^K \quad (10)$$

It is observed that the support of $\bar{\mathbf{s}}^{(\ell_2)}$ is identical to row-support of $\tilde{\mathbf{S}}$ [20] hence, $\bar{\mathbf{s}}$ is considerd as a good approximation to the spatial magnitude spectrum $\hat{\mathbf{s}}(t)$ in Eq. (1).

Thus, we can estimate $\bar{\mathbf{s}}^{(\ell_2)}$ can by solving the optimization problem [15],

$$\min ||\bar{\mathbf{s}}^{(\ell_2)}||_1 \quad \text{subject to } ||\tilde{\mathbf{Y}} - \mathbf{A}_\Omega\tilde{\mathbf{S}}||_F^2 \leq \eta^2 \quad (11)$$

where $||\cdot||_F$ is Frobenius norm and η being the regularization parameter, specifing the amount of noise to be allowed.

IV. Proposed Method for DOA Estimation

The algorithms mentioned in Section III produce inaccurate results for targets with a large power difference. The height of the spectrum peak corresponding to the lower power target is reduced compared to the strong target. Thus spatial spectrum peak corresponding to a low power target is hard to pick or observe. The performance deteriorates further if snapshots are limited and sensors are less for contiguous targets. Algorithm in [1] can be modified by re-weighting the ℓ_1-SVD by Capon-MUSIC GD (CMGD-ℓ_1-SVD) and Capon-MUSIC ℓ_1-SVD (CM-ℓ_1-SVD).

A. Capon-MUSIC Group delay ℓ_1-SVD

With fewer sensors and snapshots, MUSIC cannot distinguish the contiguous targets, as evident from MUSIC-weights in Fig.1. While Capon-MUSIC [7] can recognize the contiguous targets with power difference, the height of the spectral peak corresponding to the weak target is imperceptible. Using the high-resolution GD [21] with the ability of Capon-MUSIC to locate DOAs of targets with large power differences, CMGD-ℓ_1-SVD inherits the proprieties of the methods mentioned above for accurate DOA estimation. Using the phase information in the GD function for DOA estimation of contiguous targets with fewer sensors has been shown in [22]. The overall GD function $\tau(\theta)$ is defined as :

$$\tau(\theta) = \sum_{i=1}^{N-J}|-\frac{d\Phi_i(\theta)}{d\theta}|^2 \quad (12)$$

here phase spectrum is evaluated as inner product of the steering and noise singular vectors, i.e, $\Phi_i(\theta) = arg(\mathbf{a}_\Omega(\theta)^H\mathbf{u}_i)$, $i = 1,\dots,N-J$. Where $\mathbf{u}_i$ is the $(J+i)^{th}$ singular vector of $\mathbf{U}$.

GD function has peak preserving ability due to additive nature of phase spectrum [23] [24]. Decomposing over-complete basis as $\mathbf{A}_\Omega = [\mathbf{A} \quad \mathbf{B}]$ where $\mathbf{A} \;\in \mathbb{C}^{N\times J}$ corresponds to

Back to Contents

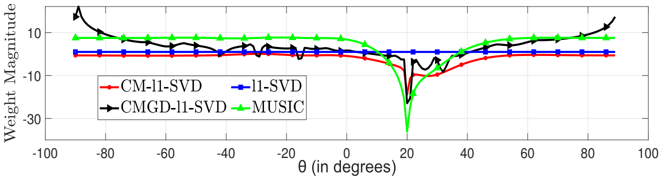


Fig. 1. Angular spectrum for different weights at N=6, SINR=20 dB and targets at 20°& 26°

the true DOAs and $\mathbf{B} \in \mathbb{C}^{N\times(K-J)}$ to remaining directions. Evaluating the phase response of the matrix we get:

$$\angle\{\mathbf{A}_{\Omega}^{H}\mathbf{U}_N\} = [\angle\{\mathbf{A}^H\mathbf{U}_N\}\ \angle\{\mathbf{B}^H\mathbf{U}_N\}] = [\Phi_A\ \Phi_B] \quad (13)$$

Complex values of $\mathbf{A}^H\mathbf{U}_N$ being small due to orthogonality of signal and noise subspace. Thus the phase response of $\Phi_A \in \mathbb{C}^{J\times(N-J)}$ will result in sharp phase shifts. Taking the gradient of unwrapped spectrum of columns as $\nabla[\Phi_A\ \Phi_B] = [\tau_A\ \tau_B]$. Thus we obtain the GD weights as $(\mathbf{w}_{Group-delay})^{-1} = [[\tau_A^{(\ell_2)}]^T\ [\tau_B^{(\ell_2)}]^T]^T$. Using Capon-MUSIC spectrum to compensate for the instability in the GD spectrum, we take the Hadamard product of the Capon-MUSIC spectrum and GD weights as:

$$\mathbf{W}_{CMGD} = \frac{1}{\mathbf{w}_{Group-delay} \odot \mathbf{P}_{C-music}} \quad (14)$$

The use of $\mathbf{W}_{CMGD}$ for re-weight the ℓ_1-SVD for robust performance at large power differences for contiguous targets, compensates SR algorithms inaccurate estimate with less sensors and also GD's noise sensitivity. After diagonalizing the weights as $\mathbf{W} = \text{diag}\,(\mathbf{W}_{CMGD})$ the minimization equation in [15] is rewritten as given in [1] as:

$$\min||\mathbf{W}\bar{\mathbf{s}}^{(\ell_2)}||_1 \text{subject to} ||\tilde{\mathbf{Y}} - \mathbf{A}\tilde{\mathbf{S}}||_F^2 \leq \eta^2, \quad (15)$$

where in the position of the largest peak of $\mathbf{W}\bar{\mathbf{s}}^{(\ell_2)}$ gives the estimate of the target DOA. The value of η should be large enough such that the probability of $||\mathbf{N}||_F^2 \geq \eta^2$ is small. With known distribution of $\mathbf{N}$, we can find the mean of $||\mathbf{N}||_F^2$ and the value as a choice for η^2 [15].

B. Capon-MUSIC ℓ_1-SVD

From Fig.1 we can observe the MUSIC weights are not able to recognise the target at low power (at 26°). While CMGD-ℓ_1-SVD weights result in spurious peaks around the true DOAs. This being due to GD's sensitivity to noise, which leads to degraded performance at significant power differences thus resulting in incorrect weighting. In another attempt to emphasize the importance of re-weighting, the re-weighting for ℓ_1-SVD was performed only using Capon-MUSIC weights as:

$$\mathbf{W}_{\mathbf{C-music}} = \frac{1}{\mathbf{P}_{C-music}} \quad (16)$$

Re-weighting the ℓ_1-SVD with the $\mathbf{W}_{C-music}$ results in far better performance in terms of DOA estimation than the Capon MUSIC GD ℓ_1-SVD (CMGD-ℓ_1-SVD). Although exempting the GD from the estimation of DOA results in the loss of high resolution, apparent peaks are still observed at true DOA. Further, the weights around the true DOAs are relatively smooth with no spurious peaks as compared to CMGD-ℓ_1-SVD, as seen from Fig.1. By re-weighting with $\mathbf{W}_{C-music}$, we can resolve contiguous targets with significant power difference with fewer sensors, as Capon-MUSIC performs well than traditional MUSIC (Fig.2 (a)). After obtaining the diagonalized weight as $\mathbf{W}_{C-music}$, the largest peak of $\mathbf{W}\bar{\mathbf{s}}^{(\ell_2)}$ in Eq.15 provides the estimate of the target DOAs as shown in Algorithm 1.

Algorithm 1 Pseudo Algorithms for DOA estimation using CMGD-ℓ_1-SVD and CM-ℓ_1-SVD

```
Input: Array Data matrix: $\mathbf{Y} \in \mathbb{C}^{N\times L}$ and $J$ targets.
Output: Estimated spatial spectrum $\mathbf{W}\bar{\mathbf{s}}^{(\ell_2)}$
1: Compute the covariance matrix $\hat{\mathbf{R}}$ from Eq. 3 and inverse as $\hat{\mathbf{R}}^{-1}$
if CMGD-$\ell_1$-SVD
    $a$: Compute the $\mathbf{P}_{\mathbf{C-music}}(\Theta)$ from Eq. 5 and $(\mathbf{w}_{Group-delay})^{-1}$
    $b$: Obtain $\mathbf{W}_{CMGD}$ from Eq.16 and Re-weight $\ell_1$-SVD.
else CM-$\ell_1$-SVD
    $a$: Compute the Capon MUSIC weights as shown in Eq.14
    $b$: Re-weight $\ell_1$-SVD using $\mathbf{W}_{\mathbf{C-music}}$
2: Estimate the spatial spectrum $\mathbf{W}\bar{\mathbf{s}}^{(\ell_2)}$ from Eq. 15.
```

V. Performance Evaluation

The performance of proposed methods is evaluated through simulations and compared with some state-of-the-art algorithms. With a ULA of half-wavelength inter sensor spacing, multi-snapshot processing is performed on $L = 200$ for contiguous targets at 20° and 26° unless stated otherwise. The efficiency of the proposed methods is evaluated with a step size of 1°. The number of sensors used for observations is $N = 6$, and all the results are evaluated over S= 10^3 Monte-Carlo simulations. By reducing the number of sensors, we have reduced the computational complexity of the complex SR algorithms, thus resulting in a better trade-off between performance and computational time. While evaluating the performance of the proposed methods, the Signal to Noise Ratio (SNR) of the strong target is fixed at 20 dB, while the SNR of the second target is varied over the range of [20 to -5] dB. The $\sigma_n^2 = 1$ or fixed, which is generally observed in real-world scenarios. We have evaluated the proposed methods in terms of ARMSE [20], and Resolution Probability [7]. Wherein ARMSE is defined as:

$$ARMSE = \frac{1}{J}\left\{\sum_{j=1}^{J}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(\hat{\theta}_j^s - \theta_j\right)^2}\right\} \quad (17)$$

and for Resolution Probability, two targets are considered well separated if the absolute difference between true and estimated target DOA is less than the difference between true target locations i.e, $(|\hat{\theta}_1 - \theta_1|\ \&\ |\hat{\theta}_2 - \theta_2|) < |\theta_1 - \theta_2|$.

A. ARMSE Vs Sensors

The number of sensors plays a crucial role in the accuracy of a DOA estimator. As can be observed from Fig. 2 (a), when

Back to Contents

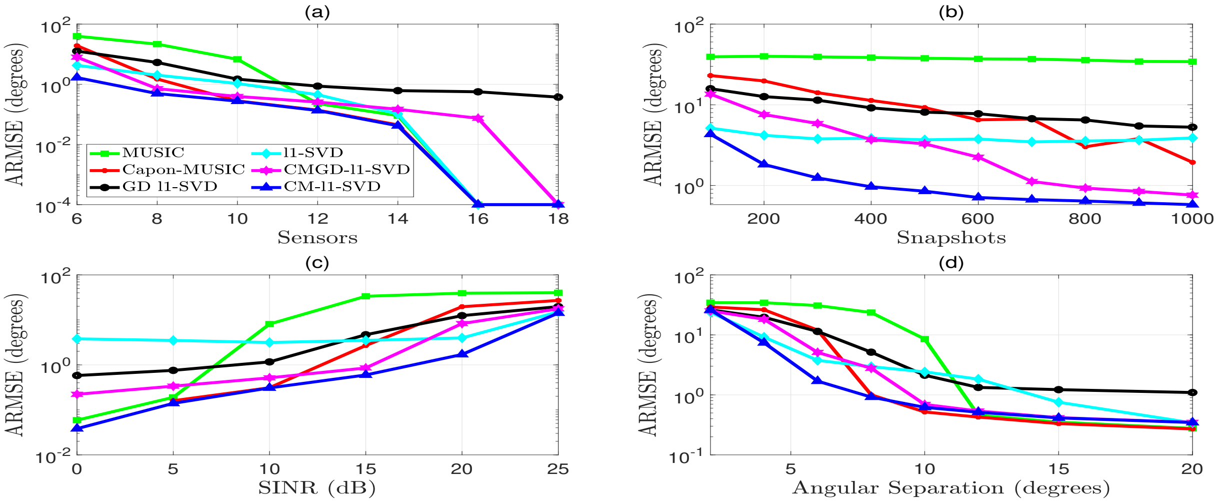


Fig. 2. Comparison of proposed methods with state-of-the-art methods using (a) ARMSE Vs. No. of Sensors, (b) ARMSE Vs. Snapshots, (c) ARMSE Vs SINR and, (d) ARMSE Vs Angular Seperation

targets are close (6° of angular separation), and at a power difference of 20 dB, the proposed algorithms, i.e., CMGD-ℓ_1-SVD and CM-ℓ_1-SVD achieve the low ARMSE score than the state-of-the-art methods with less number of sensors. However in CMGD-ℓ_1-SVD due to the use of GD, spurious peaks occur, resulting in degraded performance [20] [21] than CM-ℓ_1-SVD. One can also observe that with an adequate number of sensors (N > 14), all the mentioned algorithms can resolve and achieve a minimum ARMSE score. One also has to satisfy the Restricted Isometry Property constraint (i.e., $N > 2J$) [25] thus, the number of sensors is restricted to $N = 6$.

B. ARMSE Vs Snapshots

In dynamic environments like Oceans, stationarity is observed for a brief period, thus restricting a large number of snapshots. The performance of the algorithms degrades upon reducing the snapshots. The results evaluated in Fig.2 (b) for two contiguous targets at 20° and 26° with a power difference of 20 dB illustrate that proposed methods can achieve the lowest ARMSE score for less number of snapshots as compared to other algorithms. Here again CM-ℓ_1-SVD outperforms CMGD-ℓ_1-SVD and traditional ℓ_1-SVD at less snapshots. However ℓ_1-SVD has a better performance than CMGD-ℓ_1SVD till L = 400 snapshots. One of the plausible arguments for this anomalous behavior in CMGD-ℓ_1SVD might be due to zero's close to unit circle at low snapshots, thus causing spurious peaks to occur in the spatial spectrum.

C. ARMSE Vs SINR

In this section, we have evaluated the performance of the proposed methods for SINR (Signal-to-Interference-Noise-Ratio). The input SNR is defined as SNR = $10log_{10}\ \sigma_s^2/\sigma_n^2$ where σ_s^2 is the power of target. Here we have set a strong target T_1 with SNR_1 and other weak target T_2 with SNR_2. We also define SINR as $SINR = SNR_1 - SNR_2$. The variation of $SINR$ is shown in Fig.2 (c). Wherein SNR_1 is fixed at 20 dB and SNR_2 is reduced from 20 dB to -5 dB. The results are evaluated at $L = 200$ and $N = 6$. We can observe that the proposed method CM-ℓ_1-SVD achieves the minimum ARMSE at all SINRs. Further, it is important to note that the performance of CMGD-ℓ_1-SVD starts to degrade when the two targets have a power difference of 15 dB (i.e., SINR =15 dB). This can be attributed to the inability of GD to recognize the true DOAs at higher values of SINR. Also, the performance of the ℓ_1-SVD is somewhat constant across all SINRs. The plausible argument could be the high resolution (needle-like) peaks obtained in the spatial spectrum of ℓ_1-SVD despite the power difference present in targets.

D. ARMSE Vs Angular Separation

In this section, we evaluate the ability of the proposed methods to resolve the targets with a significant power difference at different angular separations. This is done by fixing a target T_1 at 20° and varying the angular position of the T_2 from 22° to 40°. Again these results are evaluated with $N = 6$ and $L = 200$. We can observe from Fig.2 (d) that the proposed method CM-ℓ_1-SVD can achieve the lowest ARMSE score in the current setup. Beyond angular separation of 8°, CMGD-ℓ_1-SVD outperforms traditional ℓ_1-SVD and matches the performance of its counterpart CM-ℓ_1-SVD. Upon increasing the separation, almost all algorithms perform well. The other proposed method can perform better than traditional algorithms like MUSIC and GD-ℓ_1-SVD and even Capon-MUSIC under the same setup.

E. Resolution Probability Vs. Angular Separation and SINR

During this investigation stage, we examine the Resolution Probability, i.e., the resolving capability of different algorithms in terms of probability of separation. This is done with respect to angular separation and SINR. These results are evaluated

Back to Contents

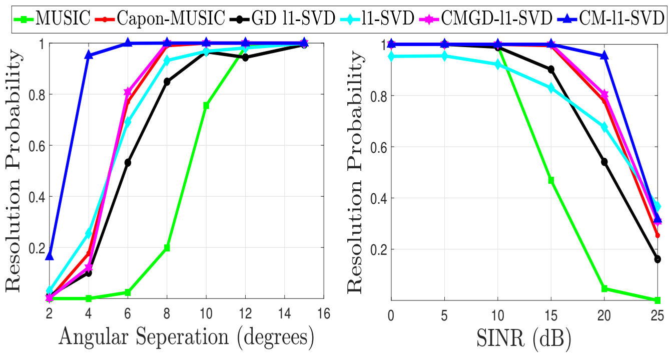


Fig. 3. Plots for Resolution Probability evaluated at $N = 6$, $L = 200$.

at $N = 6$ and $L = 200$. In the Angular separation case, it can be observed that the proposed methods can achieve good resolution probability across the angular separation axis, with CM-ℓ_1-SVD outperforming all other algorithms in comparison. Even at an angular separation of 2°, it has a resolution probability of around 0.2. Under the same setup, we can observe that both proposed methods can achieve good resolution probability in SINR. Both these methods perform well across all the SINR ranges.

VI. Conclusion

This paper presents two techniques, CMGD-ℓ_1-SVD and CM-ℓ_1-SVD, to provide accurate DOA estimation for contiguous targets with significant power differences ($SINR = 25dB$). Although uniform (constant) weighting and high-resolution property of ℓ_1-SVD provide a reasonable DOA accuracy, it starts to degrade when the power difference between targets increases at fewer sensors and snapshots. With fewer sensors (N=6) and fewer snapshots (L=100), we can achieve the lowest ARMSE scores and highest Resolution Probability for CM-ℓ_1-SVD in comparison to the state-of-the-art algorithms. This is attributed to the appropriate re-weighting achieved by the CM-ℓ_1-SVD, when the power difference between targets increases significantly, resulting in higher DOA accuracy than the traditional ℓ_1-SVD. Due to the inability of GD to perceive low power targets (missing peaks in the spectrum), the CMGD-ℓ_1-SVD performance is affected beyond a specific value of SINR. Also, it is observed that by increasing the step size of the grid, the spatial resolution is likely to improve.

VII. Acknowledgement

This work is funded by SERB, Govt. of India, via Project No. CRG/ 2018/003920 under CRG-SERB Scheme.

References

[1] M. Ali, A. Koul, and K. Nathwani, “Group Delay Based Re-Weighted Sparse Recovery Algorithms for Robust and High-Resolution Source Separation in DOA Framework,” in *Proc. Interspeech*, 2021, pp. 3031–3035.

[2] H. L. V. Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Melbourne, FL, USA: John Wiley & Sons, Inc., 2002.

[3] S. N. Bhuiya, F. Islam, and M. Matin, “Analysis of direction of arrival techniques using uniform linear array,” *International Journal of Computer Theory and Engineering*, vol. 4, pp. 931–934, 01 2012.

[4] J. Capon, “High-resolution frequency-wavenumber spectrum analysis,” *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[5] R. Schmidt, “Multiple emitter location and signal parameter estimation,” *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[6] L. Wan, X. Kong, and F. Xia, “Joint range-doppler-angle estimation for intelligent tracking of moving aerial targets,” *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1625–1636, 2018.

[7] Y. Gao, X. Jia, J. Xu, T. Long, and X.-g. Xia, “A novel doa estimation method for closely spaced multiple sources with large power differences,” in *2015 IEEE Radar Conference (RadarCon)*, 2015, pp. 1276–1279.

[8] W. L. Qingyuan Fang, Mengzhe Jin and Y. Han, “Doa estimation for sources with large power differences,” *International Journal of Antennas and Propagation*, vol. 2021, p. 103388, 2021.

[9] C. Hui and W. Yongliang, “Interference jamming doa estimation algorithm,” in *2005 IEEE Antennas and Propagation Society International Symposium*, vol. 2B, 2005, pp. 358–361 vol. 2B.

[10] D. Linebarger, R. DeGroat, E. Dowling, and P. Stoica, “Constrained beamspace music,” in *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, 1993, pp. 548–551 vol.4.

[11] Z. Z. F. Huang and Q. Zheng, “Doa estimation based on a modified spatial spectrum,” in *Proceedings of the IET International Radar Conference 2015*, vol. 4, 2015, pp. 1–4.

[12] J. X. Y. Gao and X. Jia, “Joint number and doa estimation via the eigen-beam mcapon method for closely spaced sources,” *Science China Information Sciences*, vol. 58, pp. 1–3, 2015.

[13] Y. Yang and X. Mao, “Hybrid method of doa estimation using nested array for unequal power sources,” in *2018 International Conference on Radar (RADAR)*, 2018, pp. 1–5.

[14] N. Hu, Z. Ye, D. Xu, and S. Cao, “A sparse recovery algorithm for DOA estimation using weighted subspace fitting,” *Signal processing*, vol. 92, no. 10, pp. 2566–2570, 2012.

[15] D. Malioutov, M. Cetin, and A. S. Willsky, “A sparse signal reconstruction perspective for source localization with sensor arrays,” *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, 2005.

[16] X. Zhang, J. Sun, and X. Cao, “Robust direction-of-arrival estimation based on sparse asymptotic minimum variance,” *The Journal of Engineering*, vol. 2019, no. 21, pp. 7815–7821, 2019.

[17] Q. Liu, C. Zeng, S. Li, Z. Yang, and G. Liao, “Robust estimations of doa and source number with strong and weak signals coexisting simultaneously based on a sparse uniform array,” *The Journal of Engineering*, vol. 2019, no. 20, pp. 6387–6389, 2019.

[18] F. B. Jensen, W. A. Kuperman, M. B. Porter, and H. Schmidt, *Computational Ocean Acoustics (Modern Acoustics and Signal Processing)*. Springer, 2011.

[19] C. F. Mecklenbräuker, P. Gerstoft, E. Zöchmann, and H. Groll, “Robust estimation of doa from array data at low snr,” *Signal Processing*, vol. 166, p. 107262, 2020.

[20] M. Ali, A. Koul, and K. Nathwani, “Significance of group delay spectrum in re-weighted sparse recovery algorithms for doa estimation,” *Digital Signal Processing*, vol. 122, p. 103388, 2022.

[21] L. Kumar, R. Mandala, and R. M. Hegde, “Music-group delay based methods for robust DOA estimation using shrinkage estimators,” in *7th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. IEEE, 2012, pp. 281–284.

[22] M. Shukla and R. M. Hegde, “Significance of the music-group delay spectrum in speech acquisition from distant microphones,” in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2010, pp. 2738–2741.

[23] B. Yegnanarayana, D. Saikia, and T. Krishnan, “Significance of group delay functions in signal reconstruction from spectral magnitude or phase,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 3, pp. 610–623, 1984.

[24] R. M. Hegde, H. A. Murthy, and V. R. R. Gadde, “Significance of the modified group delay feature in speech recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 190–202, 2007.

[25] R. G. Baraniuk, “Compressive sensing,” *IEEE Signal Processing Magazine*, vol. 118, Jul. 2007.

Compressive Self-Noise Cancellation in Underwater Acoustics

1st Pawan Kumar
Dept. of Electrical Engineering
IIT Jammu, India
pawanecom@gmail.com

2nd Karan Nathwani
Dept. of Electrical Engineering
IIT Jammu, India
karan.nathwani@iitjammu.ac.in

3rd Vinayak Abrol
Infosys Centre for AI
IIIT Delhi, India
abrol@iiitd.ac.in

4th Suresh Kumar
Naval Research Board
NPOL Kochi, India
sureshkumarnpol@yahoo.co.in

***Abstract*—The purpose of sonar is to detect the stealthy target in shallow water. The main barrier to locating the target is sonar's self-noise. Existing subspace-based noise suppression methods typically employ eigenanalysis-based methods involving high computational complexity. Recent approaches based on compressed sensing (CS) or sparse representations (SR) are computationally efficient. It is not straightforward to extend existing CS/SR-based methods for self-noise cancellation as, first, the energy of interference is much higher than the target, and second, it also exhibits similar sparsity properties. This work presents a novel method to combine the advantages of a subspace-based noise cancellation approach with low complexity of working with fewer CS measurements. Both target recovery and self-noise cancellation are done in the compressive domain only. Experimental results demonstrate the robustness of the proposed approach for both narrowband and broadband targets at very low signal-to-interference-noise (SINR).**

***Index Terms*—Self-noise cancellation, compressed sensing, underwater acoustics, sensor array**

I. Introduction

The problem of detecting an underwater target in the presence of background noise and estimating parameters such as range, depth, and bearing have been a point of research in the last few decades [1]–[4]. One of the major noise sources is the self-noise (interference) generated from the ship itself, which makes it challenging to perform passive signal processing onboard a moving ship to detect or locate a source. The standard approach to mitigate the effects of any interfering signal is to project the observed signal onto the subspace orthogonal to that of the interfering signal.

Existing methods proposed in various studies mainly differ in the computation of the noise subspace, which is estimated from eigenvectors of the correlation matrix of either the observed or interference signal [5]–[7]. If the correlation matrix is computed from observations, the number of sampled eigen directions corresponding to noisy subspace is often done empirically. To address this issue, work in [4] proposed a method based on eigenanalysis of cross-spectral density matrix (CSDM) of the data followed by beamforming each of the components. This helps identify the components with low target-to-interference power for robust detection of the target signal. When the energy of the interference signal is powerful compared to the target, the only reliable way of detection is to compute the noisy subspace from an estimate of the interference signal itself [7]. Nevertheless, any eigenanalysis-based approach suffers from high computational complexity, especially for high dimensional data from multiple sensors [8].

Further, in recent years, the sparse representation (SR) based methods have proved to be successful in a variety of underwater acoustic tasks such as direction-of-arrival (DOA) estimation and source localization [9]–[11]. These methods are based on the fact that it is relatively easy to find a sparse representation for target data given a suitable overcomplete basis (e.g., Fourier, wavelet) instead of noise (assumed to be additive). Other works further exploit the sparsity of signals to perform such tasks using very few random projections based on the principles of compressed sensing (CS) [12], [13]. For instance, work in [14], [15] proposed a compressive beamformer for DOA estimation. However, the CS/SR-based methods cannot simply be extended for self-noise cancellation (SNC). First, the interference energy is much higher than the target, and second, it exhibits similar sparsity properties. This work proposes a novel method to combine the advantages of a subspace-based noise cancellation approach with few CS measurements.The advantage of the proposed method is that it has reduced time and memory complexity compared to the high-dimensional subspace-based methods.

The rest of the paper is organized as follows: Section II describes array data model, The proposed compressive SNC framework follows this in Section III. Finally, the experimental results are detailed in Section IV, with a brief conclusion in Section V.

II. Array Data Model

Consider N sensor elements are arranged on a Uniform Linear Array (ULA) [1]. The received array signal $\mathbf{y}[n] \in \mathbb{R}^{T\times 1}$ at n^{th} sensor is a combination of the target signal, self-noise, and ambient noise as shown below :

$$\mathbf{y}[n] = \mathbf{A}(\theta)\mathbf{s}[n] + \mathbf{a}(\theta_0)s_o[n] + \mathbf{v}[n] \tag{1}$$

$\mathbf{A}(\theta) \in \mathbb{R}^{T\times J}$ is the steering matrix for signal vector $\mathbf{s}[n] \in \mathbb{R}^{J\times 1}$ at angle of incident θ of signal vector on array, s_0 is the self-noise (generated by the mother ship) associated with the steering vector $\mathbf{a}(\theta_0) \in \mathbb{R}^{T\times 1}$, here θ_0 is the self-noise bearing

This work is supported through Project No. NRB/4003/SSB/PG466/20-21.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

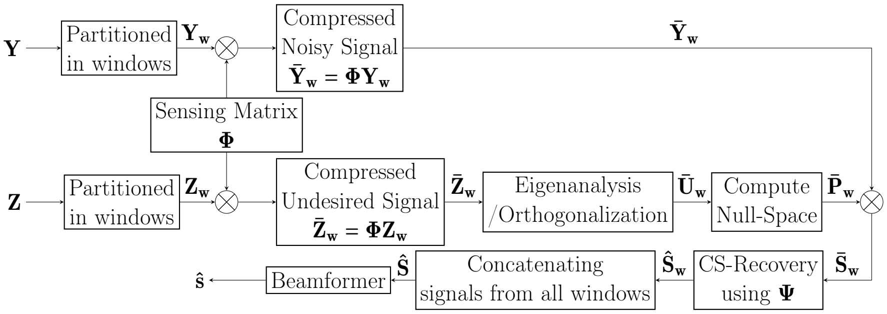

Y
Partitioned in windows
$\mathbf{Y_w}$
Compressed Noisy Signal
$\bar{\mathbf{Y}}_{\mathbf{w}} = \mathbf{\Phi Y_w}$
$\bar{\mathbf{Y}}_{\mathbf{w}}$
Sensing Matrix
$\mathbf{\Phi}$
Z
Partitioned in windows
$\mathbf{Z_w}$
Compressed Undesired Signal
$\bar{\mathbf{Z}}_{\mathbf{w}} = \mathbf{\Phi Z_w}$
$\bar{\mathbf{Z}}_{\mathbf{w}}$
Eigenanalysis /Orthogonalization
$\bar{\mathbf{U}}_{\mathbf{w}}$
Compute Null-Space
$\bar{\mathbf{P}}_{\mathbf{w}}$
$\hat{\mathbf{s}}$
Beamformer
$\hat{\mathbf{S}}$
Concatenating signals from all windows
$\hat{\mathbf{S}}_{\mathbf{w}}$
CS-Recovery using $\mathbf{\Psi}$
$\bar{\mathbf{S}}_{\mathbf{w}}$

Fig. 1. Compressed self-noise cancellation via null-space projection

angle and $\mathbf{v}[n] \in \mathbb{R}^{T\times 1}$ represents the additive Gaussian noise. Received noisy signal at ULA is

$$\mathbf{Y} = [[\mathbf{y}(1)]\,,\,[\mathbf{y}(2)]\,,\,[\mathbf{y}(3)],\cdots[\mathbf{y}(N)]] \in [T \times N] \qquad (2)$$

For simplicity, we denote the signal model in matrix form as:

$$\mathbf{Y} = \mathbf{S} + \mathbf{Z} \qquad (3)$$

where the goal is to recover/detect the signal component $\mathbf{S}$ by removing undesired component $\mathbf{Z}$ due to the ambient and self-noise.

III. Proposed Method: Compressive SNC

The conventional approach for self-noise cancellation maximizes signal-to-interference noise ratio (SINR) by utilizing the null space projection techniques. The optimal solution for detecting the desired signal signature by eliminating interference due to undesired signatures plus the noise is given as [6], [16]:

$$\hat{\mathbf{S}} = \mathbf{PY};\ \mathbf{P} = (\mathbf{I} - \mathbf{UU}^{\dagger}) \qquad (4)$$

where $\mathbf{P}$ is the projection matrix, and † denotes the pseudo-inverse. $\mathbf{U}$ are selected sampled orthogonal columns of $\mathbf{Z}$. The crucial difference between existing approaches lies in the computation of basis $\mathbf{U}$ which is estimated either from the correlation matrix of the interference or the observation matrix. In this work, we consider the former case (which is optimal here) as the energy of the interference/undesired signatures is powerful compared to the desired signature. From a numerical point of view, $\mathbf{U}$ is mostly estimated by applying orthogonal decomposition such as singular value decomposition (SVD) or rank-revealing QR decomposition and is chosen to be undercomplete as only the first few dominant directions (e.g., singular vectors) suffice to characterize the self-noise.

The inherently data-dependent nature of SVD/QR estimation involving expensive eigendecomposition [6] often hinders its use in severely resource-constrained settings such as underwater acoustics [17]. To address this issue, we propose to perform the null-space projection-based SNC in a compressed domain by projecting observed sensor data onto a random lower-dimensional subspace as highlighted in Fig.1. Here, we not only use compressed measurements to recover/detect the target signal but also perform the estimation of basis $\mathbf{U}$ for noise cancellation. To this aim, we reexpress matrices $\mathbf{Y}$ and $\mathbf{Z}$ as:

$$\begin{aligned}\mathbf{Y} &= [[\mathbf{y}(1)]\,,\,[\mathbf{y}(2)],\ldots[\mathbf{y}(T)]]^{\mathcal{T}},\\ \mathbf{Z} &= [[\mathbf{z}(1)]\,,\,[\mathbf{z}(2)],\ldots[\mathbf{z}(T)]]^{\mathcal{T}}\end{aligned} \qquad (5)$$

where $\mathbf{y}(m)$ and $\mathbf{z}(m) \in \mathbb{R}^N$ are m^{th} row of $\mathbf{Y}$ and $\mathbf{Z}$ respectively. Here $[\ .\ ]^{\mathcal{T}}$ represents transpose of a matrix. Initially, We have partitioned $\mathbf{Y}$ and $\mathbf{Z}$ in 'B' number of windows.

$$\begin{aligned}\mathbf{Y_w}(m) &= [[\mathbf{y}((m-1)L+1)],\cdots,[\mathbf{y}((m-1)L+L)]]^{\mathcal{T}},\\ \mathbf{Z_w}(m) &= [[\mathbf{z}((m-1)L+1)],\cdots,[\mathbf{z}((m-1)L+L)]]^{\mathcal{T}}\end{aligned} \qquad (6)$$

Subscript 'w' shows signal for one window. Here, $m = 1, 2, ...B$ and $T = BL$. L is the length of a window, where $\mathbf{Y_w} \in R^{L\times N}$ and $\mathbf{Z_w} \in R^{L\times N}$. In compressed sensing (CS), observations are measured using non-adaptive linear measurements:

$$\bar{\mathbf{Y}}_{\mathbf{w}} = \mathbf{\Phi Y_w} = \mathbf{\Phi}(\mathbf{S_w} + \mathbf{Z_w}) = \mathbf{\Phi}(\mathbf{\Psi A_w} + \mathbf{Z_w}) \qquad (7)$$

$$\bar{\mathbf{Z}}_{\mathbf{w}} = \mathbf{\Phi Z_w} \qquad (8)$$

where $\mathbf{\Phi} \in \mathbb{R}^{l\times L}(l << L)$ denotes the sensing matrix consisting of 'l' random orthonormal random vectors. We assume signal from each sensor exhibit a k-sparse representation (as columns of $\mathbf{A_w}$) in a basis $\mathbf{\Psi}$ [12]. Study in [18] showed that under the mild assumption of the eccentricity of dominant eigenvalues, the eigenvectors of covariance matrix $\mathbf{Z_w^T Z_w}/L$ in original domain and $\mathbf{\Phi}(\mathbf{Z_w^T Z_w})\mathbf{\Phi^T}/l$ are related. By exploiting this property, the SNC procedure in (4) can be performed in low-dimensional compressive space as:

$$\bar{\mathbf{S}}_{\mathbf{w}} = \bar{\mathbf{P}}_{\mathbf{w}}\mathbf{\Phi Y_w};\ \bar{\mathbf{P}}_{\mathbf{w}} = (\mathbf{I} - \bar{\mathbf{U}}_{\mathbf{w}}\bar{\mathbf{U}}_{\mathbf{w}}^{\dagger}) \qquad (9)$$

where $\mathbf{I}$ is the identity matrix. $\bar{\mathbf{P}}_{\mathbf{w}}$ and $\bar{\mathbf{U}}_{\mathbf{w}}$ are the projection and the orthogonal matrices respectively in the compressive domain. The processed measurements $\bar{\mathbf{S}}_{\mathbf{w}}$ can be considered as an approximation of CS measurements of signal component $\mathbf{S}_w$. It has been shown that stable recovery of $\mathbf{S}_w$ in terms of its sparse representation $\mathbf{A}_w$ is possible if $\mathbf{\Phi}$ satisfies the restricted isometry property (RIP) and is incoherent with basis

Back to Contents

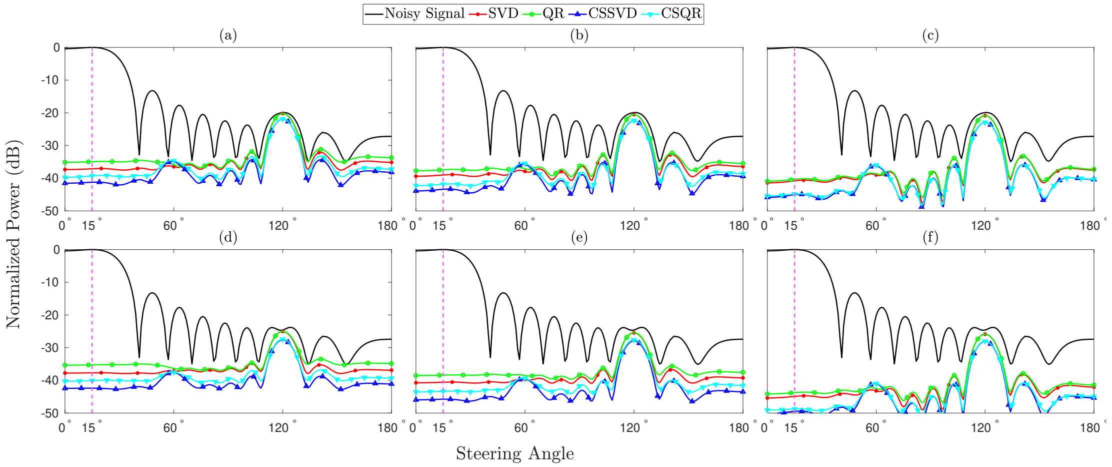


Fig. 2. Beampattern for noisy and recovered NB stationary signal. Target bearing is at 120°, and self-noise bearing is 15°. (a,b,c) at SINR -20dB and (d,e,f) at SINR -25dB using (a,d) top 10, (b,e) top 20 and (c,f) top 30 sampled orthogonal vectors (SOV), respectively. The compression ration $l/L = .2$ is used in case of CSSVD and CSQR methods.

$\mathbf{\Psi}$ [12], [13]. The estimation of the signal matrix requires solving N independent inverse-problems of the form:

$$\begin{aligned} &\text{argmin}\|\bar{\mathbf{S}}_{\mathbf{w}} - \mathbf{\Phi}\mathbf{\Psi}\mathbf{A}_{\mathbf{w}}\|_F^2 \;\; \text{s.t.} \;\; \|\mathbf{a}_i\|_0 \leq k, \\ &\mathbf{A}_w = [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \mathbf{a}_3 \quad \ldots]; \quad \hat{\mathbf{S}}_{\mathbf{w}} = \mathbf{\Psi}\hat{\mathbf{A}}_{\mathbf{w}} \end{aligned} \tag{10}$$

where $\|.\|_0$ denotes the ℓ_0-norm, $\mathbf{a}_i$ are the columns of $\mathbf{A}_w$, and k denotes the cardinality of a vector. (10) is a non-convex problem [19]–[21] and its solution can be obtained by matching pursuit-based greedy algorithms or by relaxing the sparsity constraints and using ℓ_1-norm based solvers instead [12]. In this work, we employ discrete-time cosine transform (DCT) as the sparsifying basis and random-ortho Gaussian matrix as measurement matrix as it satisfies incoherence or RIP conditions with high probability [13]. We denote the two-step procedure in (9) and (10) as compressive self-noise cancellation method. Finally, the complete recovered signal can be obtained by concatenating recovered signal for all windows:

$$\hat{\mathbf{S}} = [\hat{\mathbf{S}}_{\mathbf{w}}(1), \hat{\mathbf{S}}_{\mathbf{w}}(2), \;\cdots \hat{\mathbf{S}}_{\mathbf{w}}(n)]^{\mathcal{T}} \tag{11}$$

This is followed by post-processing using a delay-and-sum beamformer [8] to get beamformed output $\hat{\mathbf{s}}$. Thus, the proposed method reduces time complexity to $\mathcal{O}(L^2N)$ as compared to the high-dimensional subspace-based method ($\mathcal{O}(l^2N)$).

IV. Experimental Results

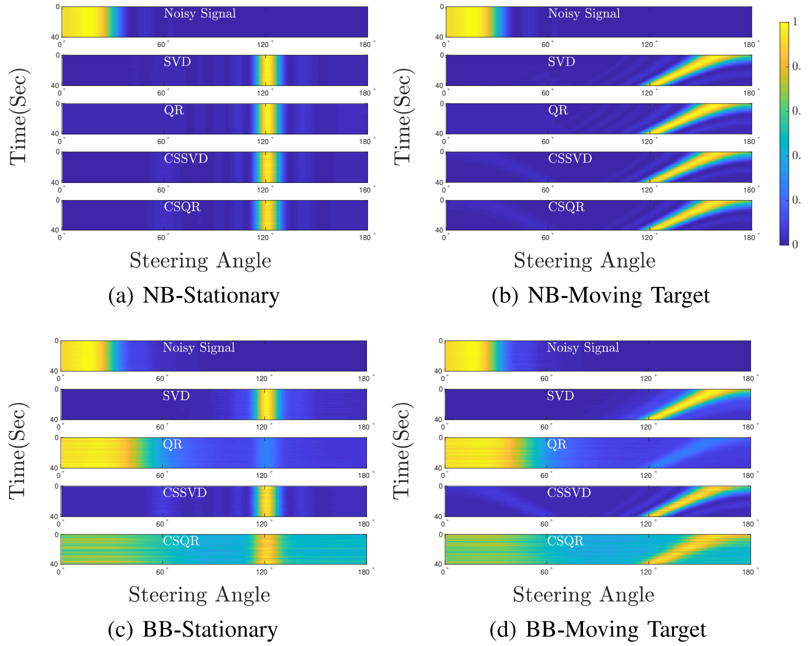
A. Experimental Setup

An experimental study is performed to evaluate the performance of the proposed compressive SNC approach for target detection in underwater acoustics. The simulation is done for Narrowband (NB) and Broadband (BB) Signals, both with stationary and moving targets in the presence of Gaussian ambient noise. Target and self-noise bearing are set to be 120° and 15°, respectively, with moving target bearing varying at 1° per second. The ULA contains 32 sensors that capture the signal at a sampling rate of 12800 Hz over an observation time of the 40s. For signal recovery, we measure and process the signal using non-overlapping rectangular windows of size 80ms. The projection matrix $\bar{\mathbf{P}}$ is estimated from CS measurements using SVD and QR decomposition, where we only sample a few top consecutive orthogonal vectors to form basis $\bar{\mathbf{U}}$. To recover the signal from projected CS samples $\bar{\mathbf{Y}}$ we employ greedy sparse recovery algorithms. In particular, we experimented with compressive sampling matching pursuit (CoSaMP) [22], and orthogonal matching pursuit (OMP) [23] algorithms and found OMP to be more robust in recovery both at low SNR and less number of measurements. The recovered signal from all sensors is beamformed using a delay-and-sum beamformer, and the recovery performance is reported using: 1) plot of normalized beam power as a function of steering angle; and 2) waterfall display (WD) of detected power signature as a function of time and steering angle. We denote the plots/curves corresponding to recovered signal after noise-cancellation in the original domain as SVD or QR and in the compressed domain as CSSVD or CSQR.

B. Results for Narrowband Signal

In this experiment, we consider the case where both target and self-noise are narrowband with a single signal frequency component of 1300 Hz and interference frequency of 1200 Hz. Fig. 2 shows the beampattern corresponding to the recovered target and the observed noisy signal at SINR of -20dB and -25dB using top 10, 20 and 30 sampled orthogonal vectors (SOV). Our baseline here is the SVD method, which can be observed to have good noise-cancellation and target localization with most of the power concentrated in the main lobe centered at 120° (see Fig. 2(a)). The QR method also exhibits comparable target recovery and localization. In contrast, both CSSVD and CSQR methods have good target localization and better noise-cancellation performance in terms of lower power

Back to Contents



(a) NB-Stationary

(b) NB-Moving Target

(c) BB-Stationary

(d) BB-Moving Target

Fig. 3. Waterfall display for noisy and recovered [NB/BB] stationary and moving target at SINR -20 dB using top 20 sampled orthogonal vectors.

in side lobes. Note that the main lobe power in the case of CSSVD/CSQR is slightly less than SVD/QR methods and is a function of the compression ratio l/L (see Section IV-D for more details). We have also analyzed the impact of a number of SOV of $\bar{\mathbf{U}}$ to form a projection matrix $\bar{\mathbf{P}}$. While the target localization is comparable, one can observe that as we sample more vectors, the uniformly distributed side lobe power becomes more concentrated at certain steering angles. These results are consistent for different SINR and both stationary and moving targets. We further evaluate the Self Noise - Power Level Reduction (SN-PLR) and Target Power Loss (TPL). It is the absolute difference in power output (in dB) between the recovered and noisy signals at self-noise and target bearing, respectively. The SN-PLR (TPL) scores at SINR-20 dB and -25 dB are illustrated in Table I and II respectively at $l/L = .2$. Although the CSSVD/CSQR show higher TPL, the target is still being localized with respectable suppression in self-noise (higher SN-PLR compared to SVD/QR). To demonstrate the bearing history of the recovered target at a specific time, we show the WD plots in Fig. 3(a) and (b). In particular, we only show WD plots for moving targets to visualize the temporal behavior better. Observe maximum power at 120° for stationary target and how the signature is localized from 120° to 180° throughout the 40s in WD plots. We see that CSSVD and CSQR methods can recover, localize, and track the target while simultaneously suppressing the ambient and self-noise even for moving targets.

C. Results for Broadband Signal

This experiment considers a more complex broadband case with the frequency range for both target and interference being 100Hz-2000Hz. Due to space constraints, we only report the results using the WD plot in Fig. 3(c) and (d). For stationary targets, it can be observed that both SVD and CSSVD methods have comparable performance in terms of noise cancellation and target localization. However, the QR method is unable to recover the target, which demonstrates that the choice of the orthogonal subspace is crucial. Here the sample vectors do not seem to correspond to self-noise leading to undesirable results. Interestingly, the CSQR method can still locate the target, and although self-noise is not fully canceled, it is distributed along with other bearing angles. We observe similar trends for the case of moving target where the SVD method performs the best, followed by CSSVD, CSQR, and QR methods, respectively. These results demonstrate the advantage of exploiting the sparsity to achieve SNC in the compressive domain.

TABLE I
SN-PLR (TPL) for NB Stationary Target at SINR -20 dB

SOV	**SVD**	**QR**	**CSSVD**	**CSQR**
10	37.12 (0.28)	34.95 (**0.27**)	**41.23** (1.99)	39.31 (1.95)
20	39.02 (0.61)	37.42 (**0.59**)	**43.38** (2.50)	41.85 (2.44)
30	40.81 (**0.93**)	40.35 (**0.93**)	**45.06** (2.99)	44.86 (2.99)

TABLE II
SN-PLR (TPL) for NB Stationary Target at SINR -25 dB

SOV	**SVD**	**QR**	**CSSVD**	**CSQR**
10	37.69 (5.10)	35.26 (**5.05**)	**42.38** (7.68)	40.07 (7.52)
20	40.58 (5.53)	38.40 (**5.48**)	**45.76** (7.69)	43.35 (7.71)
30	44.91 (5.94)	43.77 (**5.92**)	**49.49** (8.16)	48.77 (8.09)

Back to Contents

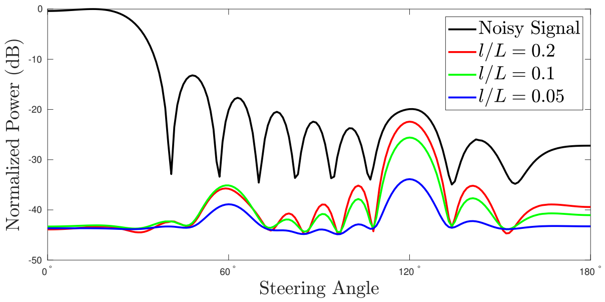


Fig. 4. Beampattern of the noisy & recovered signal at SINR -20dB using top 20 orthogonal vectors as a function of compression ratio.

D. Impact of compression ratio

In this experiment, we assess the recovery performance of the proposed compressive SNC method as a function of compression ratio (l/L). This is important to understand the trade-off between computational complexity using fewer CS samples vs. the target recovery/localization performance. In particular, we consider the case of NB stationary target where the SNC is performed using the CSSVD method at compression ratio of 0.2, 0.1, 0.05, respectively. Fig. 4 shows the beampattern corresponding to the observed noisy signal at SINR of -20dB and the recovered target using top 20 orthogonal vectors. It can be inferred that even with very few measurements, especially l/L =.05, the proposed method is able to localize the target. However, the ability to resolve the main lobe at 120° from the other side lobe improves as the ratio l/L increases.

V. Conclusion

We have presented a CS-based approach for self-noise cancellation and target localization in this work. Consistent with existing studies, we demonstrate the efficacy of the CS approach in exploiting the sparsity of the target for a robust recovery in the case of both narrowband and broadband signals. The novelty of our approach lies in the combination of the subspace-based noise-cancellation approach with CS-based target localization in the presence of self and ambient noise. Self-noise typically has much higher power than target and also exhibits sparse properties. Hence, we first employ null-space projection in the compressive domain to suppress noise followed by conventional CS-based target recovery. Finally, we experimentally demonstrated that working with various orthogonal decomposition methods to estimate noisy subspace in the compressed domain is more robust than working directly in the original high-dimensional signal domain. Our future work will focus on optimizing the sensing matrix for multiple target localization.

References

[1] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Part III*. John Wiley & Sons, Inc., 2001.

[2] C. Yuan, M. R. Azimi-Sadjadi, J. Wilbur, and G. J. Dobeck, “Underwater target detection using multichannel subband adaptive filtering and high-order correlation schemes,” *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 192–205, 2000.

[3] J. D. Tucker and M. R. Azimi-Sadjadi, “Coherence-based underwater target detection from multiple disparate sonar platforms,” *IEEE Journal of Oceanic Engineering*, vol. 36, no. 1, pp. 37–51, 2011.

[4] B. F. Harrison, “The eigencomponent association method for adaptive interference suppression,” *The Journal of the Acoustical Society of America*, vol. 115, no. 5, pp. 2122–2128, 2004.

[5] N. M. Patil and M. U. Nemade, “Audio signal deblurring using singular value decomposition (SVD),” in *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017, pp. 1272–1276.

[6] M. Remadevi, N. Sureshkumar, R. Rajesh, and T. Santhanakrishnan, “Cancellation of towing ship interference in passive sonar in a shallow ocean environment,” *Defence Science Journal*, vol. 72, no. 1, pp. 122–132, 2022.

[7] T.-M. Tu, C.-H. Chen, and C.-I. Chang, “A noise subspace projection approach to target signature detection and extraction in an unknown background for hyperspectral images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 1, pp. 171–181, 1998.

[8] H. L. V. Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Melbourne, FL, USA: John Wiley & Sons, Inc., 2002.

[9] A. Koul, G. Anand, S. Gurugopinath, and K. Nathwani, “Three-dimensional underwater acoustic source localization by sparse signal reconstruction techniques,” in *International Conference on Signal Processing and Communications (SPCOM)*, 2020, pp. 1–5.

[10] M. Ali, A. Koul, and K. Nathwani, “Group delay based re-weighted sparse recovery algorithms for robust and high-resolution source separation in doa framework,” *Interspeech*, pp. 3031–3035, 2021.

[11] A. S. W. Dmitry M. Malioutov, Müjdat Çetin, “A sparse signal reconstruction perspective for source localization with sensor arrays,” *IEEE Transactions on Signal Processing*, vol. 50, no. 8, pp. 3010 – 3022, 2005.

[12] R. G. Baraniuk, “Compressive sensing,” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118 – 121, 2007.

[13] E. J. Candes, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted $\ell 1$ minimization,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[14] P. G. Angeliki Xenaki and K. Mosegaard, “Compressive beamforming,” *Acoustical Society of America*, vol. 136, no. 1, pp. 260–271, 2014.

[15] A. X. C.F.Mecklenbraüker, Peter Gerstoft, “Multiple and single snapshot compressive beamforming,” *Acoustic Society of America*, vol. 138, no. 4, pp. 2003–2014, 2015.

[16] A. Dietrich, A. Albu-Schäffer, and G. Hirzinger, “On continuous null space projections for torque-based, hierarchical, multi-objective manipulation,” in *IEEE International Conference on Robotics and Automation*, 2012, pp. 2978–2985.

[17] F. Liu, L. Peng, M. Wei, P. Chen, and S. Guo, “An improved L1-SVD algorithm based on noise subspace for DOA estimation,” *Progress In Electromagnetics Research C*, vol. 29, pp. 109–122, 2012.

[18] J. E. Fowler, “Compressive-projection principal component analysis,” *IEEE transactions on image processing*, vol. 18, no. 10, pp. 2230–2242, 2009.

[19] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

[20] A. M. Dixon, E. G. Allstot, A. Y. Chen, D. Gangopadhyay, and D. J. Allstot, “Compressed sensing reconstruction: Comparative study with applications to ecg bio-signals,” in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*. IEEE, 2011, pp. 805–808.

[21] A. Majumdar and R. K. Ward, “Compressed sensing of color images,” *Signal Processing*, vol. 90, no. 12, pp. 3122–3127, 2010.

[22] D. Needell and J. A. Tropp, “CoSaMP: iterative signal recovery from incomplete and inaccurate samples,” *Applied and computational harmonic analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[23] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,” in *Asilomar conference on signals, systems and computers*, 1993, pp. 40–44.

Back to Contents

Non-Coherent Discrete Chirp Fourier Transform for Modulated LFM Parameter Estimation

Kaiyu Zhang, Fraser K. Coutts, and John Thompson
Institute for Digital Communications, University of Edinburgh, Edinburgh, EH9 3FG, UK
Email: Kaiyu.Zhang@ed.ac.uk

***Abstract*—Linear frequency modulated (LFM) waveforms play a significant role in both military and civilian detection and ranging applications. Based on LFM, researchers have designed a phase shift keying (PSK) modulated LFM waveform to enable both communication and radar functions simultaneously. Estimating the chirp rate and carrier frequency of data modulated LFM waveforms is therefore of interest in defence applications. In the previous research, the discrete chirp-Fourier transform (DCFT) was proposed to estimate the parameters for non-modulated waveforms. However, the DCFT method is limited with respect to the transform length and the estimation range. Thus, this paper proposes a generalised coherent DCFT that extends the previous DCFT method. We also introduce a novel non-coherent DCFT to improve parameter estimation for PSK modulated LFM waveforms. In the simulation section, this paper discusses the applicability of two modified DCFT methods and demonstrates their superior performance.**

I. Introduction

Target detection is typically a primary task for radar systems. As one of several fundamental radar signals, linear frequency modulation (LFM) waveforms are applied in various scenarios [1]. Furthermore, variants of LFM waveform radar such as the smeared synthesized LFM signal [2] and orthogonal LFM waveform [3] has arisen more recently.

For signal processing at the receiver of the radar, [4] explains the working pattern and the significance of the LFM parameter estimation. To estimate the chirp rate parameters, [5] proposed the discrete chirp-Fourier transform (DCFT) method with a clear theoretical background but with limitations on the detection range and resolution. In a follow-up publication [6], the authors modify the sampling rate in the DCFT to avoid failed detections. In addition, other researchers [7] made modifications based on [5] to improve the performance of the chirp rate parameter estimation. Recent research has investigated applications of the DCFT in other areas, e.g., in compressive sensing [8], cubic chirp parameter estimation [9], and LFM parameter detection [10]. To apply the DCFT to practical problems, [11] proposed an efficient matrix calculation to reduce the complexity of the application of the DCFT. Research in [12] illustrates the detection of a high speed target via the fast DCFT. However, research to date about the application of the DCFT mainly focusses on the original LFM waveform instead of variants of LFM. Thus, this paper also considers phase shift keying (PSK) modulated LFM waveforms as one kind of radar waveform to be characterised.

Beyond the research on the basic LFM waveform, there is major interest in joint radar and communication systems to better utilise the available frequency bands. Researchers in [13] proposed several systems that jointly implement both radar and communication functions. To design joint radar and communication waveforms, [14] discussed the modulation of radar waveforms with specially designed binary sequences prior to transmission. Then, research in [15] introduced the realisation of PSK modulated LFM waveforms and embedded the PSK data into multiple-input multiple-output radar waveforms. Reference [16] designed PSK modulated LFM waveforms for internet of things applications.

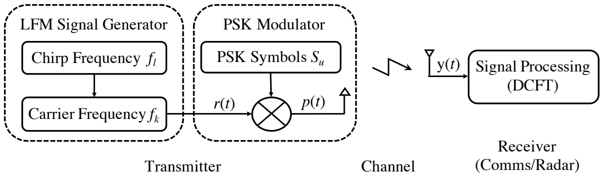


Fig. 1. Joint Radar and Communications System Model.

The novel contributions of this paper are as follows. This paper proposes two variants of the DCFT, the coherent DCFT for the original LFM waveform and the non-coherent DCFT for the PSK modulated LFM signal. First, generalised forms of the DCFT are able to arbitrarily select the length and resolution of the transform depending on prior information of the application scenario. Secondly, for the PSK modulated LFM waveform, the non-coherent DCFT is proposed to provide more robust estimation performance.

The layout of this paper is as follows: Section II describes the system model of this paper; Section III illustrates the basic DCFT method and points out some potential problems; Section IV proposes the coherent and non-coherent variants of the DCFT for the different variants of LFM waveform; in Section V, simulation results are exhibited to compare the performance of these new methods, and Section VI provides conclusions.

II. System Model

In this paper, the system model can be divided into three parts, which are the transmitter, the channel, and the receiver as shown in Fig. 1. To generate the basic LFM waveform, the transmitter continually adjusts the centre carrier frequency f_k (in Hz) where the rate of change of frequency is called the chirp frequency and is denoted as f_l in Hz. Then the PSK modulated LFM waveform can be generated by multiplying the chirp signal with the PSK symbol waveform S_u. This paper

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

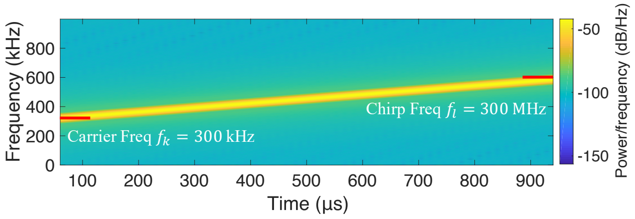


Fig. 2. Spectrogram of 1 ms of the LFM waveform with $f_l = 300\,\text{MHz}$ and $f_k = 300\,\text{kHz}$.

considers an additive white Gaussian noise (AWGN) channel and implements the DCFT at the receiver to estimate the chirp frequency $\tilde{f}_l$ and the carrier frequency $\tilde{f}_k$.

At the transmitter, the instantaneous frequency $f(t)$ in Hz at time t (in seconds) of the LFM waveform is

$$f(t) = f_l t + f_k, \tag{1}$$

where t is in the range $[0, T_c]$ and T_c is the period of the LFM waveform. Then, the conventional complex LFM waveform $r(t)$ as shown in Fig. 1 is written as

$$r(t) = \exp(j(\beta_0 t^2 + \alpha_0 t)), \tag{2}$$

where the relationships between β_0, α_0 and f_l, f_k are

$$\beta_0 = \pi f_l, \qquad \alpha_0 = 2\pi f_k. \tag{3}$$

The discrete LFM time signal $r[n]$ is obtained by sampling $r(t)$ with the sample frequency f_s in Hz and the sample period correspondingly $T_s = 1/f_s$ in seconds. Then $r[n]$ is written as

$$r[n] = \exp(j(\beta_0 (n/f_s)^2 + \alpha_0 (n/f_s))). \tag{4}$$

Fig. 2 shows the spectrogram of 1 ms of the LFM waveform with $f_s = 1\,\text{MHz}$, $f_k = 300\,\text{kHz}$, and $f_l = 300\,\text{MHz}$.

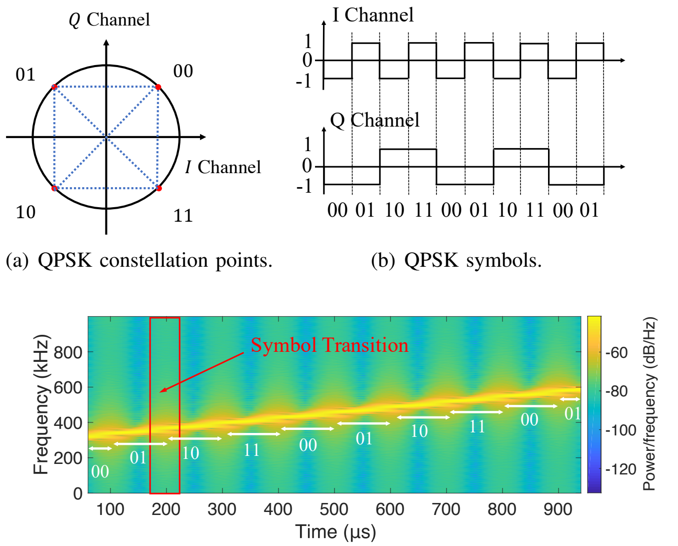
The process of generating modulated symbols is similar to that used in [15] and this paper considers QPSK as the modulation method. A constellation diagram for QPSK is shown in Fig. 3(a) and the QPSK symbol S_u can be one of four options

$$S_u = \exp(j(2u-1)\pi/4)\,, \quad u = 1, 2, 3, 4. \tag{5}$$

Then, the complex QPSK modulated LFM (QPSK-LFM) waveform $p[n]$ is

$$p[n] = S_u r[n]\,. \tag{6}$$

To illustrate the process of the QPSK-LFM generation, we consider 10 QPSK symbols as the example to be transmitted in 1 ms as shown in Fig. 3(b). Then the LFM waveform $r(t)$ as shown in Fig. 2 is multiplied by these QPSK symbols S_u and forms the QPSK-LFM waveform $p(t)$. Fig. 3(c) exhibits the spectrogram of this QPSK-LFM waveform and highlights the symbol transition and the period for each QPSK symbol. Furthermore, this paper considers another scenario that there maybe an offset time T_o between the start of the chirp signal and the first QPSK symbol. In this paper, we assume that T_o is an integer multiple of T_s, and T_o therefore adjusts the relationship between S_u and $r(t)$.



(a) QPSK constellation points.

(b) QPSK symbols.

(c) Spectrogram of 1 ms of the QPSK-LFM waveform with $f_l = 300\,\text{MHz}$ and $f_k = 300\,\text{kHz}$.

Fig. 3. QPSK-LFM Waveform Examples.

We denote the transmitted signal as $x[n]$, which will either be the LFM waveform $r[n]$ or the QPSK-LFM waveform $p[n]$. This waveform passes through the AWGN channel so the received waveform $y[n]$ is

$$y[n] = x[n] + w[n]\,, \tag{7}$$

where $w[n]$ is AWGN distributed on $\mathcal{CN}(0, \sigma^2)$ and σ^2 is the power of the AWGN. The signal-to-noise ratio (SNR) in this paper is defined as $(\text{E}[x^2[n]]/\sigma^2)$, where E[ ] is the expectation symbol.

III. The Traditional DCFT Method

This section mainly introduces the derivation of the traditional DCFT technique and highlights three potential problems that arise in application scenarios.

Based on the DFT method, reference [5] proposes an N-point DCFT technique, which is

$$X[l,k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] W_N^{ln^2+kn}, l, k = 0, 1, \ldots, N-1, \tag{8}$$

where $x[n]$ is derived from the known continuous time domain signal $x(t)$ when $t = n/f_s$, N is the length of $x[n]$, $W_N^p = \exp(-2\pi jp/N)$ and X is the $N \times N$ DCFT output matrix.

The coordinate $(\tilde{l}, \tilde{k})$ corresponding to the largest value in the recovered matrix $X[l,k]$ provides $\tilde{f}_l$ and $\tilde{f}_k$ via

$$\tilde{f}_l = 2f_s^2 \tilde{l}/N\,, \qquad \tilde{f}_k = f_s \tilde{k}/N\,. \tag{9}$$

The resolutions Δf_l and Δf_k for $\tilde{f}_l$ and $\tilde{f}_k$ of this method are

$$\Delta f_l = 2f_s^2/N\,, \qquad \Delta f_k = f_s/N\,. \tag{10}$$

The estimation ranges $(f_l^{\min}, f_l^{\max})$ in Hz and $(f_k^{\min}, f_k^{\max})$ in Hz correspondingly are

$$f_l^{\min} = 0\,\text{Hz}\,, \qquad f_l^{\max} = \frac{1}{N}(N-1)2f_s^2\,\text{Hz}\,, \tag{11}$$

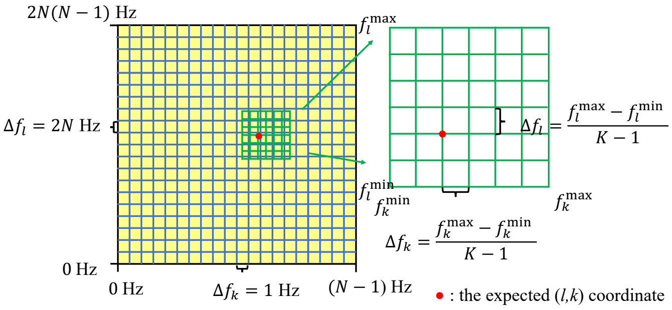

$2N(N-1)$ Hz
$f_l^{\max}$
$\Delta f_l = 2N$ Hz
$\Delta f_l = \dfrac{f_l^{\max} - f_l^{\min}}{K-1}$
$f_l^{\min}$
$f_k^{\min}$
$f_k^{\max}$
$\Delta f_k = \dfrac{f_k^{\max} - f_k^{\min}}{K-1}$
0 Hz
0 Hz
$\Delta f_k = 1$ Hz
$(N-1)$ Hz
● : the expected (l,k) coordinate

Fig. 4. Estimation ranges transformation for the DCFT modification.

$$f_k^{\min} = 0\,\text{Hz}\,, \qquad f_k^{\max} = \frac{1}{N}(N-1)f_s\,\text{Hz}\,. \tag{12}$$

Research in [5] specifies f_s to be proportional to a value of $N^{1/3}$ Hz and emphasises N should be a prime value to avoid aliasing occuring in the DCFT matrix X. By contrast, [6] modifies the DCFT definition by increasing the sample rate to $f_s = N$ Hz to improve the performance. In addition, [7] restricts l and k to integers in the range $[0, N/2)$ to improve the chirp rate resolution.

However, there are still some potential problems in application scenarios. First, when N is a very large value, the process of generating X will lead a very high computational cost. Second, from (10), when the value of f_s is much larger than N, namely $f_s \gg N$, the resolution step size Δf_l and Δf_k will become larger and possibly cause imprecise recovery of X. Furthermore, from (10), (11), and (12), there is a $(2f_s)$ times difference between the values f_l and f_k for the resolution and detection ranges, which means when f_s is a large value, the parameters for f_l and f_k will not be the same order of magnitude and this may cause significant errors.

IV. Novel Modifications to the DCFT

To solve the above problems, this section introduces the coherent DCFT method for the original LFM waveform with a custom estimation range and then proposes the non-coherent DCFT technique to extend the DCFT to handle data modulated LFM waveforms.

A. The Coherent DCFT

First of all, this modified method introduces the length parameter of the DCFT, K, which is set equal to the datasize N in the previous DCFT technique. Through decoupling the identical relationship between the length of DCFT and the number of samples, this method is able to determine the value of K depending on the requirements of the application. According to practical scenarios, [17] illustrates radars are used to detect targets under certain specific bandwidths. Thus, this modified DCFT method is designed for user-specified estimation ranges, $(f_l^{\min}, f_l^{\max})$ and $(f_k^{\min}, f_k^{\max})$. Fig. 4 compares the blue grid in (8) and the modified method on the green grid. This modified method specifically selects the green grid as the estimation ranges by introducing the real-valued coefficients a, b, c, and d. Thus, the (8) can be rewritten into the K-point coherent DCFT as

$$X[l_1, k_1] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] W_K^{(al_1+c)n^2 + (bk_1+d)n}\,. \tag{13}$$

where l_1 and k_1 are integers in the range $[0, K-1]$ while $al_1 + c$ and $bk_1 + d$ might be fraction values.

From the known customised range $(f_l^{\min}, f_l^{\max})$ and $(f_k^{\min}, f_k^{\max})$ and the similar deduction procedure in (11) and (12), the detecting ranges of this method are modified as

$$f_l^{\min} = 2f_s^2 c/K, \qquad f_l^{\max} = 2f_s^2(a(K-1)) + c)/K, \tag{14}$$

$$f_k^{\min} = f_s d/K, \qquad f_k^{\max} = f_s(b(K-1) + d)/K\,. \tag{15}$$

Then the coefficients a, b, c, and d are defined as

$$a = \frac{K(f_l^{\max} - f_l^{\min})}{2f_s^2(K-1)}\,, \qquad c = \frac{Kf_l^{\min}}{2f_s^2}\,, \tag{16}$$

$$b = \frac{K(f_k^{\max} - f_k^{\min})}{f_s(K-1)}\,, \qquad d = \frac{Kf_k^{\min}}{f_s}\,. \tag{17}$$

Via the coordinate of the largest magnitude entry of $X[l_1, k_1]$ at position, $(\tilde{l_1}, \tilde{k_1})$, estimates of f_l and f_k are

$$\tilde{f_l} = 2f_s^2(a\tilde{l_1} + c)/K\,, \quad \tilde{f_k} = f_s(b\tilde{k_1} + d)/K\,. \tag{18}$$

The resolution values Δf_l and Δf_k of this method are

$$\Delta f_l = \frac{f_l^{\max} - f_l^{\min}}{K-1}\,, \qquad \Delta f_k = \frac{f_k^{\max} - f_k^{\min}}{K-1}\,. \tag{19}$$

Correspondingly, the available values of $\tilde{f_l}$ and $\tilde{f_k}$, R_{f_l} and R_{f_k} where l and k are integers in the range $[0, \ldots, K-1]$ are

$$R_{f_l} = f_l^{\min} + l\frac{f_l^{\max} - f_l^{\min}}{K-1}\,, \; R_{f_k} = f_k^{\min} + k\frac{f_k^{\max} - f_k^{\min}}{K-1}\,. \tag{20}$$

Compared to the resolution and the detection ranges in the previous DCFT method in equations (10), (11), and (12), those parameters for this modified method are updated as equations (14), (15), and (19). For the computational complexity when $f_s = N$ Hz, the one of the previous DCFT method is N^3 while the coherent DCFT method is NK^2. According to the demand of the application scenario, this method can restrict the estimation ranges and select K to avoid computational waste through using inappropriate estimation ranges for f_l and f_k.

B. The Non-Coherent DCFT

In addition to the LFM waveform, the QPSK-LFM waveform couples S_u with $r(t)$. However, when we apply the coherent DCFT directly to $p(t)$, the modulation signal S_u can significantly degrade the estimation result. To eliminate the influence caused by PSK modulation symbols, this subsection proposes the K-point non-coherent DCFT. To apply the non-coherent DCFT to the QPSK-LFM waveform, the number of symbol blocks for the period of the LFM waveform should be known in order to divide it into different symbols. Assuming

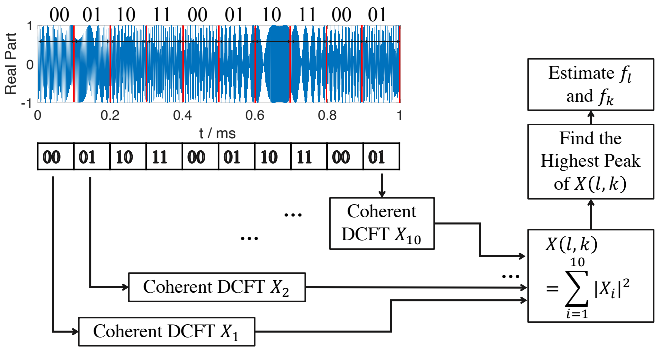


Fig. 5. Process of the non-coherent DCFT.

the number of symbol blocks is M and based on (13) the non-coherent DCFT is

$$X[l_2,k_2] = \frac{1}{\sqrt{N}} \sum_{i=1}^{M} \left| \sum_{n=m_i}^{m_i+n_i-1} x[n] W_K^{(al_2+c)n^2+(bk_2+d)n} \right|, \tag{21}$$

where $n_i = \lfloor N/M \rfloor$ is the number of samples in the ith symbol block, m_i is the initial sample value for the ith symbol block, $m_1 = 0$ and $m_i = m_{i-1} + n_{i-1}$ and l_2 and k_2 are integers in the range $[0, K-1]$.

Fig. 5 shows the real part of the time domain waveform used in Fig. 3(c) and the main processing steps of the non-coherent DCFT for the example waveform when $M = 10$, $n_i = 100$, and $m_i = 100(i-1)$. Then through $(\tilde{l_2}, \tilde{k_2})$, $\tilde{f_l}$ and $\tilde{f_k}$ can be recovered via the same process as (18).

V. Simulations and Discussion

This section discusses the properties and estimation errors for the proposed coherent and non-coherent DCFT methods and describes some simulation results of two above methods. The two modified DCFT methods proposed in Section IV fix three potential problems mentioned in the Section III and can select an appropriate value of K to achieve the resolution and detection range that is needed. Furthermore, these two DCFT methods can decrease the computational complexity, improve the resolution, and estimate $\tilde{f_l}$ and $\tilde{f_k}$ simultaneously.

Estimation errors for both $\tilde{f_l}$ and $\tilde{f_k}$ in the DCFT can be caused by the LFM signal having non-integer values of $(\tilde{l}, \tilde{k})$. In the following simulations, to avoid this effect, the ground truth of (f_l, f_k) is randomly generated from (20). Results were averaged over 10^4 Monte Carlo instances in each simulation and the key simulation parameters are as shown in Table I.

When the input waveform of the DCFT $x(t)$ is the QPSK-LFM waveform $p(t)$ with different numbers of symbol blocks M from 1 to 100, the coherent DCFT becomes unable to estimate $\tilde{f_l}$ and $\tilde{f_k}$ correctly. Fig. 6 shows that the normalised mean square error (NMSE) of f_l and f_k increases significantly as the symbol rate increases. Moreover, for a small number of symbols, as there are only a few phase transitions present in the LFM waveform, the coherent DCFT is still able to accurately determine f_l and f_k; however this method fails at higher symbol rates. Our simulation tests also show that the traditional DCFT fails in this scenario as well. To deal with this circumstance, we switch from the coherent DCFT method into the non-coherent DCFT method for the QPSK-LFM waveform. In Fig. 7-9, the results calculate the likelihood of correctly recovering the ground truth coordinate (l, k) and are plotted as the probability of accurate (l, k) recovery.

TABLE I
Key Simulation Parameters

Name	Value
K	70
$(f_l^{\max}, f_l^{\min})$	$(10, 500)$ MHz
$(f_k^{\max}, f_k^{\min})$	$(10, 500)$ kHz
T_c	1 ms
f_s	1 MHz
T_s	1 μs
SNR range	$[-30\,\text{dB}, \ldots, 10\,\text{dB}]$
Modulation type	QPSK
Baud rates	$[0, \ldots, 100]$ kbaud
T_o	$[0, \ldots, 9]T_s$

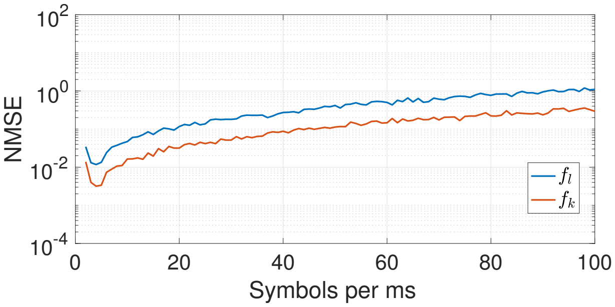


Fig. 6. Coherent DCFT recovery for different symbol rates.

To assess the influence of AWGN, we respectively apply the two proposed DCFT methods to $y(t)$ and obtain the simulation results shown in Fig. 7 and Fig. 8. From Fig. 7 and Fig. 8, the non-coherent DCFT exhibits a better performance under the same circumstance. For 60 symbols per millisecond, the prob. of recovery for the coherent DCFT is only 1.88% even for infinite SNR in Fig. 7. In Fig. 8 with the non-coherent DCFT, the prob. of recovery increases to 23.97% for $-10\,\text{dB}$ SNR and 97.83% for $0\,\text{dB}$ SNR. As the number of symbol blocks and the SNR increases, Fig. 7 shows a dramatic reduction in accurate recovery and thus proves the coherent DCFT is not suited to estimate the parameters of the QPSK-LFM signal. In Fig. 8, the dominant factor to estimate parameters is the SNR. The non-coherent DCFT recovery performance degrades as the SNR reduces or the symbol rate increases.

To further illustrate the robustness of the non-coherent DCFT, Fig. 9 simulates an imperfect synchronisation scenario with different offset times T_o as shown in Table I. The offset time T_o delays the location of the QPSK data modulation transitions compared to the coherent DCFT processor blocks shown in Fig. 5. In this simulation, the number of symbol blocks M is fixed as 100 and Fig. 9 shows that the higher accuracy is achieved at the lowest or highest sample offsets

Back to Contents

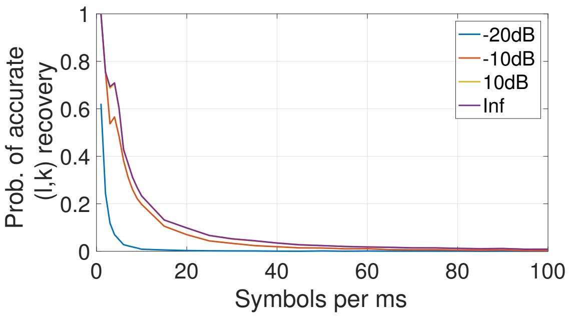


Fig. 7. Coherent DCFT simulation for QPSK-LFM with different SNRs.

compared to the near 0 result at the middle sample offset. This means the influence of imperfect synchronisation is much more severe than the presence of AWGN and should be avoided in real-world applications. Conversely, the non-coherent DCFT has the strong robustness for the small offsets with the over 90% accuracy at the 1 or 9 samples offset.

VI. Conclusion

This paper proposes a coherent variant of the DCFT to circumvent the inflexibilities of the standard DCFT. The proposed generalised coherent DCFT method is able to arbitrarily design the estimation range and determine the length of the DCFT based on prior information. Such an approach is more suitable for real-world applications.

Furthermore, this paper introduces the non-coherent DCFT for detecting PSK modulated LFM waveforms. Through simulations, this paper shows that the proposed coherent DCFT is able to recover both the chirp frequency and the carrier frequency parameters of LFM, but this method yields a high NMSE for the PSK modulated LFM waveforms. However, the non-coherent DCFT method can recover the chirp parameters of the PSK modulated LFM waveform with higher accuracy. In addition, this paper discusses the performance of the proposed DCFT methods under different values of SNR and varying symbol offsets. Simulation results have shown that the accuracy of recovery can remain high at a high SNR for a small synchronisation error.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing. The authors gratefully acknowledge insightful discussions with Prof. Bernie Mulgrew.

References

[1] M. A. Richards, *Fundamentals of radar signal processing*, 2nd ed. Chicago, Ill: McGraw-Hill Education LLC., 2014.

[2] A. Youssef *et al.*, "A novel smeared synthesized LFM TC-OLA radar system: Design and performance evaluation," *IEEE Access*, vol. 7, pp. 18 574–18 589, 2019.

[3] D. Dash and V. Jayaraman, "Ambiguity function analysis for orthogonal-LFM waveform based multistatic radar," *IEEE Sensors Letters*, vol. 5, no. 12, pp. 1–4, 2021.

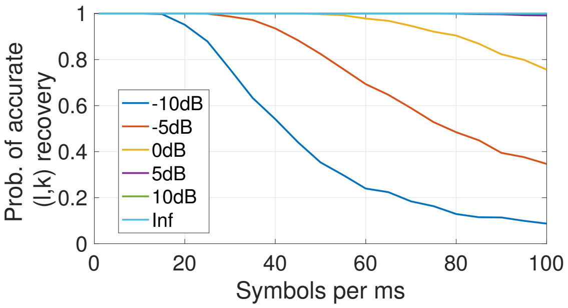


Fig. 8. Non-coherent DCFT recovery simulation for QPSK-LFM with different SNRs.

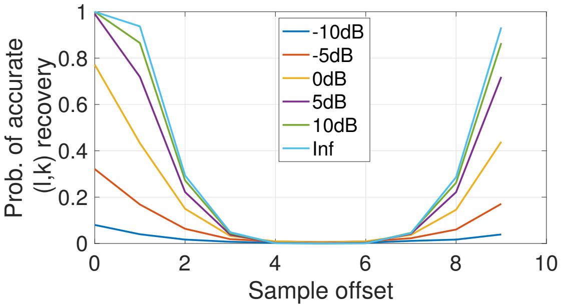


Fig. 9. Synchronisation error simulation for QPSK-LFM with different SNRs.

[4] F. Gini *et al.*, *Waveform Design and Diversity for Advanced Radar Systems*, ser. IET radar, sonar and navigation series. IET, 2012, vol. 22.

[5] X. Xia, "Discrete chirp-Fourier transform and its application to chirp rate estimation," *IEEE Trans. on Signal Process.*, vol. 48, no. 11, pp. 3122–3133, 2000.

[6] X. Guo *et al.*, "Comments on discrete chirp-Fourier transform and its application to chirp rate estimation [with reply]," *IEEE Trans. Signal Process.*, vol. 50, no. 12, pp. 3115–3116, 2002.

[7] P. Fan and X. Xia, "A modified discrete chirp-Fourier transform scheme," in *WCC 2000 - ICSP 2000.*, vol. 1, 2000, pp. 57–60 vol.1.

[8] L. A. A. Irkhis and A. K. Shaw, "Compressive chirp transform for estimation of chirp parameters," in *27th EUSIPCO*, 2019, pp. 1–5.

[9] L. Wu *et al.*, "ISAR imaging of targets with complex motion based on discrete chirp Fourier transform for cubic chirps," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 4201–4212, 2012.

[10] K. Zhang *et al.*, "Detecting LFM parameters in joint communications and radar frequency bands," in *2021 SSPD*, 2021, pp. 1–5.

[11] C. Aceros-Moreno and D. Rodriguez, "Fast discrete chirp Fourier transforms for radar signal detection systems using cluster computer implementations," in *48th MWSCAS*, 2005, pp. 1047–1050 Vol. 2.

[12] X. Huang *et al.*, "Ground-based radar detection for high-speed maneuvering target via fast discrete chirp-Fourier transform," *IEEE Access*, vol. 7, pp. 12 097–12 113, 2019.

[13] F. Liu *et al.*, "Joint radar and communication design: Applications, state-of-the-art, and the road ahead," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3834–3862, 2020.

[14] G. Cui *et al.*, *Radar Waveform Design based on Optimization Theory*, ser. Radar, Sonar and Navigation. IET, 2020.

[15] M. Bekar *et al.*, "Joint MIMO radar and communication system using a PSK-LFM waveform with TDM and CDM approaches," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6115–6124, 2021.

[16] R. Xie *et al.*, "Waveform design for LFM-MPSK-based integrated radar and communication toward IoT applications," *IEEE Internet Things J.*, vol. 9, no. 7, pp. 5128–5141, 2022.

[17] F. Engels *et al.*, "Automotive radar signal processing: Research directions and practical challenges," *IEEE J. Sel. Top. Signal Process.*, vol. 15, no. 4, pp. 865–878, 2021.

Back to Contents

Unsupervised Expectation Propagation Method for Large-Scale Sparse Linear Inverse Problems

Dan Yao, Stephen McLaughlin, Yoann Altmann
School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, UK

***Abstract*—This paper addresses the estimation of large-scale sparse coefficients from noisy linear measurements using Expectation Propagation (EP) method for unsupervised approximate Bayesian inference. In the Bayesian model, the Laplace prior, Mixture of two Gaussians (MoG2) prior, and Spike-and-Slab (SaS) prior are adopted respectively as the sparsity-promoting priors of the unknown sparse parameter. In solving high-dimensional linear inverse problems, the proposed EP method directly provides the approximate minimum mean squared error (MMSE) estimate and the approximate posterior uncertainty by an approximating posterior distribution. Furthermore, to tackle the challenging problem of hyperparameter tuning, the EP posterior approximation is embedded in a variational Expectation Maximization (EM) approach to allow for unsupervised hyperparameter estimation. Experiments are conducted on synthetic datasets, including an imaging deconvolution problem, to illustrate the efficiency of the proposed unsupervised EP method and the advantage of using MoG2 and SaS priors in solving sparse linear inverse problems.**

***Index Terms*—Unsupervised approximate Bayesian inference, Expectation Propagation, large-scale problem, sparse linear model, sparsity prior, hyperparameter estimation**

I. Introduction

Many media types in digital signal and image processing can be sparsely represented in transform-domains, and a large number of processing tasks in this field can be posed as solving the sparse linear inverse problems, where one seeks to recover a sparse signal from the degraded measurements [1]. Such problems become large-scale when the model parameters are high-dimensional. To solve the large-scale sparse linear inverse problems, the Bayesian framework provides a natural and versatile way by incorporating prior knowledge in different sparsity-promoting prior models. Combining these priors with the likelihood of the measurements, the sparse solution can then be inferred from its posterior distribution.

Uncertainty quantification (UQ) is critical in defence applications when decision-making and planning are based on the current estimate [2], [3]. While Bayesian methods feature a strong ability for uncertainty quantification, quantifying the uncertainty for large-scale sparse linear inverse problems remains challenging due to the high dimensionality [4]. As a result, most of the existing Bayesian approaches to such large-scale problems are restricted to the single point estimation without computing further uncertainty. Maximum A Posteriori (MAP) is the most widely used point estimator as it can be recast as an optimization problem and use powerful convex optimization tools when the posterior distribution is log-concave. An alternative to the MAP estimator is the Minimum Mean Squared Error (MMSE) estimator or posterior mean (when it exists), which is often associated with the posterior covariance matrix for uncertainty quantification. Unfortunately, exact computation of the MMSE estimate and posterior covariance from a high-dimensional posterior distribution is challenging as it involves high-dimensional integration, even if the parameter to be estimated is sparse.

Approximate Bayesian methods have been developed to bypass the exact computation of such posterior summary statistics by approximating the exact posterior distributions. This class of methods proposes to find approximation to the exact posterior distributions in order to improve the computational efficiency. In recent years, Expectation Propagation (EP) [5] has received growing attention [6], [7] as an approximate method, and it complements methods based on Variational Bayes (VB). There have been a number of EP methods in the literature that provide efficient solutions to sparse linear inverse problems [8]–[10]. These EP methods are applied to linear models with sparsity-promoting priors such as Laplace prior [8], Spike-and-Slab (SaS) prior [9], or Mixture of two Gaussians (MoG2) prior [10]. Yet, these EP methods with different models (likelihood+prior) have not been considered in a single study for large-scale problems. This paper compares these sparsity-promoting priors, combined with an approximate Bayesian method, for unsupervised inference, i.e. to also estimate the prior hyperparameter(s). A multivariate Gaussian distribution and a multivariate Bernoulli distribution are chosen to approximate the posterior distributions of the unknown high-dimensional sparse parameters of interest. The approximating distributions are further embedded in a variational Expectation Maximization (EM) approach to estimate the sparsity-promoting prior hyperparameters.

The rest of the paper is organized as follows. Section II presents the large-scale sparse linear inverse problem to be addressed and the associated Bayesian model. Section III describes the proposed EP method to solve the problem. Section IV evaluates the performance of the proposed EP method on synthetic experiments. Conclusion and future work are finally reported in Section V.

This work was supported by the Royal Academy of Engineering under the Research Fellowship scheme RF201617/16/31, the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

II. Problem Formulation and Bayesian Model

A. Large-scale sparse linear inverse problem

The large-scale sparse linear inverse problem investigated in this paper consists of estimating a high-dimensional sparse vector $\boldsymbol{x} \in \mathbb{R}^{N\times 1}$ from the observation $\boldsymbol{y} \in \mathbb{R}^{M\times 1}$ of the form

$$\boldsymbol{y} = \mathbf{A}\boldsymbol{x} + \boldsymbol{n}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{M\times N}$ is a known linear operator and $\boldsymbol{n} \in \mathbb{R}^{M\times 1}$ represents additive white Gaussian noise with variance σ^2. Given $\boldsymbol{y}$, $\mathbf{A}$, and σ^2, the proposed EP method casts the estimation problem as a Bayesian inference problem using the posterior distribution of $\boldsymbol{x}$. The Bayesian model is constructed by the combination of a Gaussian likelihood function and different sparsity-promoting priors for $\boldsymbol{x}$, as will be presented next.

B. Bayesian model

1) Gaussian likelihood function: given the unknown parameter $\boldsymbol{x}$, the observation $\boldsymbol{y}$ follows a Gaussian distribution and the likelihood function $f_{y|x}(\boldsymbol{y}|\boldsymbol{x})$ is given by

$$f_{y|x}(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{y}; \mathbf{A}\boldsymbol{x}, \sigma^2\mathbf{I}). \tag{2}$$

2) Sparsity-promoting priors: the prior distribution of $\boldsymbol{x}$, denoted by $f_x(\boldsymbol{x}|\boldsymbol{\theta})$, is parameterized by a hyperparameter $\boldsymbol{\theta}$. In the following, we consider three different sparsity-promoting models for $f_x(\boldsymbol{x}|\boldsymbol{\theta})$. To simplify the notation, in this section the hyperparameters of the three prior models are common to all the elements x_n ($\forall n = 1, \dots, N$) in $\boldsymbol{x}$. However, it is also possible to use different hyperparameters for each elements, or groups of elements. This will be further discussed in Section III-B.

Laplace prior: a Laplace distribution is classically adopted as a sparsity-promoting prior [8] with a scalar hyperparameter $\boldsymbol{\theta} := \lambda > 0$ and $f_x(\boldsymbol{x}|\boldsymbol{\theta})$ is expressed as

$$f_x(\boldsymbol{x}|\boldsymbol{\theta}) = \prod_{n=1}^{N} \frac{1}{2\lambda} e^{-\frac{|x_n|}{\lambda}}. \tag{3}$$

Mixture of two Gaussians (MoG2) prior: when $\boldsymbol{x}$ denotes for instance the coefficients of natural images in the Fourier or wavelet domains [11], the mixture prior model has been shown to be a more appropriate choice than the Laplace prior as it is capable of capturing better the *inactive* (nearly zero) and *active* (non-zero) states [12]. The MoG2 prior model with hyperparameter $\boldsymbol{\theta} := (\pi_0, v_1, v_2)$ of the form

$$f_x(\boldsymbol{x}|\boldsymbol{\theta}) = \prod_{n=1}^{N} \pi_0 \mathcal{N}(x_n; 0, v_1) + (1-\pi_0)\mathcal{N}(x_n; 0, v_2) \tag{4}$$

is employed as the sparsity-promoting prior in this work as it is a conjugate prior of the Gaussian likelihood function in (2), and it does not require the setting of a lot of hyperparameters when compared to the mixture of more components. The hyperparameter $\pi_0 \in [0, 1]$ is the prior probability of x_n ($n = 1, \dots, N$) being significantly different from zero, and v_1, v_2 are set to be "large and small", respectively [13].

Spike-and-Slab (SaS) prior: the SaS prior can be considered as a degenerate case of the MoG2 prior by replacing $\mathcal{N}(.; 0, v_2)$ using a Dirac delta function $\delta(.)$, i.e.

$$f_x(\boldsymbol{x}|\boldsymbol{\theta}) = \prod_{n=1}^{N} \pi_0 \mathcal{N}(x_n; 0, v_1) + (1-\pi_0)\delta(x_n). \tag{5}$$

In practice, π_0 in (4) and (5) is usually unknown and can be modeled by a binary random variable z_n using a Bernoulli distribution $Bern(z_n|\pi_0)$, where $Bern(z_n|\pi_0) = z_n\pi_0 + (1 - z_n)(1-\pi_0)$. A binary vector $\boldsymbol{z} = \{z_n\}_{n=1}^N \in \mathbb{R}^{N\times 1}$ is thus introduced in (4) and (5), and the prior for $\boldsymbol{z}$ is defined as

$$f_z(\boldsymbol{z}|\pi_0) = \prod_{n=1}^{N} Bern(z_n|\pi_0). \tag{6}$$

To prevent numerical overflow, π_0 is often computed using a logistic function $\pi_0 = \sigma(p_0) = \frac{1}{1+\exp(-p_0)}$.

Combining (4) (or (5)) and (6), the mixture prior adopted in the Bayesian model becomes

$$f(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}) = f_x(\boldsymbol{x}|\boldsymbol{z}) f_z(\boldsymbol{z}|\pi_0), \tag{7}$$

with $f_x(\boldsymbol{x}|\boldsymbol{z}) = \prod_{n=1}^{N} z_n\mathcal{N}(x_n; 0, v_1) + (1-z_n)\mathcal{N}(x_n; 0, v_2)$ or $f_x(\boldsymbol{x}|\boldsymbol{z}) = \prod_{n=1}^{N} z_n\mathcal{N}(x_n; 0, v_1) + (1-z_n)\delta(x_n)$.

3) Posterior distribution: combining the Gaussian likelihood $f_{y|x}(\boldsymbol{y}|\boldsymbol{x})$ in (2) with the sparsity-promoting prior in (3) or (7), the posterior distribution of $\boldsymbol{x}$ or $(\boldsymbol{x}, \boldsymbol{z})$ is given by

$$p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto f_{y|x}(\boldsymbol{y}|\boldsymbol{x}) f_x(\boldsymbol{x}|\boldsymbol{\theta}), \tag{8}$$

$$p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{y}, \boldsymbol{\theta}) \propto f_{y|x}(\boldsymbol{y}|\boldsymbol{x}) f(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}). \tag{9}$$

When the dimensions of $\boldsymbol{y}$ and $\boldsymbol{x}$ is large, exact computation of the posterior distributions is practically intractable, as it involves costly integral over the high-dimensional $\boldsymbol{x}$ and also the discrete variable $\boldsymbol{z}$ when using prior in (7). While $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta})$ in (8) is log-concave and its unique mode can be found by convex optimisation, estimating the posterior covariance still requires computing high-dimensional integrals or Monte Carlo approximation.

EP is applied here to approximate the intractable posterior distribution by a simpler tractable distribution, and the MMSE estimate of $\boldsymbol{x}$ (or $(\boldsymbol{x}, \boldsymbol{z})$) as well as the uncertainty are approximated via approximate Bayesian inference from the approximating posterior distribution, as will be presented in the next section.

III. Unsupervised EP Method for Large-Scale Sparse Linear Inverse Problem

In this section, a novel unsupervised EP method is proposed to provide the approximate solution when high-dimensional integrals are intractable in Bayesian posterior inference. The proposed EP method consists of two procedures following the classical EM framework, i.e. the E-step and M-step. In the E-step, given the hyperparameter $\boldsymbol{\theta}$, an approximating posterior distribution is found by EP, and in the M-step, the EP posterior approximation is used to estimate $\boldsymbol{\theta}$ by maximizing the marginal likelihood.

Back to Contents

A. Posterior approximation by EP given hyperparameter $\boldsymbol{\theta}$

Suppose for now that the hyperparameter $\boldsymbol{\theta}$ is fixed. The exact posterior distribution $p(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ (or $p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{y},\boldsymbol{\theta})$) can be seen as the product of two (or three) factors, and the posterior approximation by EP is formed by the product of approximating distributions for each of the exact factors,

$$\begin{aligned} q_{x,1}(\boldsymbol{x})q_{x,0}(\boldsymbol{x}) &\approx f_y(\boldsymbol{y}|\boldsymbol{x})f_x(\boldsymbol{x}|\boldsymbol{\theta}), \\ q_{x,1}(\boldsymbol{x})q_0(\boldsymbol{x},\boldsymbol{z})q_{z,0}(\boldsymbol{z}) &\approx f_y(\boldsymbol{y}|\boldsymbol{x})f_x(\boldsymbol{x}|\boldsymbol{z})f_z(\boldsymbol{z}|\pi_0). \end{aligned} \tag{10}$$

Note that the factor $f_y(\boldsymbol{y}|\boldsymbol{x})$, which implicitly depends on the observation $\boldsymbol{y}$, is seen as a function of the unknown parameter $\boldsymbol{x}$. To ensure the posterior approximation for $p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{y},\boldsymbol{\theta})$ remains tractable, $q_0(\boldsymbol{x},\boldsymbol{z})$ is further factorized into $q_0(\boldsymbol{x},\boldsymbol{z}) = q_{x,0}(\boldsymbol{x})q_{z,1}(\boldsymbol{z})$ using a mean-field approximation [14]. In EP, the approximating distributions are found by minimizing successively the following Kullback-Leibler (KL) divergences

$$\min_{q_{x,1}(\boldsymbol{x})} KL\left(f_{y|x}(\boldsymbol{y}|\boldsymbol{x})q_0(\boldsymbol{x})||Q_x(\boldsymbol{x})\right), \tag{11a}$$

$$\min_{q_{x,0}(\boldsymbol{x})q_{z,1}(\boldsymbol{z})} KL\left(f_x(\boldsymbol{x}|\boldsymbol{z})q_{x,1}(\boldsymbol{x})q_{z,0}(\boldsymbol{z})||Q_x(\boldsymbol{x})Q_z\left(\boldsymbol{z}\right)\right), \tag{11b}$$

$$\min_{q_{z,0}(\boldsymbol{z})} KL\left(f_z(\boldsymbol{z}|\pi_0)q_{z,1}(\boldsymbol{z})||Q_z(\boldsymbol{z})\right), \tag{11c}$$

where $q_{x,i}(\boldsymbol{x})$, $q_{z,i}(\boldsymbol{z})$, $\forall i \in (0;1)$ are parameterized by

$$q_{x,i}(\boldsymbol{x}) \propto \mathcal{N}(\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i),\ q_{z,i}(\boldsymbol{z}) = Bern(\boldsymbol{z}|\sigma(\boldsymbol{p}_i)), \tag{12}$$

and

$$Q_x(\boldsymbol{x}) \propto q_{x,1}(\boldsymbol{x})q_{x,0}(\boldsymbol{x}), Q_z(\boldsymbol{z}) \propto q_{z,1}(\boldsymbol{z})q_{z,0}(\boldsymbol{z}). \tag{13}$$

In (11b), $q_{x,0}(\boldsymbol{x})$ and $q_{z,1}(\boldsymbol{z})$ are jointly estimated when minimizing the KL divergence. For $f_x(\boldsymbol{x}|\boldsymbol{\theta})$ in (3), (11b) and (11c) reduce to $\min\limits_{q_{x,0}(\boldsymbol{x})} KL(f_x(\boldsymbol{x}|\boldsymbol{\theta})q_{x,1}(\boldsymbol{x})||Q_x(\boldsymbol{x}))$.

In each iteration, $(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$, $(\boldsymbol{\mu}_0,\boldsymbol{\Sigma}_0)$, $\boldsymbol{p}_1$, $\boldsymbol{p}_0$ are updated sequentially by matching the expected value of the sufficient statistics of the first KL argument to that of the second argument, and the matched second KL argument is then used to update the parameters of the corresponding KL minimizers. To avoid large matrix inversion, $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_0$ are constrained to be diagonal, $\Sigma_1 := \text{diag}(\boldsymbol{\Sigma}_1)$, $\Sigma_0 := \text{diag}(\boldsymbol{\Sigma}_0)$. The update procedures are presented in the following:

update of $(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$: the update of $(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)$ requires large matrix inversion. Here we employ an efficient strategy in the update without costly matrix inversion. The details are omitted for lack of space and interested readers are referred to [9], [15].

update of $(\boldsymbol{\mu}_0,\boldsymbol{\Sigma}_0)$, $\boldsymbol{p}_1$: $\boldsymbol{p}_1$ is not involved in minimizing $KL(f_x(\boldsymbol{x}|\boldsymbol{\theta})q_{x,1}(\boldsymbol{x})||Q_x(\boldsymbol{x}))$ for the Laplace prior in (3). Given the structure of $q_{x,0}(\boldsymbol{x})q_{z,1}(\boldsymbol{z})$ and the Bayesian model, the KL factorizes over the N elements of $\boldsymbol{x}$, $\boldsymbol{z}$ and the parameters w.r.t. x_n, z_n $(n = 1,\ldots,N)$ can be processed independently. For the three priors, the first argument of the KL divergence is a mixture of: (i) two 1-dimensional (1D) Truncated Gaussian distribution (for Laplace prior), (ii) two 1D Gaussian distributions (for MoG2 prior), and (iii) spike-and-slab distribution (for SaS prior). The weight ω_j, and the expected values $(\mathbb{E}_j[x_n],\mathbb{E}_j[x_n^2])$, $\mathbb{E}_j[z_n]$, $\forall j \in (1;2)$ of each mixture component can be computed analytically, such that the expected values of $(\mathbb{E}[x_n],\mathbb{E}[x_n^2])$, $\mathbb{E}[z_n]$ admit closed-form expressions. After matching $(\mathbb{E}[x_n],\mathbb{E}[x_n^2])$ to the first and second order moments of $Q_x(x_n)$, and matching $\mathbb{E}[z_n]$ to the moment of $Q_z(z_n)$ respectively. $(\mu_{0,n},\Sigma_{0,n})$ and $p_{1,n}$ are then updated by the newly updated moments of $Q_x(x_n)$ and $Q_z(z_n)$. The same update rule is applied to update the N elements of $(\boldsymbol{\mu}_0,\boldsymbol{\Sigma}_0)$ in parallel.

update of p_0: since $q_{z,0}(\boldsymbol{z})$ and $f_z(\boldsymbol{z}|\pi_0)$ are both Bernoulli distribution and p_0 is assumed common to all the elements in $f_z(\boldsymbol{z}|\pi_0)$, no update is needed and p_0 remains unchanged as the initialized value.

Upon convergence, the EP posterior approximations for $p(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ in (8) and $p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{y},\boldsymbol{\theta})$ in (9) are obtained by

$$\begin{aligned} Q_x(\boldsymbol{x}) &\propto q_{x,1}(\boldsymbol{x})q_{x,0}(\boldsymbol{x}), \\ Q_x(\boldsymbol{x})Q_z(\boldsymbol{z}) &\propto q_{x,1}(\boldsymbol{x})q_{x,0}(\boldsymbol{x})q_{z,1}(\boldsymbol{z})q_{z,0}(\boldsymbol{z}). \end{aligned} \tag{14}$$

$Q_x(\boldsymbol{x})$ is a multivariate Gaussian $\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$, and $Q_z(\boldsymbol{z})$ is a multivariate Bernoulli distribution $Bern(\boldsymbol{z}|\sigma(\boldsymbol{p}))$, whose parameters are computed by

$$\begin{aligned} &\boldsymbol{\Sigma} = \left(\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_0^{-1}\right)^{-1},\ \boldsymbol{\mu} = \boldsymbol{\Sigma}\left(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0\right), \\ &\boldsymbol{p} = \boldsymbol{p}_1 + p_0. \end{aligned} \tag{15}$$

Next, $Q_x(\boldsymbol{x})$ and $Q_z(\boldsymbol{z})$ will be used to estimate the hyperparameter $\boldsymbol{\theta}$.

B. Hyperparameter estimation using EP posterior approximation

Using $Q_x(\boldsymbol{x}|\boldsymbol{\theta}^{(t-1)})$, $Q_z(\boldsymbol{z}|\boldsymbol{\theta}^{(t-1)})$ computed given $\boldsymbol{\theta}^{(t-1)}$ which is estimated at the $(t-1)$-*th* EM iteration, in the M-step at the (t)-*th* EM iteration, $\boldsymbol{\theta}^{(t)}$ can be estimated by maximizing the marginal likelihood

$$\begin{aligned} \boldsymbol{\theta}^{(t)} &= \operatorname*{argmax}_{\boldsymbol{\theta}} \mathbb{E}_{Q_x(\boldsymbol{x}|\boldsymbol{\theta}^{(t-1)})}\left[\log f(\boldsymbol{y}|\boldsymbol{\theta})\right], \\ \boldsymbol{\theta}^{(t)} &= \operatorname*{argmax}_{\boldsymbol{\theta}} \mathbb{E}_{Q_x(\boldsymbol{x}|\boldsymbol{\theta}^{(t-1)})Q_z(\boldsymbol{z}|\boldsymbol{\theta}^{(t-1)})}\left[\log f(\boldsymbol{y}|\boldsymbol{\theta})\right]. \end{aligned}$$

For $\boldsymbol{\theta}$ in (3), it has been investigated in [15] that the expectation w.r.t. $\hat{p}(\boldsymbol{x}|\boldsymbol{\theta}^{(t-1)}) := f_x(\boldsymbol{x}|\boldsymbol{\theta})q_{x,1}(\boldsymbol{x})$ performs better for hyperparameter estimation and in turn for the estimate of $\boldsymbol{x}$ than that of $Q_x(\boldsymbol{x}|\boldsymbol{\theta}^{(t-1)})$. Closed-form solutions can be derived to estimate $\boldsymbol{\theta}$ in (3) and (7), even if $\boldsymbol{\theta}$ has more parameters, e.g. if only subsets of $\boldsymbol{x}$ share the same hyperparameters. If the hyperparameters are not widely shared, they are often assigned hyperpriors, i.e. the prior distributions of hyperparamteres, and the updates above need to be adjusted. This is not performed here due to space constraints. The two steps in III-A and III-B are iterated until convergence.

IV. Experiments

This section evaluates the performance of the proposed unsupervised EP method on synthetic experiments. The synthetic datasets are 1D vectors and 2D images with known true values. The proposed method is applied to estimate the sparse vectors from the simulated observations to illustrate its efficiency in solving large-scale sparse linear inverse problems.

A. Synthetic experiments on 1D vectors

In this subsection, three i.i.d. sparse vectors $\boldsymbol{x}$ are generated according to the three sparsity-promoting priors with known true values of hyperparameter $\boldsymbol{\theta}^*$ in Section II, i.e. $\boldsymbol{x} \sim f_x(\boldsymbol{x}|\lambda^*)$ in (3) for Laplace prior, $\boldsymbol{x}, \boldsymbol{z} \sim f(\boldsymbol{x}, \boldsymbol{z}|\pi_0^*, v_1^*, v_2^*)$ in (7) for MoG2 and SaS priors (v_2^* is not needed in SaS prior). The observations $\boldsymbol{y}$ are simulated via $\boldsymbol{y} \sim \mathcal{N}(\mathbf{A}\boldsymbol{x}, \sigma^2\mathbf{I})$, where $\mathbf{A}$ is a $M \times N$ (N=100, M=2N, M=N, M=$N/2$) random matrix with i.i.d. standard Gaussian entries and $\sigma^2 = 0.01$. The proposed EP method is applied to estimate $\boldsymbol{x}$, $(\boldsymbol{x}, \boldsymbol{z})$ from $\boldsymbol{y}$. The estimated mean values of $\boldsymbol{x}$, $\boldsymbol{z}$ are obtained by the approximate posterior means of $Q_x(\boldsymbol{x})$ and $Q_z(\boldsymbol{z})$.

Table I reports the mean values with standard deviation of the estimated hyperparameter $\hat{\boldsymbol{\theta}}$ and root-mean-square error (RMSE) of $\boldsymbol{\mu}$ over 200 noise realizations. Note that the observations $\boldsymbol{y}$ are different for each row in the table as $\boldsymbol{x}$ in $\boldsymbol{y} \sim \mathcal{N}(\mathbf{A}\boldsymbol{x}, \sigma^2\mathbf{I})$ are different. It can be seen that the estimated hyperparameters of the three priors are close to the true values, and the RMSE values are relatively low. When the available information in observation $\boldsymbol{y}$ becomes less as M decreases, RMSE increases and the estimated hyperparameters become less accurate but still in good agreement with the true values. Figure 1 plots the estimation results from the second row of Table I. For MoG2 and SaS priors, in addition to the estimates of $\boldsymbol{x}$, the proposed EP method also provides the estimates of the sparsity indicating variable $\boldsymbol{z}$.

B. Synthetic experiments on 2D image deconvolution

In this subsection, synthetic experiments are conducted on 2D image deconvolution in the wavelet domain. A noise-free *Cameraman* image of size N=128×128 pixels is used as the true image $\boldsymbol{x}$. The true sparse vector is the coefficient of $\boldsymbol{x}$ transformed in Haar wavelet domain over 4 scales (including the coarse scale). The observed image $\boldsymbol{y}$ is generated by blurring $\boldsymbol{x}$ using a 5×5 pixels uniform kernel. The blurred signal-to-noise ratio (BSNR) is 20dB. The proposed unsupervised EP method is applied to estimate the deconvolved image from $\boldsymbol{y}$ in the Haar wavelet domain, as presented in Figure 2. The estimated images are obtained by transforming the estimated wavelet coefficients back to the image domain. RMSE and structural similarity (SSIM) are computed between the estimated images and true images. The wavelet coefficients and their uncertainties are obtained from the EP approximate posterior mean $\boldsymbol{\mu}$ and marginal variances diag($\boldsymbol{\Sigma}$), For MoG2 and SaS priors, the sparsity maps obtained by the mean of $Q_z(\boldsymbol{z})$ are shown to indicate the probability of the wavelet coefficients being significantly different from zero at different scales. Furthermore, the proportion of sparsity coefficients over different scales by the MoG and SaS priors are close to the true values. Observe that using the three sparsity-promoting priors, the proposed EP method manages to recover the texture of wavelet coefficients, where the MoG2 and SaS priors perform better than the Laplace prior. Moreover, the approximate marginal variances diag($\boldsymbol{\Sigma}$) by the EP method directly quantify the uncertainty of the estimated wavelet coefficients.

	Laplace prior	MoG2 prior	SaS prior
true values	λ^* = 0.7	π_0^* = 0.1 v_1^* = 30 v_2^* = 0.1	π_0^* = 0.1 v_1^* = 30
M=2N	$\hat{\lambda}$ = 0.6 $\pm 9.2 \times 10^{-5}$ RMSE($\boldsymbol{\mu}$) = 0.001 $\pm 1.1 \times 10^{-4}$	$\hat{\pi}_0$ = 0.1 $\pm 1.9 \times 10^{-5}$ $\hat{v}_1$ = 31 $\pm 6 \times 10^{-3}$ $\hat{v}_2$ = 0.1 $\pm 6.6 \times 10^{-5}$ RMSE($\boldsymbol{\mu}$) = 0.001 $\pm 1 \times 10^{-4}$	$\hat{\pi}_0$ = 0.1 $\pm 1.6 \times 10^{-4}$ $\hat{v}_1$ = 31 $\pm 7 \times 10^{-2}$ RMSE($\boldsymbol{\mu}$) = 0.0002 $\pm 5 \times 10^{-5}$
M=N	$\hat{\lambda}$ = 0.78 $\pm 4.4 \times 10^{-3}$ RMSE($\boldsymbol{\mu}$) = 0.04 $\pm 2.6 \times 10^{-2}$	$\hat{\pi}_0$ = 0.1 $\pm 1.7 \times 10^{-4}$ $\hat{v}_1$= 34 $\pm 7.3 \times 10^{-2}$ $\hat{v}_2$ = 0.1 $\pm 9.2 \times 10^{-4}$ RMSE($\boldsymbol{\mu}$) = 0.0231 $\pm 1.1 \times 10^{-2}$	$\hat{\pi}_0$ = 0.1 $\pm 3.3 \times 10^{-5}$ $\hat{v}_1$ = 28 $\pm 1.6 \times 10^{-2}$ RMSE($\boldsymbol{\mu}$) = 0.0002 $\pm 6.8 \times 10^{-5}$
M=N/2	$\hat{\lambda}$ = 0.74 $\pm 1.3 \times 10^{-4}$ RMSE($\boldsymbol{\mu}$) = 0.54 $\pm 8 \times 10^{-5}$	$\hat{\pi}_0$ = 0.1 $\pm 6.6 \times 10^{-5}$ $\hat{v}_1$ = 26 $\pm 1.7 \times 10^{-2}$ $\hat{v}_2$ = 0.18 $\pm 3.1 \times 10^{-4}$ RMSE($\boldsymbol{\mu}$) = 0.3509 $\pm 2.2 \times 10^{-4}$	$\hat{\pi}_0$ = 0.2 $\pm 2.15 \times 10^{-4}$ $\hat{v}_1$ = 21 $\pm 3 \times 10^{-2}$ RMSE($\boldsymbol{\mu}$) = 0.0006 $\pm 1.4 \times 10^{-4}$

TABLE I

MEAN VALUE ± STANDARD DEVIATION OF THE ESTIMATED HYPERPARAMETER $\hat{\boldsymbol{\theta}}$ AND RMSE OF $\boldsymbol{\mu}$ OVER 200 NOISE REALIZATIONS.

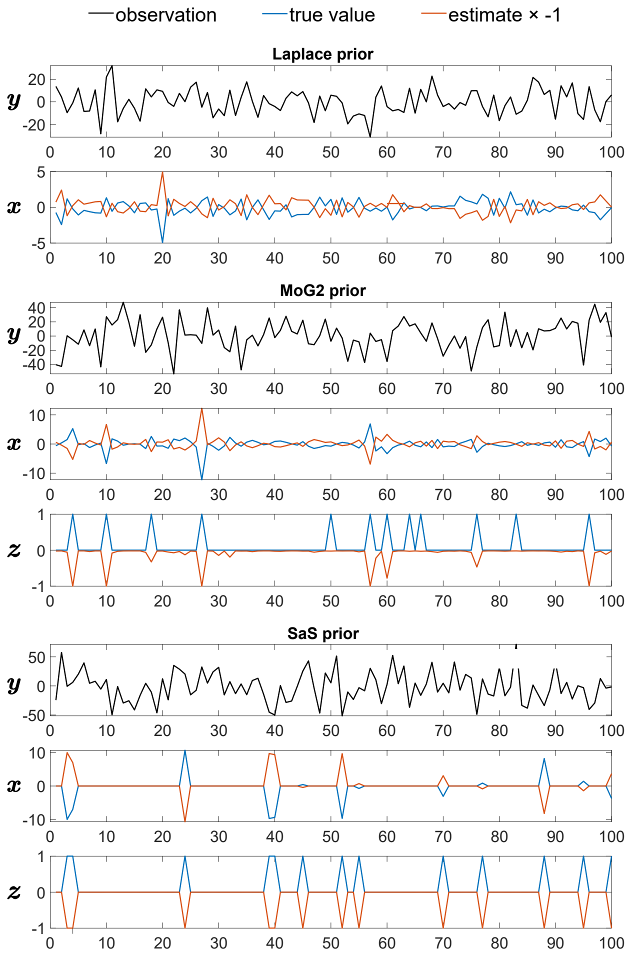


Fig. 1. Estimation results of $\boldsymbol{x}$, $\boldsymbol{z}$ when $M = N$. The estimates in orange are shown by multiplying '-1' for better visualization.

V. CONCLUSION AND FUTURE WORK

In this paper, a new unsupervised EP method is proposed for large-scale sparse linear inverse problems. A Bayesian model is constructed by a Gaussian likelihood and different sparsity-promoting priors, including the Laplace, MoG2, and

Back to Contents

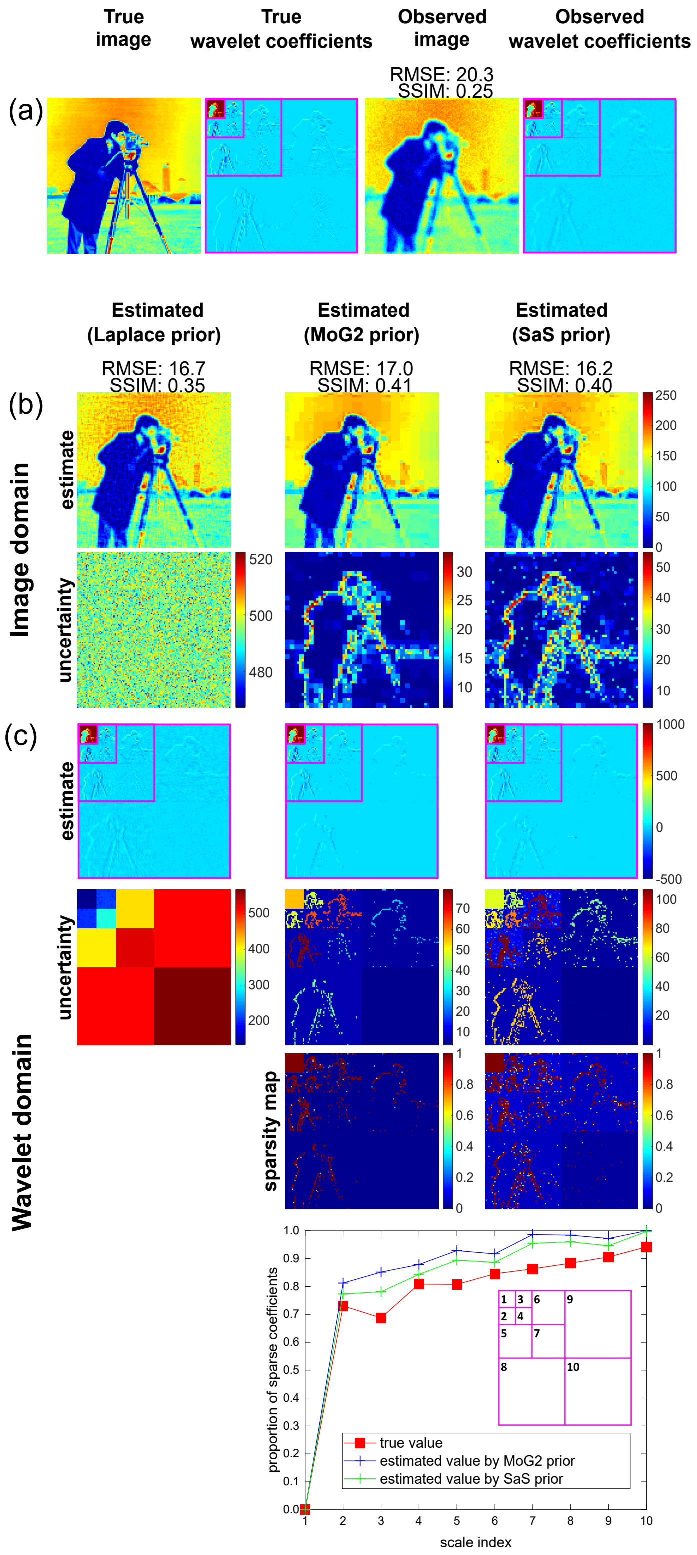


Fig. 2. Results of 2D image deconvolution in Haar wavelet domain. The wavelet coefficients are decomposed over 4 scales (include the coarse scale) and the scales boundaries are shown using the manually added boxes in pink. The scale indices for the proportion of sparse coefficients are listed in the pink boxes.

SaS priors. Approximate Bayesian inference is performed on the posterior approximation found by EP. The MMSE estimate and posterior covariance of the large-scale sparse vector are approximated by the mean vector and covariance matrix of the multivariate approximating distributions. Furthermore, the EP posterior approximation for the unknown model parameters is embedded in a variational EM approach for hyperparameter estimation. Experiments conducted on synthetic datasets illustrate that the proposed EP method can provide not only the approximate MMSE estimates that are close to the true values, but also the uncertainty quantification of the estimates. In particular, MoG2 and SaS priors exhibit the advantages over Laplace prior in providing additional sparsity indicating information. Future work considers replacing the likelihood function of Gaussian i.i.d. noise by other noise models and building more informative sparsity-promoting priors to improve the performance of the proposed method for other large-scale sparse linear inverse problems, such as Poisson noise model and structured prior sparsity in wavelet-based Bayesian compressive sensing.

References

[1] Michael Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*, vol. 2, Springer, 2010.

[2] Roger Ghanem, David Higdon, Houman Owhadi, et al., *Handbook of uncertainty quantification*, vol. 6, Springer, 2017.

[3] Yakov Ben-Haim, "Dealing with uncertainty in strategic decision-making," *The US Army War College Quarterly: Parameters*, vol. 45, no. 3, pp. 8, 2015.

[4] Taewon Cho, Hodjat Pendar, and Julianne Chung, "Computational tools for inversion and uncertainty estimation in respirometry," *Plos one*, vol. 16, no. 5, pp. e0251926, 2021.

[5] Thomas P. Minka, "Expectation Propagation for approximate Bayesian inference," in *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, 2001, UAI'01, p. 362–369, Morgan Kaufmann Publishers Inc.

[6] Sami Bourouis and Nizar Bouguila, "Unsupervised learning using Expectation Propagation inference of inverted Beta-Liouville mixture models for pattern recognition applications," *Cybernetics and Systems*, pp. 1–25, 2022.

[7] Aki Vehtari, Andrew Gelman, Tuomas Sivula, Pasi Jylänki, Dustin Tran, Swupnil Sahai, Paul Blomstedt, John P Cunningham, David Schiminovich, and Christian P Robert, "Expectation Propagation as a way of life: A framework for Bayesian inference on partitioned data.," *J. Mach. Learn. Res.*, vol. 21, no. 17, pp. 1–53, 2020.

[8] Matthias Seeger, Florian Steinke, and Koji Tsuda, "Bayesian inference and optimal design in the sparse linear model," in *Artificial Intelligence and Statistics*. PMLR, 2007, pp. 444–451.

[9] José Miguel Hernández-Lobato, Daniel Hernández-Lobato, and Alberto Suárez, "Expectation Propagation in linear regression models with spike-and-slab priors," *Machine Learning*, vol. 99, no. 3, pp. 437–487, 2015.

[10] Altmann Yoann, "On approximate Bayesian methods for large-scale sparse linear inverse problems," *2022 30th European Signal Processing Conference (EUSIPCO), to appear*.

[11] Ingrid Daubechies, Michel Defrise, and Christine De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 57, no. 11, pp. 1413–1457, 2004.

[12] Bruno Olshausen and K Millman, "Learning sparse codes with a mixture-of-Gaussians prior," *Advances in neural information processing systems*, vol. 12, 1999.

[13] Edward I George and Robert E McCulloch, "Approaches for Bayesian variable selection," *Statistica Sinica*, pp. 339–373, 1997.

[14] Giorgio Parisi, *Statistical field theory*, Addison-Wesley, 1988.

[15] Dan Yao, Stephen Mclaughlin, and Yoann Altmann, "Fast scalable image restoration using total variation priors and Expectation Propagation," *ArXiv*, vol. abs/2110.01585, 2021.

Movement Classification and Segmentation Using Event-Based Sensing and Spiking Neural Networks

Paul Kirkland
Neuromorphic Sensor Signal Processing Lab
University of Strathclyde
Glasgow, Scotland
paul.kirkland@strath.ac.uk

Gaetano Di Caterina
Neuromorphic Sensor Signal Processing Lab
University of Strathclyde
Glasgow, Scotland
gaetano.di-caterina@strath.ac.uk

***Abstract*—The development of Spiking Neural Networks (SNN) and the discipline of Neuromorphic Engineering has resulted in a paradigm shift in how Machine Learning (ML) and Computer Vision (CV) problems are approached. At the heart of this shift is the adoption of event-based sensing and processing methods. The production of sparse and asynchronous events that are dynamically connected to the scene is possible with an event-based vision sensor, allowing for the acquisition of not just spatial data but also high-fidelity temporal data. In this work, we describe a novel method for performing instance segmentation of objects, only using their spatio-temporal movement patterns, by utilising the weights of an unsupervised Spiking Convolutional Neural Network that was originally trained for object recognition and extending it to instance segmentation. This takes advantage of the network's spatial and temporal characteristics encoded within its internal feature representation, to offer this additional discriminative ability. We demonstrate this through a track path identification problem, where 6 identical blobs complete complex movement patterns within the same area at the same time. The network is able to successfully identify all 6 individual movements and segment the movement patterns belonging to each. The work then also explains how these methods map into the more complex Track before Detect problem. A complex track initiation problem, where detection can only be completed after an integration period, due to the low signal, high noise environment. These problem characteristics seem to complement the properties of event-based sensing and processing and initial test results are shown.**

***Index Terms*—Neuromorphic Engineering, Neuromorphic Algorithms, SNN, STDP, Computer Vision, Unsupervised Learning, Instance Segmentation, Event-Based Vision**

I. Introduction

In most defence applications, identification of any target is a time-sensitive and crucial function. However, it is not only detection and identification that is vital, as the exact location is also an important consideration. With the recent take over of deep learning (DL) in the computer vision domain, much research and effort have gone into turning the state of the art in object detection [1], [2], into the instance recognition of video information [3]. However, the reality of the situation in a defence scenario is that the target object is often extremely small (one or few pixels), and it contains no relevant spatial information to discriminate it from background noise and clutter. This then rules out the idea of performing frame-based detection. In cases like this, the requirement for a recurrent approach to allow the accumulation of information over time is required [4]. However, the drawback to this solution is that the longer the integration period, the higher the computational overhead required. Once this issue gets into the low signal or low signal high noise realm, where methods such as Track before Detect (TBD) are used, then DL approaches appear to have had a minimal impact [5].

Neuromorphic Engineering introduces a new paradigm to the sensing and processing domain with the use of event-driven asynchronous sparse binary information. Taking inspiration from biological systems, Neuromorphic sensor signal processing aims to take methods from the breadth of the machine learning community, including DL, and to combine them with the new event-based method of sensing data. This way of thinking is driven by innate abilities that exist in nature. For instance, even in the presence of various background and foreground distractors, human vision has the natural ability to recognise, localise, and discriminate items of interest. This is all done in real-time within a minimal power budget, usually while also completing a number of other complex tasks. Neuromorphic simply means brain-like, in that biological inspiration is taken in how to handle information. Specifically, this makes use of event-driven binary spikes, rather than numerical values, to sense and process data. This means the information precision lies in the timing or rate of the spikes rather than in their magnitude. Neuromorphic sensors give a high temporal resolution without the computational burden, while the event-driven nature of the sensing means the processing would naturally accumulate information over time. This results in high fidelity spatio-temporal patterns to be resolved, where detection and localisation are computed simultaneously.

Neuromorphic, or event-based, sensors have matured over recent years, with vision sensors becoming particularly popular. So much so even consumer products are available, as for example the asynchronous time-based image sensor (ATIS [6]), backed by Sony and sold by Prophesee, and the Dynamic Vision Sensor (DVS [7]), backed by Samsung and sold by Inivation. Event-based sensing is done typically

This work is funded by the Defence Science and Technology Laboratory (DSTL) under the DASA Advanced Vision 2020 Project (ACC6010078 - Neuromorphic processing detection and tracking of fast moving targets) contract DSTLX1000147830.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

through change detection, where a large enough change in the signal causes the sensor to output spikes. The level of this change can be set on the sensor to ensure a suitable output. This change detection greatly helps to sparsify the output. The spikes output by the sensor then represent a high resolution and asynchronous temporal record of the changes occurring in the scene. Even though there is a high degree of spatial and temporal resolution, the data is still sparse compared to a traditional frame-based imaging approach, since not every pixel changes at the same time. This means the sensor has a dynamical relationship to the scene. To exploit this feature, we pair the sensing with a processing method that has a variable integration period, thus capturing the movement period precisely and collecting the relevant information.

Neuromorphic processing is typically carried out using the 3rd generation of neural networks, referred to as Spiking Neural Networks (SNN). The SNN exhibits properties such as asynchronous and event-driven processing, fast inference, low power consumption, massive parallelism and online learning. All of which makes it an interesting prospect in many applications, and ideal for processing information that requires integration over time. In this sense, it means the SNN benefits from not requiring recurrency to extract sequential or temporal information, as such networks are naturally time-dependent. Another benefit of the SNN is that it can exploit being related to Artificial Neural Networks (ANN), as methods of feature extraction can be ported from one to the other. One such method, the Convolutional Neural Networks (CNN), is an efficient and effective method for both learning and extracting features, due to the natural local continuity of objects in both space and time.

II. Algorithmic Development

The main theme to the algorithmic work is to exploit the SNN in the application of instance segmentation. The detection and pixel-wise delineation of each separate item of interest present in a picture is known as instance segmentation. In essence, instance segmentation is a mixture between object recognition and semantic segmentation, two important computer vision problems. Detecting instances of things belonging to a specific class, while simultaneously determining their physical position, usually using a bounding box, is known as object detection. Semantic segmentation, on the other hand, is the challenge of grouping areas of an image that belong to the same object class together, resulting in a considerably more thorough pixel-wise localisation. This problem only gets more complex when attempted on video instead of images, as now the processing time must be less than the time interval available until the next frame, otherwise extra latency is added to the system. For fast-moving objects or scenes, this only becomes more difficult as the rate at which one senses must increase, i.e. higher frame rate, thus forcing a shorter processing interval available.

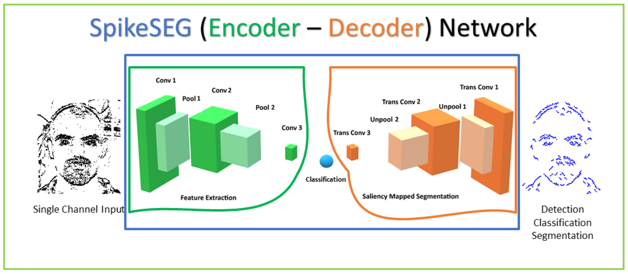


Fig. 1. The SpikeSEG network used to segment spiking images. The encoder is featured in green and the decoder is featured in orange.

A. Spatial Scene Understanding

To initially approach the extraction of useful spatial features from a spiking event-based scene, this work borrows from the previous own SpikeSEG [8], which details how a convolutional encoder-decoder network can be utilised to extract commonly occurring spatial features in a scene (within the encoder). Then it maps this semantically contextualised information into the pixel space again (through the decoder). This in essence allows semantic segmentation to be performed on spiking event data within an unsupervised regime. An example of the network along with an input/output is shown in Fig. 1. The network architecture illustrated here is made up of two main sections seen in green and orange, that relate to the encoding and decoding layers respectively. The network is split into these two sections where training only occurs on the encoding side, while the weights are tied to the mirrored decoding layers. This allows an integrate and fire neuron with layer-wise STDP mechanism, and with adaptive thresholding and pruning, to be used to help represent spatial features of the input. These features are then learned through the encoder, which in turn allows the decoder to segment the image based on the *Conv3 / Trans Conv3* pseudo classification layers.

This encoding-decoding structure symbolises a feature extraction and then a shape generation process. The learning of the encoding process aims to extract common spatial structures as useful features, then it decodes those learned features over to the shape generation process, unravelling the latent space classification representation, although with a reduction in spiking activity due to the max-pooling process. The network has 9 computational layers *(Conv1-Pool1-Conv2-Pool2-Conv3-TransConv3-UnPool2-TransConv2-UnPool1-TransConv1)* as seen in Fig. 1. Between the *Conv3* and *TransConv3* layers, there is a user-defined attention inhibition mechanism / classification, which can operate in two manners: ‘No Inhibition’, which allows semantic segmentation of all recognised classes from the pseudo classification layer; or ‘With Inhibition’, which only allows one class to propagate forward to the decoding layers. This attention not only provides a reduction in the amount of computation, but also simplifies the output of the network, for simpler handover to downstream systems. For further information contained within this section regarding the process of encoding, decoding, thresholding and pruning, see [9]. Fig. 2 helps visualise the internal working of the network. This illustration details the internal network dynamics, with each coloured pixel representing the corresponding region's feature map activation. Classification is the joint representation of *Conv3* and *TransConv3*, which in this case would be the same, as only one class is present.

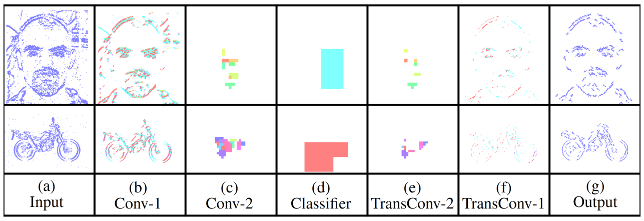


Fig. 2. The internal network representation of SpikeSEG for two class examples.

B. Featural-Temporal Decomposition

Building upon the successful feature extraction of the SpikeSEG network, it was noted that items within particular classes seemed to exhibit rather unique temporal patterns in which the features (neurons) inside the network would be active. This can be simply explained due to the STDP process of learning by looking for the most salient and occurring features. Therefore the more salient the feature, the larger likelihood it would be activated earlier in time. From this hypothesis, the Hierarchical Unravelling of Linked Kernels (HULK) and Similarity Matching through Active Spike Hashing (SMASH) algorithms were designed.

HULK is the process of taking each spiking instance from the last layer of the encoder and unravelling its path through the decoder, no longer at a semantic level, but at the instance level. So for each spike in that feature map, one can track it back to the pixel space, rather than doing it from all the spikes in any given feature map, as was shown previously in Fig. 2. Instead, there is a more granular process now as shown within Fig. 3, which depicts a flow chart of the HULK SMASH process, along with examples from sections of the process [9]. The image highlights the process starting with the SpikeSEG network, but looks at each spike within the last convolution layer leading to the HULK ASH image. Another representation for this featural-temporal representation is shown just below with the red and blue spike trains, which highlight the differences more clearly. The final example image depicts the SMASH process, where the similarity and proximity scores are combined to decide on the number of instances present in the image. Further details regarding specific parts of the process can be found in the [9].

Overall the HULK SMASH process was able to show that not only is the spatial feature information useful in identifying objects within the image, but also that the temporal sequencing in which the features occur can be utilised for more specification identifications. This finding underpins the importance of the temporal nature of the spiking event data: it is the ability to encode the saliency of features, simply by allowing them to occur earlier than less salient features.

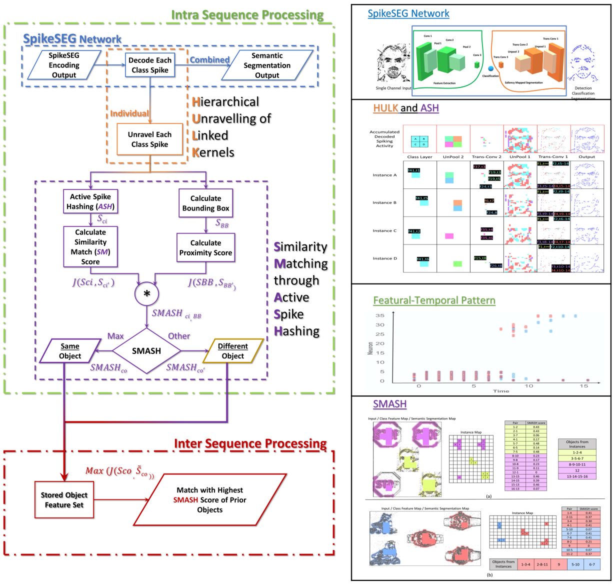

Intra Sequence Processing
SpikeSEG Network
SpikeSEG Encoding Output
Decode Each Class Spike
Combined
Semantic Segmentation Output
Individual
Unravel Each Class Spike
Hierarchical Unravelling of Linked Kernels
Active Spike Hashing (ASH)
S_{ci}
Calculate Similarity Match (SM) Score
Calculate Bounding Box
S_{BB}
Calculate Proximity Score
$J(Sci, S_{ci'})$
$J(SBB, S_{BB'})$
*
$SMASH_{ci, BB}$
Max
Other
SMASH
Same Object
Different Object
$SMASH_{co}$
$SMASH_{co'}$
Similarity Matching through Active Spike Hashing
Inter Sequence Processing
$Max\ (J(Sco, \tilde{S}_{co}))$
Stored Object Feature Set
Match with Highest SMASH Score of Prior Objects
SpikeSEG Network
HULK and ASH
Featural-Temporal Pattern
Neuron
Time
SMASH

Fig. 3. Flow chart for HULK SMASH with examples for each section.

C. Spatio-Temporal-Featural Decomposition

Once it was established that featural-temporal information could be extracted from the spatial features of the spiking event data, the next step was to test the feature extraction ability on spatio-temporal information. As such, the spatial information alone is not representative of anything meaningful, so a longer integration period is required to ascertain if there is a temporal component to the spatial information presented. This was tested under the assumption of an unknown object (small dot) completing a set number of movement patterns, as seen in Fig. 4. It would then be the movement pattern that would be the identifying feature of the data. The SpikeSEG network allows a temporally invariant classification of known movement patterns to be determined, while the HULK process re-enables the temporal variance to further determine the temporal aspect of the feature occurrence. In essence, it allows the system to further resolve if the movement was completed fast or slow. An example of the feature breakdown is shown per layer in the encoder and decoder in Fig. 5. This is a time integrated view of the accumulation of features showing the mapping from pixel to classification latent space and back to pixel domain. The HULK and ASH process ensures the temporal continuity is also captured to be used for further comparing and contrasting of event sequences.

III. Use Case Demonstration

The demonstrator envisages a particularly challenging tracking example, where the three previously mentioned movement

Back to Contents

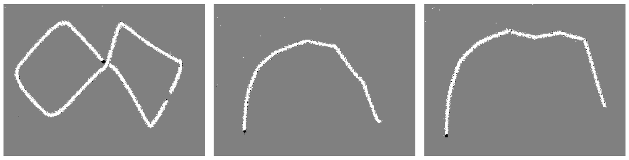


Fig. 4. Movement patterns.

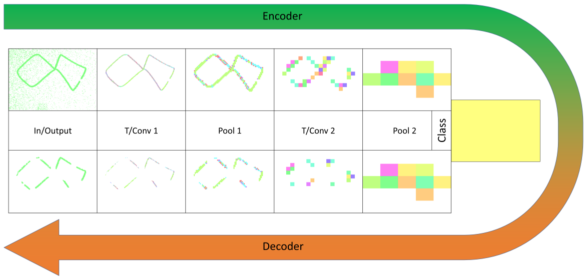


Fig. 5. HULK breakdown of spatio-temporal features.

patterns are occurring in close proximity to one another and at about the same time. This happens along with the mirrored (over the y-axis) version of the events. The resultant integrated over time image for this scene is shown in top left of Fig. 6. The difficulty in this task is that spatially the target object that is moving around it the same in all examples. It also is occluded and crosses paths of the other targets, together with some unpredictable movements (i.e. figure of eight). This demo is supposed to rule out the possibility of just simple identifying each of the moving targets as individuals, instead meaning one relies on the movement of the target, to be able to classify it.

Testing of this complex scenario highlights the strength of the SpikeSEG and HULK SMASH methods. A breakdown of the integrated feature extraction process for the whole multi-target movement scene is illustrated in Fig. 6, where there is a high degree of spatial overlap from the scene which is represented within all the feature extraction layers. However, due to the high temporal resolution of the event data from the scene, the spatio-temporal overlap of the target is rather minimal. This results in only minimal overlap of features allowing the movement patterns to be resolved, as shown in Fig. 6.

The accumulated result of this is that the 6 movement patterns can be distinguish between as seen in Fig. 6, where although there was a large overlap in the spatial location of the movements, each movement path could be classified and segmented.

The image appears fragmented as the pooling layers are still active on the decoding side, meaning only the most relevant information passes through to the pixel space again. This was to ensure the output of the network was more specific than it was sensitive. The high degree of spatial overlap means that certain regions were not the most salient in terms of the classification process and therefore are not shown in the segmentation. The segmentation is quite literally a saliency mapping of the found features. However, now the output of the network is an instance and semantically contextualised version of the input. Meaning, that if only wanting to look for a figure of eight movements, one could inhibit all the other classes, and the output from the demonstrator would only show the two figure of eight movements. A number of spatial and temporal variations of this demo were tested (i.e. X,Y displacement, time displacement, temporal continuity). The SpikeSEG network was able to semantically classify each movement successfully, while the HULK SMASH algorithm was able to determine instances within the classes. As such, it was possible to notice changes in the temporal structure of the spiking event data (i.e. the scene was faster/slower than the previous, and if the features occurred in a different order). This means the system is invariant to movement and the location of the movement within the scene does not matter, while being temporally variant, as the timing of the occurrence of the features does matter. This clarifies that the SpikeSEG network is invariant to both space and time, while the HULK SMASH algorithm adds the variance to the feature data. This is only permitted due to the SpikeSEG network being an asynchronous processing Spiking Convolutions Neural Network, which maintains the temporal continuity of the incoming data due to the neurons firing, even though the network itself is invariant to time.

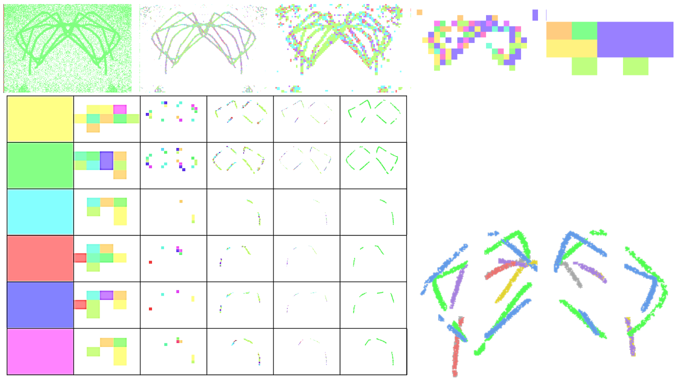


Fig. 6. Breakdown of the features used to segment one movement from the multi-movement scene, with final coloured segmentation

A. Track before Detect Problem

This section covers the initial testing that has been carried out using the same network as described above, but in the situation of a low signal to noise ratio (SNR). This problem is highly related to the principle behind Track before Detect (TBD), as detection is based on tracking or accumulating information on any objects of interest within a scene. However, the time scales required for movement detection are far shorter than that required in the previous classification task. Regardless, it became clear that neuromorphic sensing and processing could be utilised to great effect in the more challenging TBD domain. The neuromorphic event-

Back to Contents

based sensor allows for the accumulation of spatio-temporal information on higher fidelity and variable/incoherent scale, due to its high temporal resolution and asynchronous readout. This means the sensor can accumulate small enough amounts of time to detect pixel motion, while mitigating the effects of the sensor noise and clutter. Fig. 7 illustrates how a longer integration time has lower levels of SNR (left), compared to a less noisy shorter integration time (right). It is in this non-linear relationship between the signal and noise where the benefits of an asynchronous approach are most seen, which is somewhat similar to the benefits of incoherence in randomly sampling for PF. This asynchronous sensing also allows a dynamical relationship to movement in the scene, meaning those moments when movement is detected can be extracted as needed, exploiting the ability to collect high SNR values over this short period. This is in contrast to the fixed temporal rate in a traditional frame-based imaging sensor, which will just accumulate over a set period irrespective of signal movement.

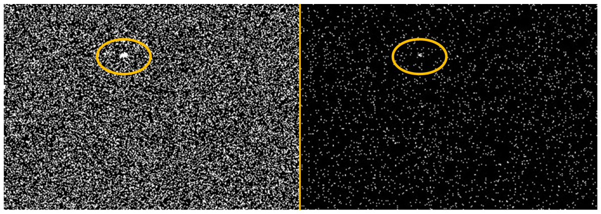


Fig. 7. High and low noise due to integration time.

Preliminary testing of our previously designed systems on an example of a TBD problem has resulted in very encouraging results, with a similar laser pointer example as shown earlier, creating a non-distinct moving blob, but with a high level of noise present due to the closing of the aperture of the sensor. This results in a very low SNR value of around -21dB for the movement sequence (based on signal strength captured in a relatively noise-free environment compared to a signal-free noise environment). This scenario was initially tested against simple implementations of a Kalman filter and a particle filter in both the high and low SNR scenarios. All three systems are not optimised for the task, but manage to perform tracking very well on the clean data. However, when tested on the highly noisy data, only the neuromorphic processing can extract the moving point, as illustrated in Fig. 8. The Kalman filter case shows two predicted points, one of which is close to the object, including briefly tracking the point, but then losing it. The particle filter case shows the particles as a red plus and the mean point a yellow star, and none of the particles are aligned with the object. The SNN case shows the output of the system with only the pixels that were first activated in the system (time to first spike), so operating on a single layer spiking convolution process with a matching encoding and decoding layer, then inhibiting all other feature neurons to produce this output. For this to work in a continuous asynchronous manner, the time to first spike method would need to be changed to a rate-based approach based on spatio-temporal correlation neurons firing.

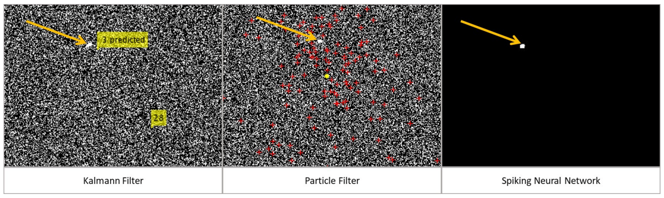


Fig. 8. Output from noisy data for Kalman Filter, Particle Filter and Spiking Neural Network.

IV. Conclusion

In this paper, we have presented how the paradigm of neuromorphic engineering and its event-based sensing and processing can provide an efficient and effective method of extracting complex spatio-temporal patterns from a visual scene, without the requirement for recurrency. This method is then also shown to have promise in TBD, a more relevant defence scenario of low SNR track initiation. Here engineering and its event-based sensing and processing can allow recovering the movement pattern from a highly noisy scene by exploiting the non-linear relationship between the noise distribution and the movement induced signal.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[2] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, pp. 213–229, Springer, 2020.

[3] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9157–9166, 2019.

[4] G. Ning, Z. Zhang, C. Huang, X. Ren, H. Wang, C. Cai, and Z. He, "Spatially supervised recurrent convolutional neural networks for visual object tracking," in *2017 IEEE international symposium on circuits and systems (ISCAS)*, pp. 1–4, IEEE, 2017.

[5] E. Peters and J. Roecker, "Hybrid tracking of low snr targets," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–6, IEEE, 2021.

[6] C. Posch, D. Matolin, and R. Wohlgenannt, "A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS," *IEEE JOURNAL OF SOLID-STATE CIRCUITS*, vol. 46, no. 1, p. 259, 2011.

[7] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 120 dB 15micro s Latency Asynchronous Temporal Contrast Vision Sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[8] P. Kirkland, G. Di Caterina, J. Soraghan, and G. Matich, "Spikeseg: Spiking segmentation via stdp saliency mapping," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2020.

[9] P. Kirkland, G. Di Caterina, J. Soraghan, and G. Matich, "Perception understanding action: adding understanding to the perception action cycle with spiking segmentation," *Frontiers in Neurorobotics*, vol. 14, 2020.

Enhanced Space-Time Covariance Estimation Based on a System Identification Approach

Faizan A. Khattak, Ian K. Proudler, and Stephan Weiss
Department of Electronic & Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, Scotland
{faizan.khattak,ian.proudler,stephan.weiss}@strath.ac.uk

Abstract—**The error inflicted on a space-time covariance estimate due to the availability of only finite data is known to perturb the eigenvalues and eigenspaces of its z-domain equivalent, i.e., the cross-spectral density matrix. In this paper, we show that a significantly more accurate estimate can be obtained if the source signals driving the signal model are also accessible, such that a system identication approach for the source model becomes viable. We demonstrate this improved accuracy in simulations, and discuss its dependencies on the sample size and the signal to noise ratio of the data.**

I. Introduction

For broadband array data in a vector $\mathbf{x}[n] \in \mathbb{C}^M$ with time index $n \in \mathbb{Z}$, signal processing problems are often formulated using second order statistics, such as when aiming to minimise a mean squared error [1], [2]. Since relative delays between signal components are key to addressing the broadband nature of the signals, the space-time covariance matrix $\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n-\tau]\}$, with $\mathcal{E}\{\cdot\}$ the expectation operator, therefore includes a lag parameter $\tau \in \mathbb{Z}$. Solutions to such problems typically rely on a diagonalisation of $\mathbf{R}[\tau]$. Since the standard eigenvalue decomposition (EVD) can only decouple $\mathbf{R}[\tau]$ for one specific value of τ, an EVD to diagonalise $\mathbf{R}[\tau]$ for all τ, or equivalently its z-transform $\boldsymbol{R}(z) = \sum_\tau \mathbf{R}[\tau]z^{-\tau}$ for all $z \in \mathbb{C}$, is required.

The problem of diagonalising a matrix $\boldsymbol{R}(z)$ is well-understood. An eigenvalue decomposition $\boldsymbol{R}(z) = \boldsymbol{Q}(z)\boldsymbol{\Lambda}(z)\boldsymbol{Q}^{\mathrm{P}}(z)$ exists for almost all analytic matrices [3], [4], such that the diagonal matrix $\boldsymbol{\Lambda}(z)$ contains the eigenvalues, and the $\boldsymbol{Q}(z)$ their corresponding, orthonormal eigenvectors, with $\boldsymbol{Q}^{\mathrm{P}}(z) = \boldsymbol{Q}^{\mathrm{H}}(1/z^*)$ involving the parahermitian, Hermitian, and complex coonjugation operators $\{\cdot\}^{\mathrm{P}}$, $\{\cdot\}^{\mathrm{H}}$, and $\{\cdot\}^*$, respectively [5]. This decomposition can be approximated by various algorithms, including the second order sequential best rotation (SBR2) [6]–[8], sequential matrix diagonalisation (SMD) [9]–[11], and a number of discrete Fourier transform (DFT)-based families of algorithms [12]–[20].

A number of application examples have been successfully addressed by the above algorithms, ranging, e.g., from coding [7], [21], beamforming [22], [23], angle of arrival estimation [24]–[26], speech enhancement [27]–[29], optimum precoder and equaliser resign for MIMO communications systems [30]–[33], and subspace scanning for weak transient signals [34]–[36].

In almost all of these applications, the space-time covariance matrix cannot be obtained via expectations but must be estimated from finite data. The estimate $\hat{\mathbf{R}}[\tau]$ will be prone to estimation errors, and the variance of the unbiased estimator based on N snapshot of data $\mathbf{x}[n]$, $n = 0, \cdots, (N-1)$ has been investigated in [37]. This deviation from the ground truth $\mathbf{R}[\tau]$ will in turn result in a perturbation of the eigenvalues and eigenspaces [38]–[40].

The impact of estimation errors is twofold. Firstly, an estimation error causes imprecision e.g. through subspace leakage for the above applications [41]. Secondly, e.g. overestimating the support of the space-time covariance matrix will result in polynomial matrices of higher order than necessary [43], counteracting many efforts to keep computational complexity low via e.g. numerical efficiency [44]–[47] or trimming of polynomials [48]–[50].

Therefore, in this paper we aim to enhance the estimate $\hat{\mathbf{R}}[\tau]$ and thus reduce the perturbation of its eigenvalue decomposition, as well as aid in keeping the polynomial orders of all factors low. This is achieved the source signals are accessible, such that the convolutive mixing system that contributes to $\mathbf{x}[n]$ can be estimated via system identication. This type of estimation for $\hat{\mathbf{R}}[\tau]$ is possible e.g. in loudspeaker-microphone setup such as in [26]–[29]. For this purpose, we review the EVD of a space-time covariance matrix in Sec. II. The source model that defined $\mathbf{R}[\tau]$ is introduced in Sec. III together with the unbiased estimator of [37]. Our proposed alternative system identification approach is outlined in Sec. IV, and compared to the unbiased estimator via simulations in Sec. V. Conclusions are drawn in Sec. VI.

II. Parahermitian Matrix EVD and Perturbation

A. Parahermitian Matrix EVD

The diagonalisation of the space-time covariance matrix $\mathbf{R}[\tau]$ was motivated in Sec. I as a way to solve broadband problems. Since the model of $\boldsymbol{R}[\tau]$ in Sec. III typically contains causal, stable system components, the z-transform $\boldsymbol{R}(z) = \sum_\tau \mathbf{R}[\tau]z^{-\tau}$ is analytic in $z \in \mathbb{C}$. To diagonalise $\mathbf{R}[\tau]$ for every lag value τ, or $\boldsymbol{R}(z)$ for every z, a standard EVD is insufficient. Instead, a parahermitian matrix EVD [3], [4]

$$\boldsymbol{R}(z) = \boldsymbol{Q}(z)\boldsymbol{\Lambda}(z)\boldsymbol{Q}^{\mathrm{P}}(z) \qquad (1)$$

This work was supported in parts by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1 and the MOD University Defence Research Collaboration in Signal Processing. F. Khattak is the recipient of a Commonwealth Scholarship.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

is required, where the diagonal parahermitian matrix $\boldsymbol{\Lambda}(z)$ contains the eigenvalues $\lambda_m(z)$, $m = 1, \ldots, M$. The corresponding eigenvectors form the columns of $\boldsymbol{Q}(z)$, which is paraunitary such that $\boldsymbol{Q}(z)\boldsymbol{Q}^{\mathrm{P}}(z) = \mathbf{I}$. Both $\boldsymbol{\Lambda}(z)$ and $\boldsymbol{Q}(z)$ can be selected to be analytic, such that (1) can be approximated well by Laurent polynomial terms.

Under some circumstances, the ground truth eigenvalues $\lambda_m(z)$, when evaluated on the unit circle, may satisfy spectral majorisation, such that

$$\lambda_1(\mathrm{e}^{\mathrm{j}\Omega}) \geq \lambda_2(\mathrm{e}^{\mathrm{j}\Omega}) \geq \ldots \geq \lambda_M(\mathrm{e}^{\mathrm{j}\Omega}) \, . \tag{2}$$

Though, generally (2) is not a given, and the ground truth eigenvalues of (7) may overlap. Note that the factorisations provided by the SBR2 [6], [7] and SMD [9], [11], [49] families of PEVD algorithm generally encourage (or can even be shown to guarantee [42]) spectral majorisation, thus conflicting with the analytic solution; in particular, the approximation of spectrally majorised eigenvalues can converge very slowly, requiring Laurent polynomials of much high order than for the analytic solution.

B. Perturbation of Eigenvalues

To investigate how a discrepancy between the ground truth $\mathbf{R}[\tau]$ and the estimated $\hat{\mathbf{R}}[\tau]$ perturbs the eigenvalues, recall from [40] that when evaluated at a specific normalised angular frequency Ω_0, the error in the eigenvalues is bounded due to the Hoffman-Wielandt theorem [39]

$$\sum_{m=1}^{M} \left(\hat{\lambda}_m(\mathrm{e}^{\mathrm{j}\Omega_0}) - \lambda_m(\mathrm{e}^{\mathrm{j}\Omega_0})\right)^2 \leq \|\boldsymbol{E}(\mathrm{e}^{\mathrm{j}\Omega_0})\|_{\mathrm{F}}^2 \, , \tag{3}$$

where $\boldsymbol{E}(\mathrm{e}^{\mathrm{j}\Omega_0}) = \boldsymbol{R}(\mathrm{e}^{\mathrm{j}\Omega_0}) - \hat{\boldsymbol{R}}(\mathrm{e}^{\mathrm{j}\Omega_0})$, and $\hat{\lambda}_m(\mathrm{e}^{\mathrm{j}\Omega_0})$ are the eigenvalues of $\hat{\boldsymbol{R}}(z)$ evaluated for $z = \mathrm{e}^{\mathrm{j}\Omega_0}$. Thus the bin-wise perturbation of the eigenvalues depends directly on the accuracy of the space-time covariance estimate $\hat{\boldsymbol{R}}(z)$. Dependencies similar to (3) can be shown for the eigenspaces.

In the remainder of this paper we will concentrate on limiting the perturbation in (3) by reducing the error in $\hat{\boldsymbol{R}}(z)$.

III. Signal Model and Space-Time Covariance

A. Source Model

We assume that L zero-mean unit-variance uncorrelated sources $u_\ell[n]$, $\ell = 1, \ldots, L$, contribute to the measurements at M sensors via a matrix $\mathbf{H}[n] \in \mathbb{C}^{M\times L}$ of impulse responses as shown in Fig. 1. This system matrix $\mathbf{H}[n]$ is given as

$$\mathbf{H}[n] = \begin{bmatrix} h_{1,1}[n] & h_{1,2}[n] & \ldots & h_{1,L}[n] \\ h_{2,1}[n] & h_{2,2}[n] & \ldots & h_{2,L}[n] \\ \vdots & & \ddots & \vdots \\ h_{M,1}[n] & h_{M,2}[n] & \ldots & h_{M,L}[n] \end{bmatrix} \, , \tag{4}$$

where an element $h_{m,\ell}[n]$ is the impulse response connecting the ℓth source to the mth sensor. Using $\mathbf{H}[n]$, the contribution of all L sources at the mth sensor is

$$x_m[n] = \sum_{\ell=1}^{L} h_{m,\ell}[n] * u_\ell[n] + v_m[n] \, , \tag{5}$$

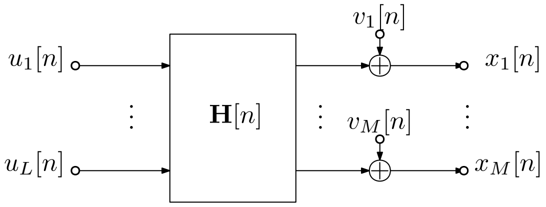


Fig. 1. Source model for the measurement vector $\mathbf{x}[n]$.

where $*$ denotes the convolution operator, and $v_m[n]$ is additive spatially and temporally uncorrelated noise. In matrix notation, for the measurement vector $\mathbf{x}[n] = [x_1[n], \ldots, x_M[n]]^{\mathrm{T}}$ we obtain

$$\mathbf{x}[n] = \mathbf{H}[n] * \mathbf{u}[n] + \mathbf{v}[n] \, , \tag{6}$$

with $\mathbf{u}[n] \in \mathbb{C}^L$ and $\mathbf{v}[n] \in \mathbb{C}^M$ the source signal and noise vectors, respectively, that are defined akin to $\mathbf{x}[n]$. We assume that $\mathbf{H}[n]$ is a finite impulse response system of order L_H.

B. Space-Time Covariance Matrix

With the source covariance $\mathcal{E}\{\mathbf{u}[n]\mathbf{u}^{\mathrm{H}}[n-\tau]\} = \mathbf{I}_L\delta[\tau]$ and the noise covariance $\mathcal{E}\{\mathbf{v}[n]\mathbf{v}^{\mathrm{H}}[n-\tau]\} = \sigma_v^2\mathbf{I}_M\delta[\tau]$, where $\mathcal{E}\{\cdot\}$ is the expectation operator and $\delta[\tau]$ the Kronecker function, the space time covariance $\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^{\mathrm{H}}[n-\tau]\} \in \mathbb{C}^{M\times M}$ can be tied to the source model of Fig. 1 as

$$\mathbf{R}[\tau] = \sum_n \mathbf{H}[n]\mathbf{H}^{\mathrm{H}}[n-\tau] + \sigma_v^2\mathbf{I}_M\delta[\tau] \, . \tag{7}$$

Each element of $\mathbf{R}[\tau]$ is a cross-correlation

$$r_{\ell,m}[\tau] = \mathcal{E}\{x_\ell[n]x_m^*[n-\tau]\} \tag{8}$$

$$= \sum_n \sum_{k=1}^{L} h_{\ell,k}[n]h_{m,k}^*[n-\tau] + \sigma_v^2\delta[\tau]\delta[l-m] \, . \tag{9}$$

C. Unbiased Estimation

In applications, $\mathbf{R}[\tau]$ typically has to be estimated from finite data, leading to an estimated space-time covariance matrix $\hat{\mathbf{R}}[\tau]$. If N measurements $\mathbf{x}[n]$, $n = 0, \ldots, (N-1)$ are available, then an un-biased estimator for (8) can be defined as

$$\hat{r}_{\ell,m}[\tau] = \begin{cases} \frac{1}{N-|\tau|}\sum_{n=0}^{N-|\tau|-1} x_\ell[n+\tau]x_m^*[n], & \tau \geq 0 \;\; , \\ \frac{1}{N-|\tau|}\sum_{n=0}^{N-|\tau|-1} x_\ell[n]x_m^*[n-\tau], & \tau < 0 \, . \end{cases} \tag{10}$$

The variance of the estimator (10) is derived in [37] which forms the average power of the estimation error. It depends on both $\mathbf{R}[\tau]$ and N, and for the variance of one element $\hat{r}_{\ell,m}[\tau]$, we can state [37]

$$\mathrm{var}\{\hat{r}_{\ell,m}[\tau]\} = \frac{1}{(N-|\tau|)^2}\sum_{t=-N+|\tau|+1}^{N-|\tau|-1}(N-|\tau|-|t|) \cdot (r_{\ell,\ell}[t]r_{m,m}^*[t] + \bar{r}_{\ell,m}[\tau+t]\bar{r}_{\ell,m}^*[\tau-t]), \tag{11}$$

Back to Contents

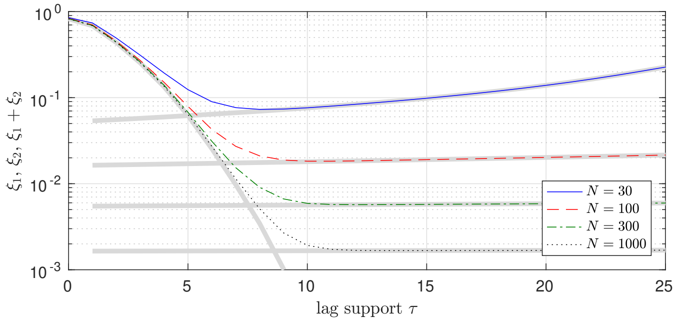


Fig. 2. Overall error ξ when estimating $\mathbf{H}[n]$ from data, in dependence of the number of lags τ, with truncation and estimation error terms ξ_1 and ξ_2, respectively, underlaid in grey for different sample sizes N.

where $\bar{r}_{\ell,m}[\tau] = \mathcal{E}\{x_\ell[n]x_m[n-\tau]\}$ is the complementary cross-correlation sequence. The overall estimation error

$$\xi = \sum_\tau \|\mathcal{E}\Big\{\mathbf{R}[\tau] - \hat{\mathbf{R}}[\tau]\Big\}\|_\text{F}^2 \ , \tag{12}$$

with $\|\cdot\|_\text{F}$ the Frobenius norm, can be minimised by judiciously setting the lag support [43].

Example. For some space-time matrix $\mathbf{R}[\tau] \in \mathbb{C}^{20\times 20}$ of polynomial order 88, Fig. 2 show the truncation error ξ_1 as well as the estimation error ξ_2, which make up the error term $\xi = \xi_1 + \xi_2$ in (12). Note that an increase of the sample size N reduces the estimation error, and increases the optimum lag support, i.e., the value for τ where ξ takes on its minimum in Fig. 2.

IV. Estimation via System Identification

In case we have significantly more access to the system in Fig. 1 and in addition to $\mathbf{x}[n]$ are able to acquire N samples of the source vector $\mathbf{u}[n]$, we can obtain an estimate for $\mathbf{R}[\tau]$ directly via (7), such that

$$\hat{\mathbf{R}}[\tau] = \sum_n \hat{\mathbf{H}}[n]\hat{\mathbf{H}}^\text{H}[n-\tau] + \hat{\sigma}_v^2\mathbf{I}_M\delta[\tau] \ , \tag{13}$$

where $\hat{\mathbf{H}}[n]$ is an estimate of the convolutive mixing system $\mathbf{H}[n]$, which we can obtain via adaptive system identification [2]. The estimate for the noise variance, $\hat{\sigma}_v^2$, can be reached via the minimum mean squared error. We outline these two steps below, followed by some thoughts on how to optimise the lag support in combination with the convolution operation in (13). Because with $\mathbf{u}[n]$, we know significantly more about our system, we also expect (13) to significantly exceed the estimate via (10) based on only $\mathbf{x}[n]$.

A. Adaptive System Identification

Various approaches can be used to perform system identification, including the least mean square and recursive least squares algorithms [2]. In order to operate analogously to the estimation of statistics over N time instances in Sec. III-C, we here select the Wiener solution to identify M separate L-channel adaptive filter problems based on (5),

$$\hat{x}_m[n] = \sum_{\ell=1}^{L} \hat{\mathbf{h}}_{m,\ell}^\text{H}\mathbf{u}_\ell[n] = \hat{\mathbf{w}}_m^\text{H}\mathbf{y}[n] \ . \tag{14}$$

In (14), $\hat{\mathbf{h}}_{m,\ell}^* \in \mathbb{C}^{L_f}$ contains the L_f estimated coefficients of the impulse response $h_{m,\ell}[n]$, and $\mathbf{u}_\ell[n] = [u_\ell[n], \ldots, u_\ell[n-L_f+1]]^\text{T}$ is a tap delay line vector. For compactness of the mean square error problem

$$\hat{\mathbf{w}}_{m,\text{opt}} = \min_{\hat{\mathbf{w}}_m} \mathcal{E}\big\{|\mathbf{x}_m[n] - \hat{\mathbf{x}}_m[n]|^2\big\} \ , \tag{15}$$

we can further define

$$\hat{\mathbf{w}}_m = \begin{bmatrix} \hat{\mathbf{h}}_{m,1} \\ \vdots \\ \hat{\mathbf{h}}_{m,L} \end{bmatrix} \ , \qquad \mathbf{y}[n] = \begin{bmatrix} \mathbf{u}_1[n] \\ \vdots \\ \mathbf{u}_L[n] \end{bmatrix} \ , \tag{16}$$

as utilised in (14). With a sample covariance matrix $\hat{\mathbf{R}}$ and a vector $\hat{\mathbf{p}}_m$ estimating the quantities $\mathcal{E}\big\{\mathbf{y}[n]\mathbf{y}^\text{H}[n]\big\}$ and $\mathcal{E}\{\mathbf{y}[n]x_m[n]\}$ over N time instances, we obtain [1], [2]

$$\hat{\mathbf{w}}_{m,\text{opt}} = \hat{\mathbf{R}}^{-1}\hat{\mathbf{p}}_m \tag{17}$$

as the minimum mean square error estimate of the coefficients in the mth row of $\mathbf{H}[n]$.

B. Minimum Mean Squared Error

In the ideal case where $\hat{\mathbf{w}}_{m,\text{opt}}$ accurately reflects the approriate coefficients of $\mathbf{H}[n]$, the variance estimate $\hat{\sigma^2}_v$ is equivalent to the minimum mean square error,

$$\hat{\sigma}_{v,m}^2 = \hat{\sigma}_{x_m}^2 - \hat{\mathbf{p}}_m^\text{H}\hat{\mathbf{R}}^{-1}\hat{\mathbf{p}}_m \ , \tag{18}$$

where $\hat{\sigma}_{x_m}^2$ is the power estimated over the N samples of $x_m[n]$. Since we need to perform M multichannel adaptive filter calculations, $\hat{\sigma}_v^2$ can be averaged over the M different evaluations of (18), such that $\hat{\sigma}_v^2 = \frac{1}{M}\sum_m \hat{\sigma}_{v,m}^2$.

C. Filter Length and Lag Support

Using the elements of the system matrix $\hat{\mathbf{H}}[n]$ identified via Sec. IV-A and the noise variance as discussed in Sec. IV-B, we can estimate $\mathbf{R}[\tau]$ using (13). Similar to the un-biased estimator, two terms contribute to the error ζ_SI defined akin to (12) between $\mathbf{H}[n]$ and $\hat{\mathbf{H}}[n]$: (i) a truncation term in case the adaptive filter length L_f falls short of the ground truth system length L_H; and (ii) a perturbation term that impacts on the coefficients of $\hat{\mathbf{w}}_{m,\text{opt}}$ in (17), which grows with the number of coefficients. Therefore, we expect to find an optimum length $L_{f,\text{opt}}$, where the two error terms are in balance.

Example. We perform an experiment with an ensemble consisting of 300 instances of a parahermitian matrix $\mathbf{R}[\tau] \in \mathbb{C}^{2\times 2}$ with $L_H = 30$. For the noise variance σ_v^2, we define an average SNR at the sensors,

$$\text{SNR} = \frac{\sum_n \|\mathbf{H}[n]\|_\text{F}^2}{M\sigma_v^2} \ , \tag{19}$$

where the numerator reflects the total power due to the sources and the denominator the total power due to the additive noise

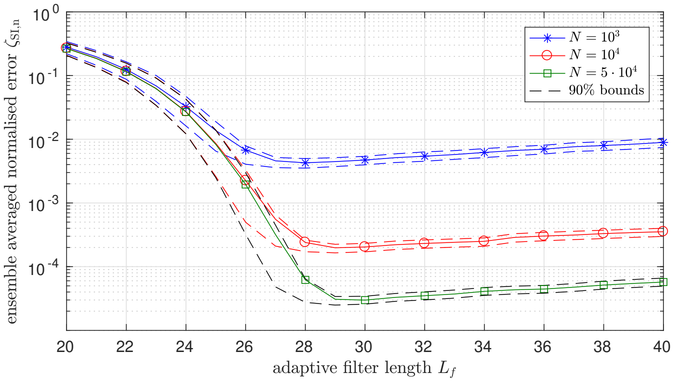


Fig. 3. Ensemble results for ζ when obtaining $\hat{\mathbf{R}}[\tau]$ in dependence of adaptive filter length, L_f.

at the sensors measuring $\mathbf{x}[n]$. In the experiments, we set σ_v^2 to provide an SNR of 20 dB. The ensemble instances are identified with L_f varying from 20 to 40 using various sequence lengths N. The results are illustrated in terms of normalised errors $\zeta_{\mathrm{SI,n}} = \zeta_{\mathrm{SI}} / \sum_n \|\mathbf{H}[n]\|_{\mathrm{F}}^2$ in Fig. 3, which highlights the above trade-off: while for low values of L_f, the truncation error dominates, the error at higher values of L_f increases to the noisy coefficients in the adaptation process.

In addition, the ensemble optimum depends on the filter length. In Fig. 3, note that $L_{\mathrm{f,opt}}$ is 28, 29 and 30 for $N = 1e3$, $1e4$ and $5e4$ respectively. The filter length, for which the minimum is reached, therefore converges towards the ground truth support L_H.

V. Simulations and Comparison

This section provides a comparison of the two approaches to obtain $\hat{\mathbf{R}}[\tau]$ discussed in this paper, and an assessment of the consequences for the perturbation of its eigenvalues.

A. Performance Metric

The performance metric for a comparison of both methods is given as

$$\zeta = \frac{\sum_\tau \|\mathbf{R}[\tau] - \hat{\mathbf{R}}[\tau]\|_{\mathrm{F}}^2}{\sum_\tau \|\mathbf{R}[\tau]\|_{\mathrm{F}}^2} \; . \tag{20}$$

Note that the numerator of this metric relates to the bin-wise perturbation bound on the eigenvalues in (3) via Parseval's theorem [51]. The normalisation by the Frobenius norm of the ground-truth ensures that the metric can be applied to extract ensemble results for different instances of $\mathbf{R}[\tau]$.

B. Scenario and Parameters

To compare both methods, we employ an ensemble of 500 random instances of $\mathbf{R}[\tau] \in \mathbb{C}^{2\times 2}$ with moderately large support $L_H = 30$. The estimates are made over various sample sizes N ranging from 10^3 to 10^6 and noise levels of 10 and 20 dB SNR according to (19). The optimal lag support for the unbiased estimator is selected on the basis of the lowest value of ζ by varying the lag support between 1 and 29 because $\tau_{\mathrm{opt}} < \tau_{\mathrm{gt}} = 30$. In contrast, the support value for SI estimate is set equal to $\tau_{\mathrm{gt}} = 30$.

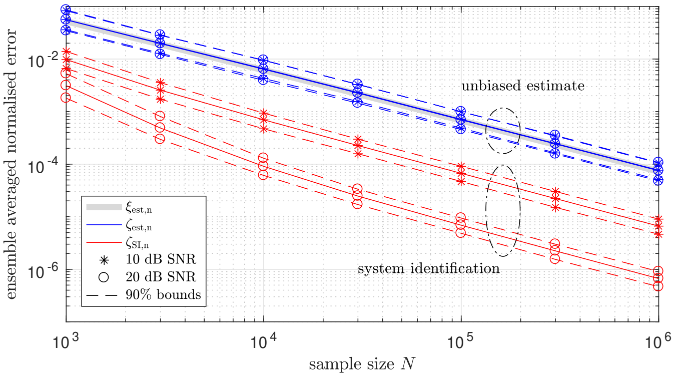


Fig. 4. Comparison of estimation methods via an ensemble of $\mathbf{R}[\tau] \in \mathbb{C}^{2\times 2}$, showing the theoretical and measured error via the unbiased estimator, $\xi_{\mathrm{est,n}}$ and $\zeta_{\mathrm{est,n}}$, respectively, as well as the measured error using the system identification approach, $\zeta_{\mathrm{SI,n}}$.

C. Ensemble Results

Fig. 4 shows the ensemble results for the experiment. The normalised error $\zeta_{\mathrm{est},n}$ is adopted from (20) for the unbiased estimator based on (10); likewise, $\zeta_{\mathrm{SI,n}}$ is the normalised error for the system identification approach. For each case, curves for 10 dB and 20 dB SNR are shown, together with the bounds within which 90% of the ensemble results fall. Further, the theoretical normalised variance for the unbiased estimator, a normalised version of (11), is underlaid in grey, and matches the simulation results well.

We firstly observe that the unbiased estimator, which treats measurement noise as part of the data, is independent of the SNR. In contrast, the noise terms acts as observation noise for the system identication approach, which therefore yields increased accuracy as the SNR grows. All curves converge with approximately $1/N$, but the system identification approach generally is capable of reaching better accuracy than the unbiased estimator. This is due to the additional information that in this case is known for the system — the source signals $u_\ell[n]$. In contrast, for lower SNR, the system identification performance will drop below that of the unbiased estimator, as the known signals $u_\ell[n]$ will be dwarfed by the unknown observation noise $v_m[n]$ which then start to dominate.

VI. Conclusions

We have compared the unbiased estimator with a system identification approach for the estimation of a space time covariance matrix. The latter can be exploited in case the source signals are known, and consists of the identification of the convolutive mixing system by a Wiener filter approach, and the estimation of the additive noise power via the minimum mean square error of the Wiener filter. An ensemble experiment carried out at various noise levels demonstrates that the system identification approach performs significantly better than the unbiased estimator for reasonable to high SNRs. This is important, as the enhanced accuracy results in a lower bin-wise perturbation of the eigenvalue decomposition of this matrix, which is key to formulating and solving a number of relevant broadband array problems.

Back to Contents

References

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, New York: Prentice Hall, 1985.

[2] S. Haykin, *Adaptive Filter Theory*, 2nd ed., Englewood Cliffs: Prentice Hall, 1991.

[3] S. Weiss, J. Pestana, and I. K. Proudler, "On the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix," *IEEE Trans. SP*, **66**(10):2659–2672, May 2018.

[4] S. Weiss, J. Pestana, I. Proudler, and F. Coutts, "Corrections to "on the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix"," *IEEE Trans. SP*, **66**(23):6325–6327, Dec. 2018.

[5] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs: Prentice Hall, 1993.

[6] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD Algorithm for Para-Hermitian Polynomial Matrices," *IEEE Trans. SP*, **55**(5):2158–2169, May 2007.

[7] S. Redif, J. McWhirter, and S. Weiss, "Design of FIR paraunitary filter banks for subband coding using a polynomial eigenvalue decomposition," *IEEE Trans. SP*, **59**(11):5253–5264, Nov. 2011.

[8] S. Redif, S. Weiss, and J.G. McWhirter, "An approximate polynomial matrix eigenvalue decomposition algorithm for para-hermitian matrices," in *11th IEEE ISSPIT*, Bilbao, Spain, pp. 421–425, Dec. 2011,

[9] S. Redif, S. Weiss, and J. McWhirter, "Sequential matrix diagonalization algorithms for polynomial EVD of parahermitian matrices," *IEEE Trans. SP*, **63**(1):81–89, Jan. 2015.

[10] J. Corr, K. Thompson, S. Weiss, J. McWhirter, S. Redif, and I. Proudler, "Multiple shift maximum element sequential matrix diagonalisation for parahermitian matrices," in *IEEE SSP*, Gold Coast, Australia, pp. 312–315 ,June 2014.

[11] V.W. Neo and P.A. Naylor, "Second order sequential best rotation algorithm with householder reduction for polynomial matrix eigenvalue decomposition," in *IEEE ICASSP*, Brighton, UK, pp. 8043–8047, May 2019.

[12] M. Tohidian, H. Amindavar, and A. M. Reza, "A DFT-based approximate eigenvalue and singular value decomposition of polynomial matrices," *EURASIP J. Adv. SP*, **2013**(1):1–16, 2013.

[13] F.K. Coutts, K. Thompson, J. Pestana, I. Proudler, and S. Weiss, "Enforcing eigenvector smoothness for a compact DFT-based polynomial eigenvalue decomposition," in *IEEE SAM*, Sheffield, UK, July 2018.

[14] F.K. Coutts, K. Thompson, I. K. Proudler, and S. Weiss, "An iterative dft-based approach to the polynomial matrix eigenvalue decomposition," in *Asilomar Conf. Signals, Systems, and Computers*, pp. 1011–1015, Pacific Gove, CA, Oct. 2018.

[15] S. Weiss and M. D. Macleod, "Maximally smooth dirichlet interpolation from complete and incomplete sample points on the unit circle," in *IEEE ICASSP*, Brighton, UK, May 2019.

[16] S. Weiss, I.K. Proudler, and M.D. Macleod, "Measuring smoothness of real-valued functions defined by sample points on the unit circle," in *SSPD*, Brighton, UK, May 2019.

[17] S. Weiss, I.K. Proudler, F.K. Coutts, and J. Pestana, "Iterative approximation of analytic eigenvalues of a parahermitian matrix EVD," in *IEEE ICASSP*, Brighton, UK, May 2019.

[18] S. Weiss, J. Selva, and M. Macleod, "Measuring smoothness of trigonometric interpolation through incomplete sample points," in *EUSIPCO*, Amsterdam, Netherlands, pp. 1–5, 2020.

[19] S. Weiss, I.K. Proudler, and F.K. Coutts, "Eigenvalue decomposition of a parahermitian matrix: Extraction of analytic eigenvalues," *IEEE Trans. SP*, 69:722–737, 2021.

[20] S. Weiss, I. Proudler, F. Coutts, and F. Khattak, "Eigenvalue decomposition of a parahermitian matrix: Extraction of analytic eigenvectors," *IEEE Trans. SP*, submitted 2022.

[21] S. Weiss, S. Redif, T. Cooper, C. Liu, P. Baxter, and J. McWhirter, "Paraunitary oversampled filter bank design for channel coding," *EURASIP J. Adv. SP*, **2006**:1–10, 2006.

[22] S. Weiss and I. K. Proudler, "Comparing Efficient Broadband Beamforming Architectures and Their Performance Trade-Offs," in *14th Int. Conf. DSP*, Santorini, Greece, pp. 417–422, July 2002.

[23] S. Weiss, S. Bendoukha, A. Alzin, F. Coutts, I. Proudler, and J. Chambers, "MVDR broadband beamforming using polynomial matrix techniques," in *EUSIPCO*, Nice, France, pp. 839–843, Sep. 2015.

[24] M. Alrmah, S. Weiss, and S. Lambotharan, "An extension of the MUSIC algorithm to broadband scenarios using polynomial eigenvalue decomposition," in *EUSIPCO*, Barcelona, Spain, pp. 629–633, Aug. 2011.

[25] S. Weiss, M. Alrmah, S. Lambotharan, J. McWhirter, and M. Kaveh, "Broadband angle of arrival estimation methods in a polynomial matrix decomposition framework," in *IEEE CAMSAP*, pp. 109–112, Dec. 2013.

[26] A. Hogg, V. Neo, S. Weiss, C. Evers, and P. Naylor, "A polynomial eigenvalue decomposition music approach for broadband sound source localization," in *IEEE WASPAA*, New Paltz, NY, Oct. 2021.

[27] V.W. Neo, C. Evers, and P.A. Naylor, "PEVD-based speech enhancement in reverberant environments," in *ICASSP*, 2020, pp. 186–190.

[28] ——, "Polynomial matrix eigenvalue decomposition of spherical harmonics for speech enhancement," in *IEEE ICASSP*, 2021, pp. 786–790.

[29] V. Neo, C. Evers, S. Weiss, and P. A. Naylor, "Polynomial matrix eigenvalue decompositionexploiting spherical microphone array processing," *IEEE Trans. SP*, submitted 2022.

[30] C. H. Ta and S. Weiss, "A design of precoding and equalisation for broadband MIMO systems," in *Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, pp. 1616–1620, Nov. 2007.

[31] W. Al-Hanafy, A. P. Millar, C. H. Ta, and S. Weiss, "Broadband SVD and non-linear precoding applied to broadband MIMO channels," in *Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, pp. 2053–2057, Oct. 2008.

[32] N. Moret, A. Tonello, and S. Weiss, "MIMO precoding for filter bank modulation systems based on PSVD," in *IEEE VTC*, May 2011.

[33] D. Hassan, S. Redif, J.G. McWhirter, and S. Lambotharan, "Polynomial gsvd beamforming for two-user frequency-selective mimo channels," *IEEE Trans. SP*, **69**:948–959, Jan. 2021.

[34] S. Weiss, C. Delaosa, J. Matthews, I. Proudler, and B. Jackson, "Detection of weak transient signals using a broadband subspace approach," in *SSPD*, Edinburgh, Scotland, pp. 65–69, Sep. 2021.

[35] V.W. Neo, S. Weiss, and P.A. Naylor, "A polynomial subspace projection approach for the detection of weak voice activity," in *SSPD*, London, UK, pp. 1–5, Sep. 2022,

[36] V. W. Neo, S. Weiss, S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Polynomial eigenvalue decomposition-based target speaker voice activity detection in the presence of competing talkers," in *IWAENC*, Bamberg, Germany, Sep. 2022.

[37] C. Delaosa, J. Pestana, N.J. Goddard, S. Somasundaram, and S. Weiss, "Sample space-time covariance matrix estimation," in *ICASSP*, pp. 8033–8037, May 2019.

[38] T. Kato, *Perturbation Theory for Linear Operators*. Springer, 1980.

[39] G.W. Stewart and J.-g. Sun, *Matrix Perturbation Theory*. Academic Press, 1990.

[40] C. Delaosa, F.K. Coutts, J. Pestana, and S. Weiss, "Impact of space-time covariance estimation errors on a parahermitian matrix EVD," in *IEEE SAM*, Sheffield, UK, July 2018.

[41] C. Delaosa, J. Pestana, S. Weiss, and I.K. Proudler, "Subspace perturbation bounds with an application to angle ofarrival estimation using the music algorithm," in *SSPD*, Edinburgh, Scotland, 2020.

[42] J.G. McWhirter and Z. Wang, "A novel insight to the SBR2 algorithm for diagonalising para-hermitian matrices," in *11th IMA Conf. Math. in SP*, Birmingham, UK, Dec. 2016.

[43] C. Delaosa, J. Pestana, N. J. Goddard, S. D. Somasundaram, and S. Weiss, "Support estimation of a sample space-time covariance matrix," in *SSPD*, Brighton, UK, pp. 1–5, May 2019.

[44] F.K. Coutts, J. Corr, K. Thompson, S. Weiss, I.K. Proudler, and J.G. McWhirter, "Memory and complexity reduction in parahermitian matrix manipulations of PEVD algorithms," in *EUSIPCO*, Budapest, Hungary, Aug. 2016.

[45] F.K. Coutts, J. Corr, K. Thompson, S. Weiss, I.K. Proudler, and J.G. McWhirter, "Complexity and search space reduction in cyclic-by-row PEVD algorithms," in *Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2016.

[46] F.K. Coutts, I.K. Proudler, and S. Weiss, "Efficient implementation of iterative polynomial matrix evd algorithms exploiting structural redundancy and parallelisation," *IEEE Trans. CAS II*, **66**(12):4753–4766, Dec. 2019.

[47] F.A. Khattak, S. Weiss, and I.K. Proudler, "Fast givens rotation approach to second order sequential best rotation algorithms," in *SSPD*, Edinburgh, Scotland, pp. 40–44, Sep. 2021.

[48] C. H. Ta and S. Weiss, "Shortening the order of paraunitary matrices in SBR2 algorithm," in *ICICSP*, Singapore, Dec. 2007.

[49] J. Corr, K. Thompson, S. Weiss, I. Proudler, and J. McWhirter, "Row-shift corrected truncation of paraunitary matrices for PEVD algorithms," in *EUSIPCO*, Nice, France, pp. 849–853, Aug. 2015.

[50] ——, "Shortening of paraunitary matrices obtained by polynomial eigenvalue decomposition algorithms," in *SSPD*, Edinburgh, Scotland, Sep. 2015.

[51] B. Girod, R. Rabenstein, and A. Stenger, *Signals and Systems*. Chichester: J. Wiley & Sons, 2001.

Back to Contents

Fast Trajectory Forecasting With Automatic Identification System Broadcasts

Yicheng Wang and Murat Üney
Department of Electrical Engineering and Electronics
University of Liverpool
L69 3GJ, Liverpool, United Kingdom
Emails: { Y.Wang496, M.Uney } @ liverpool.ac.uk

***Abstract*—This work proposes a fast trajectory forecasting algorithm to use with automatic identification system (AIS) broadcasts of vessels. The algorithm involves fast sub-optimal model parameter estimation from AIS messages and the computation of Gaussian location predictions for a series of future timestamps. The underlying trajectory model is a stochastic process that uses six parameters to generate near-constant velocity trajectories. These parameters include the desired cruise heading and speed of the vessel and velocity standard deviations along the heading direction and its perpendicular complement. We demonstrate the performance of our approach using a real AIS data set.**

I. Introduction

Accurate predictions of vessel locations are very useful in maritime traffic safety [1], surveillance [2] and situational awareness [3], [4] applications. Forecasting is often performed using generative trajectory models. Stochastic process models offer advantages in capturing the physics of motion and the uncertainties involved: Examples include Gaussian processes [2], bridging density models [5], [6], Ornstein-Uhlenbeck (OU) process velocity models [7] and data-driven change-point models [8].

This work is motivated by the real-time availability of secondary surveillance data such as Automatic Identification System (AIS) broadcasts from vessels. These broadcast messages inform recipients on the position of the transmitting vessel as measured by the Global Positioning System (GPS), its velocity vector in terms of its speed and heading angle with respect to North, vessel's class and similar traits, all tagged by the vessel's unique identification number – the maritime mobile service identity (MMSI). AIS messages from the same vessel thus form a trajectory data stream using which its future state can be predicted.

Stochastic process models provide future position predictions by extracting model parameters from data streams and using these parameters in the model to evaluate the statistics of the process at a future time instant. To model 2-D trajectories observed at arbitrary time instants, [9] introduced a 2-D OU velocity model with six parameters and studied their maximum likelihood (ML) estimation and Cramér-Rao lower bounds (CRLB). ML estimation in this model is a constrained problem which can be solved using second order iterative methods. The iterations, however, must start at a good initial point for convergence to the global optimum.

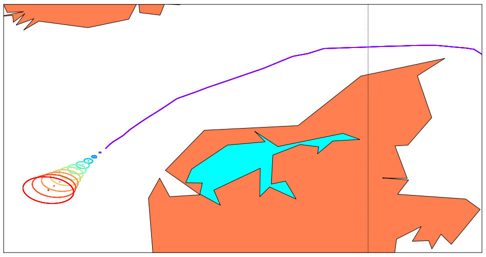


Fig. 1. Example location forecasts and uncertainty ellipses for multiple time steps separated by 1 000 s starting from the last message.

In this work, we propose a fast non-iterative approach to find the parameters of AIS trajectories. We use these estimates within the six-degree-of-freedom model to make future location forecasts – an example is depicted in Fig. 1. Then, we perform a performance study using an AIS data set that contains all the cargo ship messages recorded during February 2022 and made publicly available by the Danish Maritime Authority [10].

The structure of the article is as follows: Section II gives the mathematical problem definition followed by the six-degree-of-freedom model used in Section III. The proposed fast parameter estimation method is detailed in Section IV. The results of the performance study performed using a real AIS data set are given in Section V. Then, in Section VI we conclude.

II. Problem Definition

A. AIS trajectories

Trajectory observations of a vessel are provided by its AIS broadcasts tagged by the MMSI number. In particular, these messages report the latitude and the longitude of the vessel and its velocity vector. For processing, we consider projected versions of these quantities on a plane using the universal transverse Mercator (UTM) projection. The concatenation of the projected position and velocity yield the target state vector. The continuous trajectory of the vessel is a time function of this evolving state vector $x(t)$. Thus, L observations form the continuous trajectory $\mathbf{x}$ form a $4 \times L$ array $\mathbf{x} = [\mathbf{x}_1, ..., \mathbf{x}_L]$

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

where samples are collected at $\boldsymbol{t} = [t_1, \ldots, t_L]$, i.e., $\mathbf{x}_k = x(t_k)$ for $k = 1, \ldots, L$.

B. Forecasting with uncertainty quantification

The problem of forecasting is to find an estimate of the state vector $x(t_f)$ where t_f is a time stamp in the future, i.e., $t_f > t_L$. Let us denote this estimate by $\hat{x}_{t_f}$. We are also interested in finding the uncertainty associated with this estimate, specifically a Gaussian distribution with density $\mathcal{N}(.; \hat{x}_{t_f}, \mathbf{\Sigma}_{t_f})$ centred at the position and velocity forecast and distributing the probability mass according to the covariance matrix $\mathbf{\Sigma}_{t_f}$.

III. The trajectory model

We use the stochastic generative model introduced in [9] to model $\mathbf{x}$ which is based on OU velocity processes. The underlying assumption is that any vessel would maintain a cruise velocity V and speed S related by

$$V = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} V_C \tag{1}$$

$$V = \begin{bmatrix} v_E \\ v_N \end{bmatrix}, \; V_C = \begin{bmatrix} S \\ 0 \end{bmatrix}, \tag{2}$$

where α is the cruise heading angle the vessel aims to maintain and V_C is the cruise velocity in the cruise coordinate system. In other words, the cruise coordinates of the vessel with its front as the first axis and the left perpendicular axis as the second axis is rotated by α rad with respect to the East-North plane.

In the cruise coordinate frame, the velocity coordinate along the first axis is an OU process [11] with a mean value equals to the speed S, i.e.,

$$\dot{v}_1(t) = \gamma_1 \left(S - v_1(t)\right) + \sigma_1 \dot{n}_1(t), \tag{3}$$

where n_1 is a Wiener process. Along the second axis, the mean value to maintain is zero (following (2) and (1)), i.e.,

$$\dot{v}_2(t) = -\gamma_2 v_2(t) + \sigma_2 \dot{n}_2(t), \tag{4}$$

where n_2 is a Wiener process independent from n_1. As a result

$$v_1(t) = v_1(0)e^{-\gamma_1 t} + S\left(1 - e^{-\gamma_1 t}\right) + \sigma_1 \int_0^t e^{-\gamma_1 t} \mathrm{d}n_1, \tag{5}$$

along the heading direction and

$$v_2(t) = v_2(0)e^{-\gamma_2 t} + \sigma_2 \int_0^t e^{-\gamma_1 t} \mathrm{d}n_2, \tag{6}$$

along the perpendicular direction. The position vector generated by these velocities equals their integration over time. The block diagram of these processes is illustrated in Fig. 2.

The stochastic processes v_1 and v_2 in (3) and (4) will deviate around S and 0, respectively, due to the stochasticity input by the Wiener processes. This is easy to see if we discretise (5) and (6) by uniform sampling with period Δt yields the following difference equations:

$$v_1(t + \Delta t) = v_1(t)e^{-\gamma_1 \Delta t} + S\left(1 - e^{-\gamma_1 \Delta t}\right) + \epsilon_1, \tag{7}$$

$$v_2(t + \Delta t) = v_2(t)e^{-\gamma_2 \Delta t} + \epsilon_2, \tag{8}$$

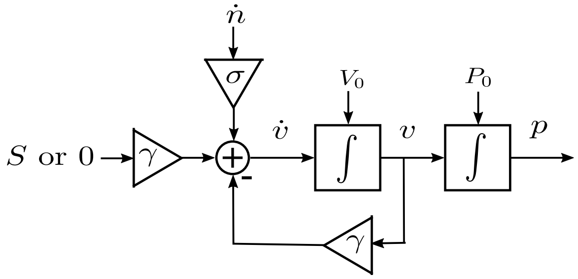


Fig. 2. An OU velocity process generating position with parameters S , γ, σ and initial conditions given by V_0 and P_0.

where the noise terms are normal with $\epsilon_1 \sim \mathcal{N}(.; 0, \frac{1}{\gamma_1}\sigma_1^2(1 - e^{-2\gamma_1 \Delta t}))$ and $\epsilon_2 \sim \mathcal{N}(.; 0, \frac{1}{\gamma_2}\sigma_2^2(1 - e^{-2\gamma_2 \Delta t}))$.

Here, the deviations around S and 0 are first order autoregressive processes, respectively, and σ_1 and σ_2 specify the standard deviations in these processes whereas γ_1 and γ_2 determine how fast step function like deviations tend to decay.

This model allows for prediction of the position at a future time by inducing a probability density over the state when the initial conditions of the stochastic differential equations are set to the last observation values. Given α and S, starting from the last observed position $P_0 = [p_{0,1}, p_{0,2}]^T$ and velocity $V_0 = [v_{0,1}, v_{0,2}]^T$ at time t_0 in the cruise coordinate frame, the probability density over the state (i.e., the concatenation of the position and velocity) at time t_f is given by $\mathcal{N}(.; \hat{x}_{t_f}, \mathbf{\Sigma}_{t_f})$ where [9]:

$$\hat{x}_{t_f} = \begin{bmatrix} p_{0,1} + \frac{1-e^{-\gamma_1 \Delta t}}{\gamma_1} v_{0,1} + (\Delta t - \frac{1-e^{-\gamma_1 \Delta t}}{\gamma_1})S \\ p_{0,2} + \frac{1-e^{-\gamma_2 \Delta t}}{\gamma_2} v_{0,2} \\ v_{1,0}e^{-\gamma_1 \Delta t} + S(1 - e^{-\gamma_1 \Delta t}) \\ v_{2,0}e^{-\gamma_2 \Delta t} \end{bmatrix}, \tag{9}$$

where $\Delta t = t_f - t_0$.

The variances associated with position variables in the state forecast formula in (9), i.e., the first two diagonal entries in $\mathbf{\Sigma}_{t_f}$ are given by [9]

$$\sigma_{p,1}^2 = \frac{\sigma_1^2}{\gamma_1^3}(2e^{-\Delta t \gamma_1} - e^{-2\Delta t \gamma_1}/2 + \Delta t \gamma_1 - 3/2) \tag{10}$$

$$\sigma_{p,2}^2 = \frac{\sigma_2^2}{\gamma_2^3}(2e^{-\Delta t \gamma_2} - e^{-2\Delta t \gamma_2}/2 + \Delta t \gamma_2 - 3/2).$$

The above model is characterised by six parameters $\theta = [\alpha, S, \gamma_1, \sigma_1, \gamma_2, \sigma_2]$, where α is the heading of the route the vessel aims to follow, S is the speed aimed at, σ_1 specifies the magnitude of the stochastic fluctuations around S and γ_1 is the reciprocal system time constant along the route. Similarly, σ_2 tunes the standard deviation of the deviations in the direction perpendicular to the route and γ_2 is the reciprocal time constant along this direction.

The next section discusses fast estimation of these parameters from AIS tracks to evaluate the forecast equations (9) and (10) at selected future time instants.

Back to Contents

IV. The fast forecasting algorithm

The proposed approach is motivated by the non-iterative maximum likelihood estimation results in [12] for 1-D OU processes observed at uniformly spaced time instances. Given α and S in the parametric model detailed in the previous section, the estimation problem is split into two estimation problems for finding the 1-D OU process parameters: The first one is the process along the heading direction α and the second one is the process in the perpendicular direction.

A. Sub-optimal model parameter estimation

1) Estimation of the heading α, S: Given L AIS trajectory observations $\mathbf{x} = [\mathbf{x}_1, ..., \mathbf{x}_L]$, we use the sample average of the velocity fields in $\mathbf{x}$, i.e., third and the fourth rows, to estimate the cruise velocity V. Let us denote this estimate by $\hat{V} = [\hat{v}_E, \hat{v}_N]^T$. The cruise heading and speed estimates $\hat{\alpha}$ and $\hat{S}$ follow as

$$\begin{aligned} \hat{S} &= \sqrt{\hat{v}_E^2 + \hat{v}_N^2}, \\ \hat{\alpha} &= \arctan(\hat{v}_N, \hat{v}_E). \end{aligned} \tag{11}$$

2) Estimation of the OU process parameters: After finding $\hat{\alpha}$, we use the transformation

$$\mathbf{x}_C = \begin{bmatrix} \mathbf{E}^T, & \mathbf{0} \\ \mathbf{0}, & \mathbf{E}^T \end{bmatrix} \mathbf{x}, \tag{12}$$

$$\mathbf{E} = \begin{bmatrix} \cos\hat{\alpha} & -\sin\hat{\alpha} \\ \sin\hat{\alpha} & \cos\hat{\alpha} \end{bmatrix}, \tag{13}$$

to find the coordinates of the state vectors in the cruise coordinate system — here, $\mathbf{x}_C$ denotes the target state in the cruise coordinate system. The third and the fourth row of $\mathbf{x}_C$ can now be treated as independent 1-D OU processes along the direction of the cruise and its perpendicular complement, respectively.

Following the results in [12], γ_1 of the first process is found by first finding $y(l) = \mathbf{x}_{C,3}(l) - \hat{S}$ where $\mathbf{x}_{C,3}(l)$ is the lth entry in the third row of the AIS trajectory $\mathbf{x}$, and then computing

$$\hat{\gamma}_1 = \frac{-1}{\bar{\Delta} t} \log \left| \frac{\sum_{l=2}^{L} y(l) y(l-1)}{\sum_{l=2}^{L} y(l)^2} \right|, \tag{14}$$

where $\bar{\Delta} t$ is the average time step between two messages, i.e., $\bar{\Delta} t = 1/(L-1) \sum_{l=2}^{L} t_l - t_{l-1}$.

The standard deviation σ_1 is estimated by [12]

$$\hat{\sigma}_1 = \Big(\frac{2\hat{\gamma}_1}{L(1 - e^{-2\hat{\gamma}_1 \bar{\Delta} t})} \sum_{l=2}^{L} (y(l) - y(l-1) e^{-2\hat{\gamma}_1 \bar{\Delta} t})^2 \Big)^{1/2}. \tag{15}$$

The parameters of the second process are similarly estimated by first assigning $y(l) = \mathbf{x}_{C,4}(l)$ and then computing the right-hand-side of (14) and (15) to find $\hat{\gamma}_2$ and $\hat{\sigma}_2$, respectively.

TABLE I
Statistics of cargo ship messages used.

	Time length (h)	Number of AIS messages per MMSI trajectory	Speed $(\mathrm{m\,s^{-1}})$
Average	13.76	4 576.52	5.46
Maximum	23.99	31 367	11.95
Minimum	0.21	101	0

B. The algorithm

The computational steps of the algorithm are as follows.

1) Fetch the AIS messages for a selected MMSI, use UTM conversion to create the AIS trajectory $\mathbf{x}$.
2) Find the cruise heading and speed as described in Section IV-A1.
3) Find the OU process parameters as described in Section IV-A2.
4) Fetch the future time stamps from the user and evaluate the future forecast equations in (9) and (10).
5) Display the position forecast (i.e., the first two fields of (9)) and the uncertainty ellipse implied by the standard deviations in (10).

C. Implementation

The forecast algorithm is implemented using the Python programming language. The system stores AIS data files into a database and makes MMSI queries and similar operations through the data base. For example, the message database can be queried to select only messages from cargo ships. After entries in the database are read, UTM conversion is used to load AIS trajectories in the memory (Section II-A). The user enters an MMSI to run the forecasting algorithm. Fig. 1 shows an example in which forecast positions (crosses in different colours) and uncertainty ellipses for the blue AIS trajectory for 1 000 s time steps from the last message are computed. It can be seen that the uncertainty becomes larger for longer forecast times.

V. Performance Study

In this section, we demonstrate our approach using a real AIS data set.

A. The data set

The data set consists of AIS messages received and recorded by the Danish Maritime Authority [10]. In particular, a subset that contains messages from only cargo ships with more than 100 consecutive AIS messages in February 2022 are used. Fig. 3 illustrates the corresponding AIS trajectories and TABLE I gives fundamental statistics of this data set.

B. The test set-up and metrics

The performance study is based on backtesting, i.e., the prediction model is tested using historical data. The aforementioned AIS trajectories are divided into two segments: The first segment holds the first 80% of the entire trajectory. The remaining 20% of the trajectories are used to quantify the

Back to Contents

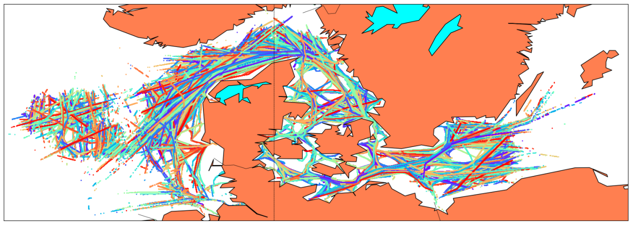


Fig. 3. All cargo ship messages recorded in February 2022 by the Danish Maritime Authority [10]. Messages with different MMSIs correspond to different ships and are depicted in different colours.

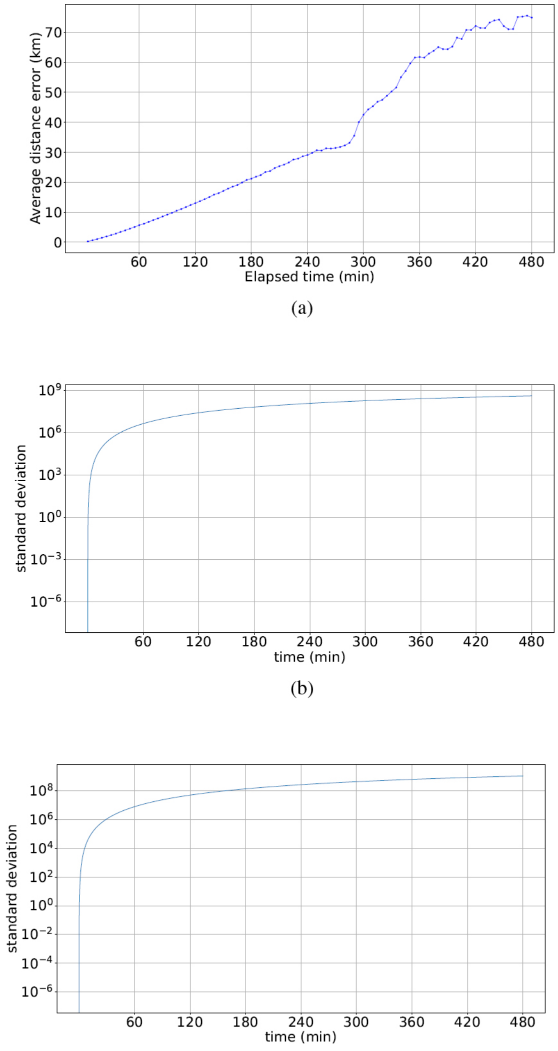
forecasts based on the first segments of the trajectories, i.e., the second segment is the validation set.

We use two methods to assess the accuracy of forecasts: First, we find the distance error by finding the Euclidean distance of the forecast positions to the ground truth of the second segments of the trajectories. Second, we find "hits" as a binary metric that indicates whether the ground truth is inside the ellipse surrounding the significant probability mass of the Gaussian uncertainty associated with the forecast. In particular, we find ellipses centred at the forecast position and have $4\sigma_1, 4\sigma_2$ semi-minor/major axes for time horizons up to one hour, and $3\sigma_1, 3\sigma_2$ for time horizons larger than one hour, respectively.

C. Results

Fig. 4(a) depicts the average forecast error versus the length of the time step into the future. The average mislocation increases linearly for approximately 250 minutes. Given that the distance to horizon is approximately 5.1km, average errors stay within the horizon limits up until 50 minutes. The prediction uncertainties along the two directions as quantified by the OU standard deviation parameters $\hat{\sigma}_1$ and $\hat{\sigma}_2$ are given in Fig. 4(b) and Fig. 4(c), respectively, in the logarithmic scale. Note that during the first 60 min the standard deviation rapidly increases departing from overconfidence.

The predicted position and the associated uncertainty are used to calculate hourly hit rates (see, Section V-B). Hit rates versus increasing time in the hours scale is given in Fig. 5. The bar plots indicate that predictive models found using the proposed approach are overconfident until an hour long prediction time. For prediction times longer than an hour, the confidence of the predicted models become more reasonable peaking at almost 70% hit rate for position forecasts between $2-3$ hours into the future. For predictions up to and including one hour, the hit rate is much smaller despite the use of a radius of 4σ. This points at the overconfidence of the predictive model. Note that the model errors both in average error and the computed uncertainty become unreliable after 6 hours: The error regime in the average errors in Fig. 4a becomes non-linear and despite increasing standard deviation in Fig. 4b and Fig. 4c, the hit rate in Fig. 5 deteriorates pointing to a loss of predictive power of the model in these time scales into the future.



(a)

(b)

(c)

Fig. 4. (a) Average prediction error versus the length of time step into the future. (b) Predictive model standard deviation along the cruise direction with heading $\hat{\alpha}$ versus the prediction time.(c) Predictive model standard deviation along the direction perpendicular to the cruise direction versus prediction time.

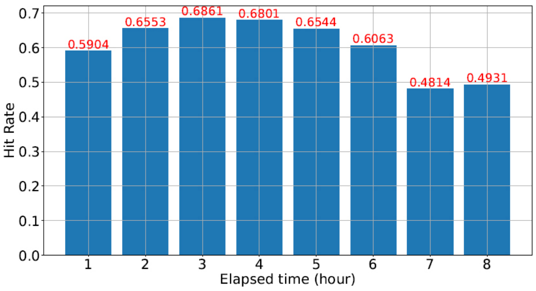


Fig. 5. Prediction hit rate versus prediction time.

Back to Contents

VI. Conclusions

In this article, we proposed a fast trajectory forecasting algorithm which works with Automatic Identification System data streams. Using historical data for testing, we discussed the reliability of this forecast algorithm through the average distance error, uncertainties, and hit rate. We also give analysis for the obtained forecast accuracy. This algorithm is sub-optimal and can be further used to initiate iterative algorithms to find optimal ML solutions.

Acknowledgment

This work was supported by the ELEC340 BENG project module at the Department of Electrical Engineering and Electronics of the University of Liverpool.

References

[1] Z. Xiao, X. Fu, L. Zhang, and R. S. M. Goh, “Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1796–1825, 2020.

[2] B. Ristic, B. L. Scala, M. Morelande, and N. Gordon, “Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction,” in *Fusion’08*, June 2008, pp. 1–7.

[3] G. Pallotta, M. Vespe, and K. Bryan, “Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction,” *Entropy*, vol. 15, no. 6, pp. 2218–2245, 2013.

[4] M. Uney, L. M. Millefiori, and P. Braca, “Prediction of rendezvous in maritime situational awareness,” in *2018 21st International Conference on Information Fusion (FUSION)*, July 2018, pp. 622–628.

[5] B. I. Ahmad, J. K. Murphy, P. M. Langdon, and S. J. Godsill, “Bayesian intent prediction in object tracking using bridging distributions,” *IEEE Transactions on Cybernetics*, vol. 48, no. 1, pp. 215–227, Jan 2018.

[6] J. Liang, B. I. Ahmad, R. Gan, P. Langdon, R. Hardy, and S. Godsill, “On destination prediction based on markov bridging distributions,” *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1663–1667, 2019.

[7] S. Coraluppi and C. Carthel, “Stability and stationarity in target kinematic modeling,” in *2012 IEEE Aerospace Conference*, 2012, pp. 1–8.

[8] M. Uney, L. M. Millefiori, and P. Braca, “Data driven vessel trajectory forecasting using stochastic generative models,” in *2019 ICASSP*. IEEE, May 2019.

[9] ——, “Maximum likelihood estimation in a parametric stochastic trajectory model,” in *2019 Sensor Signal Processing for Defence Conference (SSPD)*, 2019, pp. 1–5.

[10] [Online]. Available: https://dma.dk/safety-at-sea/navigational-information/ais-data

[11] S. Sarkka and A. Solin, *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

[12] T. Karvonen, F. Tronarp, and S. Sarkka, “Asymptotics of maximum likelihood parameter estimates for gaussian processes: The Ornstein-Uhlenbeck prior,” in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2019, pp. 1–6.

Deep Learning for Spectral Filling in Radio Frequency Applications

1st Matthew Setzler* 2nd Elizabeth Coda* 3rd Jeremiah Rounds* 4th Michael Vann* 5th Michael Girard*

***Abstract*—Due to the Internet of Things (IoT) proliferation, Radio Frequency (RF) channels are increasingly congested with new kinds of devices, which carry unique and diverse communication needs. This poses complex challenges in modern digital communications, and calls for the development of technological innovations that (i) optimize capacity (bitrate) in limited bandwidth environments, (ii) integrate cooperatively with already-deployed RF protocols, and (iii) are adaptive to the ever-changing demands in modern digital communications. In this paper we present methods for applying deep neural networks for *spectral filling*. Given an RF channel transmitting digital messages with a pre-established modulation scheme, we automatically learn novel modulation schemes for sending extra information, in the form of additional messages, "around" the fixed-modulation signals (i.e., without interfering with them). In so doing, we effectively increase channel capacity without increasing bandwidth. We further demonstrate the ability to generate signals that closely resemble the original modulations, such that the presence of extra messages is undetectable to third-party listeners. We present three computational experiments demonstrating the efficacy of our methods, and conclude by discussing the implications of our results for modern RF applications.**

***Index Terms*—deep learning, signal generation, communications, machine learning, radio frequency**

I. Introduction

The Internet of Things (IoT) proliferation poses novel, and complex challenges for digital communications [2], [13], [15]. Radio Frequency (RF) channels are increasingly congested with new kinds of devices, which carry unique communication needs [16]. Meeting these challenges requires the development of new technologies that (i) optimize capacity in limited bandwidth environments, (ii) integrate seamlessly with existing, already-deployed communications protocols, and (iii) are adaptive to the continuous flux in consumption requirements of modern digital comms environments.

Here we present novel methods for applying deep neural networks (DNNs) for *spectral filling*. Given an RF channel transmitting digital messages via some pre-established modulation scheme, we show that we can automatically learn novel modulation schemes to send extra information, in the form of an additional message, "around" the fixed-modulation signals (i.e., without interfering with them), thus increasing channel capacity without increasing bandwidth. We further demonstrate the ability to constrain the spectral shape of learned signals, such that they resemble the original modulations or conform to arbitrary spectral shapes.

Recent years have seen a nascent, but growing interest in leveraging deep learning for RF applications. One such application is "spectrum sensing", where DNNs are trained to classify the modulations of signals in an RF environment [3], [22]. Neural networks have also been trained to demodulate RF signals [1], [12], [14], [18], [19], [26], and even for end-to-end communications systems, although success of these efforts has been mixed [7], [20]. Despite these early efforts, deep learning in RF applications is still a relatively unexplored area, and much remains to be learned about what kinds of model architectures are well-suited to the RF domain and what kinds of problems DNNs are apt to address.

In particular, the spectrum filling problem introduced in this paper has not yet been addressed by the research community. Earlier efforts have shown that DNNs can be used in model communications systems, but it is not clear how they would be deployed in real-world scenarios, in which the learned RF signals would need to cooperate with existing signals defined by pre-established modulation protocols. Conversely, in the present work, insofar as we are able to learn modulations that adapt to existing RF protocols, we demonstrate the suitability of our methods to be integrated with already-deployed communications systems in the wild.

Our work also differs from previous efforts in that our DNN architectures utilize Transformer networks, which have proven to be powerful architectures for modeling temporal relationships in time-series data such as NLP, music, and signal processing [5], [6], [10], [25]. This is a departure from previous efforts, which have typically used convolutional networks [21], [27], which were originally developed in computer vision [9], [11], and thus not optimally-suited for modeling time-series. There has been some work on applying autoregressive Long-Short Term Memory (LSTM) networks to RF data [3], [22], but these efforts lag behind the state-of-the-art in deep learning, because LSTMs are almost unanimously outperformed by Transformers in a variety of time-series applications [8], [25]. To our knowledge, there has only been one previous application of transformers to the RF domain [24], which showed promising results, though it was not geared towards the problem addressed here: spectral filling.

A. Problem Statement: Spectral Filling

We considered a scenario where two radios communicate over a traditional digital signal pipeline (see Supporting Information for a high-level schematic). This communication scheme is bounded in its capacity by Shannon's Limit [23],

*Pacific Northwest National Labs; Washington, USA
correspondance to: mattsetz@gmail.com and michael.girard@pnnl.gov
supporting information: https://arxiv.org/pdf/2204.01536.pdf

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

meaning that for the bandwidth the radios are using and the amount of noise in their environment, the speed that they can transmit information is fixed. This is defined by the equation

$$C = B \log_2 \left(1 + \frac{S}{N}\right) \tag{1}$$

where C is the capacity in bits/sec, B is the bandwidth in Hz, S and N are the power in the signal and noise respectively. Modern digital communication schemes can come close to this limit, however there is usually a gap in the actual speed of data transmission and the theoretical maximum. This means that there is the possibility for extra data to be transmitted alongside the fixed, traditional scheme.

However, another important theorem of digital communications that stops full utilization of this gap is the central coding limit theorem. The coding limit theorem states that while the rate, R of data transfer in a channel is less than Shannon's capacity, $R < C$, the rate at which errors occur in the communication channel can be made arbitrarily small. If the rate exceeds the Shannon limit then the error rate will be, in general, large. We plan to exploit this gap in actual vs. theoretical rates of communication, while still being able to make the error rates of communication small.

We label a traditional digital communication signal from one radio to the other as the A message. This consists of a sequence of ones and zeros and is generally long. If this signal does not reach Shannon's limit than there is the possibility for a second message that uses some of the unused bandwidth. This is the B message but is generally not as long as the A message. We have developed a novel method for generating a time series that can transmit these two different types of messages without greatly affecting the accuracy of the A message.

A secondary goal of ours is to constraint properties of learned signals using auxiliary loss terms. In Experiment 1, we constrain learned signals to resemble the original modulations, such that a third-party would not be able to identify the presence of message B based on spectral properties or other signatures of the generated signals. In Experiment 2 we go one step further and show that it is possible to constrain learned signals to match to arbitrary spectral shapes, while still retaining the ability to transmit both messages. In the remainder of this paper we specify our methods, report results from three experiments demonstrating success with respect to each of our goals, and conclude by discussing the implications for modern RF applications.

II. Methods

Our goal is to transmit an RF signal[1] that carries information from two messages (A and B) over-the-air. Both messages are sequences of discrete symbols. Experiments 1 & 2 utilize Quadrature Shift Keying (QPSK), Message A comprises four symbols. As reported in Section A, we also ran a preliminary experiment utilizing Binary Phase Shift Keying (BPSK), in which Message A comprises two symbols. In all experiments, Message B was a binary sequence. The lengths of messages A and B need not be equal, and we refer to length of A message as $length_A$ and length of B message as $length_B$ ($length_B$ is typically shorter than $length_A$). In all experiments we assume a sample rate of 1 Hz and oversampling of 1 with respect to A, such that $length_A$ is equal to the number of IQ samples in the signal.

A. Model Architecture

Our model includes two transformer-based DNNs — the Modulator and Demodulator networks. These networks are jointly trained to modulate and demodulate extra information from message B without degrading the original signal carrying message A. The model also includes fixed modules for modulation and demodulation of message A, as well as a channel model that simulates Additive White Guassian Noise (AWGN). Complete details and a block diagram of the model architecture are included in the Supporting Information.

Message A is first modulated with a standard RF protocol, such as BPSK or QPSK. This yields a signal — an IQ sequence of dimensionality (2, $length_A$) — which we denote IQ_A. The Modulator Network receives IQ_A and message B as inputs, and outputs IQ_{AB}, an IQ signal encoding information from both messages. IQ_{AB} is then passed through the channel model, which applies AWGN according to a specified signal-to-noise-ratio (SNR), producing $IQ_{channel}$, a noised signal representing what would be received over-the-air.

The received signal is then separately demodulated by a fixed module, which uses standard demodulation (either BPSK or QPSK) to recover Message A, and the Demodulator Network, which predicts bits in Message B. The discrepancy between ground-truth and predicted symbols in messages A and B serve as two loss terms for training our models, as described in Section II-B. Note that the fixed demodulator is completely naive to the learned modulation; it processes the transmitted signal as if it were a typical BPSK or QPSK signal. Therefore, in order to achieve high accuracy with respect to message A, the Modulator Network must not interfere with the fixed modulation.

B. Training Procedure

Our models were jointly trained to minimize two loss terms: $loss_A$ and $loss_B$. For $loss_A$ we took the binary cross-entropy (BCE) loss of each IQ sample in $IQ_{channel}$ compared to the original IQ_A. For $loss_B$ we took BCE of each prediction logit in the output of the Demodulator Network compared to the bits in the ground-truth Message B. In both cases, prediction logits were passed through a sigmoid function before BCE was computed. These two loss terms were combined into a single loss function that implicitly encouraged the Modulator Network to modulate message B in such a way that it did not degrade original QPSK message. The overall loss is:

[1] RF signals typically comprise two orthogonal components, I and Q, which can be thought of as cosine and sine components of a complex waveform. Sampling from these components yields a two-dimensional IQ sequence, which for the purposes of this paper is synonymous with an RF signal.

$$loss = \alpha\ loss_A + (1 - \alpha)\ loss_B \tag{2}$$

where α tunes the degree to which $loss_A$ is weighted with respect to $loss_B$.

Through preliminary experimentation, we found it was best to initialize $\alpha = 1$ at the beginning of training (keeping it fixed at 1 for first three epochs), and then gradually decrease it over subsequent epochs (at a rate of 0.01 per epoch) until it reached $\alpha = 0.5$. This encouraged the model to first minimize $loss_A$ — which should be trivial, since the Modulator Network is given the ground truth QPSK IQ values for message A, and can in principle learn to ignore message B — and then gradually learn to include information from message B without degrading the original IQ sequence. We also experimented with different auxiliary losses for constraining various properties of the generated signals, as described in subsequent sections.

A dataset consisting of 16,384 examples was synthesized. Each example consisted of a tuple of (message A, IQ_A, message B). 80% of these examples were used for training, and the remaining 20% were held out as a test set. Unless otherwise reported, the batch size was 64, and SNR was varied across all examples within each batch by sampling over a uniform distribution ranging from 5–15 dB. The AdaBelief optimizer was used with a learning rate of 0.01 [28]. Models were trained for 128 epochs, unless otherwise specified.

III. Results and Discussion

Experiment 1: Constraining learned signals in time-domain (QPSK)

In this experiment we used an auxiliary loss term to explicitly encourage the model to generate signals resembling the original QPSK signal (IQ_A). We used mean-squared error (MSE) on the learned IQ sequence ($signal_{combined}$), with respect to the original QPSK signal (IQ_A). This loss term is denoted $loss_{MSE}$, and it was incorporated into the overall loss function as defined by the equation:

$$loss = \frac{\alpha}{2}\ loss_A + (1 - \alpha)\ loss_B + \frac{\alpha}{2}\ loss_{MSE} \tag{3}$$

This closely resembles Equation 2, except that the weight of α is equally distributed across $loss_A$ and $loss_{MSE}$. This was done because these two loss terms are complementary – constraining $signal_{combined}$ to match IQ_A (via $loss_{MSE}$) necessarily makes it easier for a QPSK demodulator to recover Message A by processing $signal_{combined}$ as if it were a typical QPSK signal. In this sense a high value of α still biases training to optimize for Message A, and low or intermediate α values reward successfully transmitting and demodulating Message B.

Model Performance: The best model from this training run was evaluated on a held-out test set over a range of SNRs. At each SNR, we passed every example in the test set through the model, and independently evaluated Bit Error Rate (BER) of messages A and B. Results are depicted in Figure 1. The x-axis represents noise level at which our AWGN channel was simulated, expressed in E_s/N_0 (energy per symbol to noise power spectral density ratio), a normalized SNR measure. The y-axis represents empirically determined BER at each noise-level. Blue points represent BER with respect to message A, and the yellow points represent BER with respect to message B. As expected, BER decreases with more favorable noise levels, until it plateaus at an E_s/N_0 of about 8 dB. Most importantly, the model achieves an acceptably low BER for both messages, and this is robust across a range of SNRs.[2]

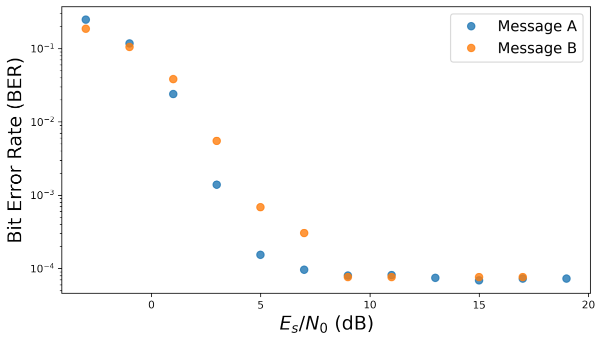


Fig. 1: High model accuracy across a range of noise levels. y-axis represents empirically determined Bit Error Rate (BER) for bits from Message A (blue points) and Message B (yellow points). x-axis represents noise level at which our AWGN channel was simulated, expressed in E_s/N_0 (energy per symbol to noise power spectral density ratio), a normalized SNR measure. Missing yellow points (at $E_s/N_0 = 13$ and $E_s/N_0 = 18$) are instances where 100% accuracy was achieved for Message B. Both messages are consistently transmitted and demodulated with high fidelity over a range of noise levels.

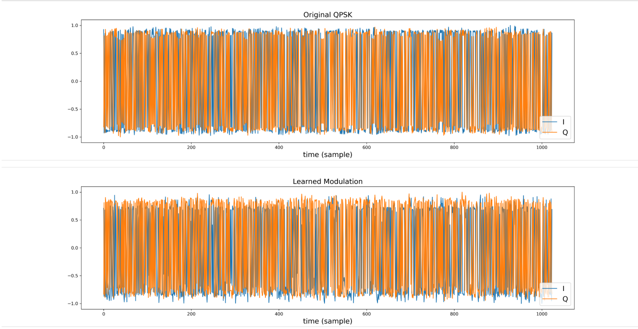


Fig. 2: Fixed (QPSK-modulated) and learned signals for an arbitrary example. The learned signal (bottom) carries Message A and B, whereas the QPSK signal (top) only carries Message A. I and Q components are colored blue and orange, respectively.

In response to our primary research question, this demonstrates the ability to successfully learn a modulation that can transmit extra information (Message B) in the same channel

[2] It is also worth noting that these BER values can be further enhanced with forward-error correction strategies [17], which would be straightforward to integrate with our model.

Back to Contents

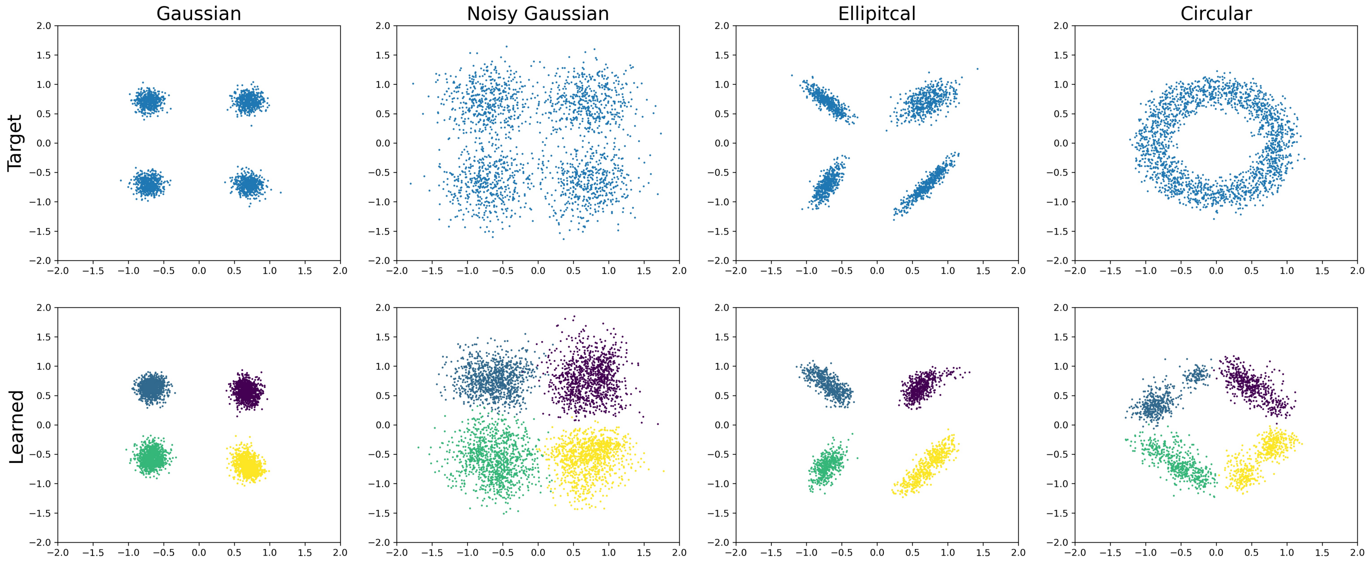


Fig. 3: Constraining learned modulation to arbitrary spectral shapes. Top row presents constellation plots of target distributions, and bottom row presents constellation plots of learned signals, color-coded by the ground-truth QPSK symbol encoded by each IQ sample. Using an auxiliary loss term, we were able to constrain generated signals to conform to arbitrary spectral shapes, while still retaining high fidelity with respect to both messages.

as a fixed-modulation signal without degrading the original signal.

Next we turn to our secondary research question: can we constrain the structure of learned signals? In this experiment we were interested in constraining the learned signal to match the original $signal_A$. To get a sense of this, we visualized examples of learned signals generated by our best-performing model, and compared them to the original QPSK signals. Figure 2 shows time-domain signals for an arbitrary example (I and Q components are colored blue and orange, respectively). The top plot shows signals corresponding to a vanilla QPSK signal carrying Message A, and the bottom plot shows the learned signal carrying Message A & B. (See the Supporting Information for constellation plots of these signals.) There is high resemblance between the two signals, indicating that not only did our model successfully learn to transmit information from both messages, it did so in such a way that the learned signals were nearly identical to original modulations. This has important implications for our methods in real world RF applications – we can learn to transmit extra information in "hidden" messages, such that the generated signals look nearly identical to typical QPSK signals from the perspective of a third-party.

Experiment 2: Constraining Constellation Plots of Learned Signals (QPSK)

We also experimented with different methods for constraining the constellation plots of learned signals. In Experiment 1, we showed it is possible to produce a learned signal that maximally resembles the fixed modulation; here we show that related techniques can also be used for arbitrary signal shapes.

To learn a particular shape, we add an auxiliary term to the loss function that encourages the distribution of values in the learned signal to match a target distribution. Given a sample of m points from a target distribution and a learned signal, we define the $n \times m$ distance matrix M as: $M_{ij} = MSE(s_i, q_j)$ where each s_i is a single value sampled from the learned signal, and each q_i is from a sample of the target distribution . The auxiliary loss is then:

$$loss_{shape} = \frac{1}{n}\sum_{i=0}^{n} min_j(M_{ij}) + \frac{1}{m}\sum_{j=0}^{m} min_i(M_{ij}) \quad (4)$$

This first sum encourages each learned signal value to be near a point in the target distribution sample. The second term ensures that the learned signal shape takes on the entire target structure. For example, in the case of a multimodal distribution, without the second loss term, the shape loss could be minimized if all the learned signal points cluster on one of the modes. Notably, this loss function does not require a closed-form density function for the target distribution so the learned signal can resemble any shape compatible with QPSK or another established communication protocol. The complete loss equation becomes:

$$loss = \alpha\ loss_A + (1 - \alpha)\ loss_B + \beta\ loss_{shape} \quad (5)$$

We have tested this with several shapes and depict the results in Figure 3. From left to right, the first two shapes were trained to resemble a QPSK signal at different noise levels. We trained at a fixed SNR of 10 dB for 50 epochs. For computational efficiency, the distance matrix was calculated

Back to Contents

using 2500 random values from the learned signal and 2500 random values from the target distribution. BER at SNR = 10 dB for the A message was 1.49e-7 and 5.19e-3 for the less noisy and noisy targets, respectively. The BER for the B message was zero for both models.

The other two shapes demonstrate the flexibility of this method. For these two shapes, we trained with SNR fixed at 10 dB for 200 epochs using the sampling adjustment described in the previous paragraph. For the elliptical distribution the BER for the A message was 1.71e-5 and for the B message was 0. For the circular distribution the BER for the A message was 2.4e-3 and 7.62e-5 for the B message. Thus, these methods allow us to constrain generated signals to conform to arbitrary spectral shapes, while still retaining high fidelity with respect to both messages.

IV. Conclusion

We have demonstrated the ability to use deep, transformer-based neural networks for "spectral filling." Given an original message (Message A), encoded with some pre-defined modulation protocol (e.g., BPSK/QPSK), these networks can learn to augment and reconstruct the IQ sequence, such that it carries an additional message (Message B) without degrading the original signal. This has promising implications for congested IoT applications, as it establishes a methodology for increasing the capacity of existing fixed-bandwidth RF channels without costly human-engineered protocols, and without disrupting existing communications protocols. This last point is crucial, because a major challenge in leveraging generative deep learning for RF applications is how to deploy these technologies without disrupting pre-established RF environments.

We have further demonstrated that with the help of auxiliary loss terms, it is possible to constrain learned signals to closely resemble the original signals, or to match arbitrary spectral shapes, while still transmitting information from both messages at high fidelity. The fact that extra information can be sent without significantly altering the original signal means this technique can be used in sensitive contexts, to send additional *in cognito* messages, undetectable to third-party listeners.

References

[1] M. R. Amini and E. Balarastaghi, "Improving ann bfsk demodulator performance with training data sequence sent by transmitter," in *2010 Second International Conference on Machine Learning and Computing*. IEEE, 2010, pp. 276–281.

[2] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," *Wireless personal communications*, vol. 58, no. 1, pp. 49–69, 2011.

[3] S. Chandhok, H. Joshi, S. J. Darak, and A. V. Subramanyam, "Lstm guided modulation classification and experimental validation for sub-nyquist rate wideband spectrum sensing," in *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*. IEEE, 2019, pp. 458–460.

[4] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser *et al.*, "Rethinking attention with performers," *arXiv preprint arXiv:2009.14794*, 2020.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[6] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888.

[7] S. Dörner, S. Cammerer, J. Hoydis, and S. Ten Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, 2017.

[8] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 10 335–10 342.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, I. Simon, C. Hawthorne, A. M. Dai, M. D. Hoffman, M. Dinculescu, and D. Eck, "Music transformer," *arXiv preprint arXiv:1809.04281*, 2018.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[12] S. Lerkvaranyu, K. Dejhan, and Y. Miyanaga, "M-qam demodulation in an ofdm system with rbf neural network," in *The 2004 47th Midwest Symposium on Circuits and Systems, 2004. MWSCAS'04.*, vol. 2. IEEE, 2004, pp. II–II.

[13] K. Lu, Z. Wu, and X. Shao, "A survey of non-orthogonal multiple access for 5g," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–5.

[14] W. Lyu, Z. Zhang, C. Jiao, K. Qin, and H. Zhang, "Performance evaluation of channel decoding with deep neural networks," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[15] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad hoc networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[16] S. C. Mukhopadhyay and N. K. Suryadevara, "Internet of things: Challenges and opportunities," in *Internet of Things*. Springer, 2014, pp. 1–17.

[17] A. Nafaa, T. Taleb, and L. Murphy, "Forward error correction strategies for media streaming over wireless networks," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 72–79, 2008.

[18] K. Ohnishi and K. Nakayama, "A neural demodulator for quadrature amplitude modulation signals," in *Proceedings of International Conference on Neural Networks (ICNN'96)*, vol. 4. IEEE, 1996, pp. 1933–1938.

[19] M. Önder, A. Akan, and H. Doğan, "Neural network based receiver design for software defined radio over unknown channels," in *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)*. Ieee, 2013, pp. 297–300.

[20] T. O'shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[21] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[22] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.

[23] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[24] B. Shevitski, Y. Watkins, N. Man, and M. Girard, "Digital signal processing using deep neural networks," *arXiv preprint arXiv:2109.10404*, 2021.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[26] H. Wang, Z. Wu, S. Ma, S. Lu, H. Zhang, G. Ding, and S. Li, "Deep learning for signal demodulation in physical layer wireless communications: Prototype platform, open dataset, and analytics," *IEEE Access*, vol. 7, pp. 30 792–30 801, 2019.

[27] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2017, pp. 1–6.

[28] J. Zhuang, T. Tang, Y. Ding, S. Tatikonda, N. Dvornek, X. Papademetris, and J. S. Duncan, "Adabelief optimizer: Adapting stepsizes by the belief in observed gradients," *arXiv preprint arXiv:2010.07468*, 2020.

Back to Contents

OMASGAN: Out-of-distribution Minimum Anomaly Score GAN for Anomaly Detection

Nikolaos Dionelis
Electronics and Electrical Engineering
The University of Edinburgh, UK
Contact email: nikolaos.dionelis@ed.ac.uk

Sotirios A. Tsaftaris
Electronics and Electrical Engineering
The University of Edinburgh
Edinburgh, UK

Mehrdad Yaghoobi
Electronics and Electrical Engineering
The University of Edinburgh
Edinburgh, UK

***Abstract*—Generative models trained in an unsupervised manner may set high likelihood and low reconstruction loss to Out-of-Distribution (OoD) samples. This leads to failures to detect anomalies, overall decreasing Anomaly Detection (AD) performance. In addition, AD models underperform due to the rarity of anomalies. To address these limitations, we develop the OoD Minimum Anomaly Score GAN (OMASGAN) which performs retraining by including the proposed minimum-anomaly-score OoD samples. These OoD samples are generated on the boundary of the support of the normal class data distribution in a proposed self-supervised learning manner. Our OMASGAN retraining algorithm leads to more accurate estimation of the underlying data distribution including multimodal supports and also disconnected modes. For inference, for AD, we devise a discriminator which is trained with negative and positive samples either generated (negative or positive) or real (only positive). The evaluation of OMASGAN on image data using the leave-one-out method shows that it achieves an improvement of at least 0.24 and 0.07 points in AUROC on average on the MNIST and CIFAR-10 datasets, respectively, over other benchmark models for AD.**

***Index Terms*—Out-of-Distribution (OoD) detection, Anomaly detection, Generative Adversarial Networks (GAN)**

I. Introduction

In spite of progress in Anomaly Detection (AD), models, including Generative Adversarial Networks (GAN) [1], learn to assign high probability to the seen data, but are not trained to assign zero probability to Out-of-Distribution (OoD) samples [2], [3]. During inference, anomalies might still be assigned non-zero probability, leading to failures to detect anomalies. To address such limitations, we propose the OoD Minimum Anomaly Score GAN (OMASGAN) which performs retraining by including our proposed new minimum-anomaly-score OoD samples. These OoD samples are generated on the boundary of the support of the normal class distribution in a proposed self-supervised learning manner. Our OMASGAN retraining algorithm leads to more accurate estimation of the underlying distribution including multimodal supports with disconnected modes. For inference, for AD with negative sampling and training [4], [5], we devise a discriminator which is trained with negative and positive samples either generated (negative or positive) or real (only positive). Our contributions are:

- We propose OMASGAN to more accurately (a) learn the underlying data distribution for improved AD, and (b) discern between the true and generated in-distribution and the self-generated (near boundary) and provided negative samples.
- We perform model retraining by including samples on the boundary of the support of the normal class data distribution. To address the rarity of anomalies, we generate abnormal samples using data samples only from the normal class.
- We train a discriminator to separate the data distribution from its complement and use it as an inference mechanism for AD. The evaluation of OMASGAN using the Leave-One-Out (LOO) methodology shows that it achieves good performance in the Area Under the Receiver Operating Characteristics curve (AUROC) and outperforms benchmarks. Our OMASGAN is also evaluated using One-Class-Classification (OCC), outperforming GAN- and AE-based benchmark models.

OMASGAN performs retraining by including the learned OoD samples generated on the boundary of the support of the normal class data distribution, and not randomly somewhere in the data space, to more accurately learn the underlying data distribution. We address the generators knowing *what* they do not know problem [2], [3] in an optimal manner, as the OoD samples are as close as possible to the data distribution, i.e. tightest-possible data description. All other methods allow for slack space. Both [6] and [7] create OoD data samples in an ad hoc way using a single-epoch generator and blurry low-quality reconstructions. The pseudo-anomaly mixup module [6] leads to a limiting definition of anomaly. [6] and [7] use a restrictive definition of anomaly, as they create OoD samples that are not well-scattered, not covering the OoD part of the data space.

We improve upon the work in [8] on invertible models by using negative retraining and extending the methodology to a greater class of models. The gain and novelty of OMASGAN is its technical simplification, but more importantly the extension of the methodology to a variety of *metrics*, i.e. f-divergence distribution metrics, without using likelihood or invertibility.

II. Our Proposed OMASGAN Algorithm

Flowchart. Figure 1 presents our OMASGAN model: We train a f-divergence GAN to obtain the generator, $G(\mathbf{z})$, where $\mathbf{z}$ is a latent variable. We denote the latent space by $\mathscr{Z} \in \mathbb{R}^l$. The GAN samples, $G(\mathbf{z})$, and the data, $\mathbf{x}$, lie in the data space, $\chi \in \mathbb{R}^k$, where $l < k$. We train a boundary data generator, $B(\mathbf{z})$, to obtain boundary samples to be used as negatives, for active negative sampling. We compute the divergence between B and G, and we then find the boundary of G. OMASGAN generates samples corresponding to a generalized notion of the boundary

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

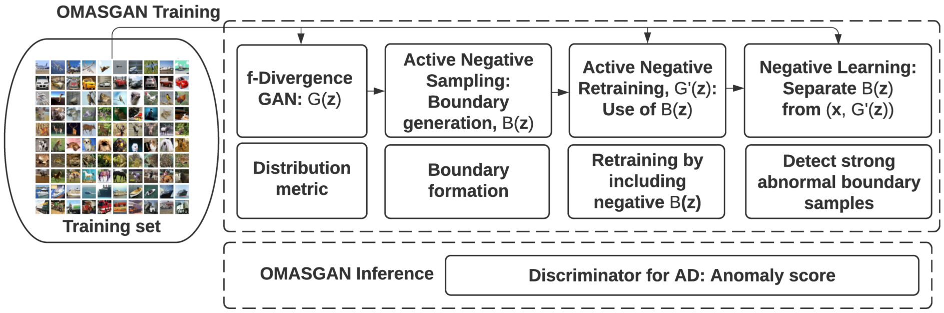

OMASGAN Training
Training set
f-Divergence GAN: G(z)
Active Negative Sampling: Boundary generation, B(z)
Active Negative Retraining, G'(z): Use of B(z)
Negative Learning: Separate B(z) from (x, G'(z))
Distribution metric
Boundary formation
Retraining by including negative B(z)
Detect strong abnormal boundary samples
OMASGAN Inference
Discriminator for AD: Anomaly score

Fig. 1: Flowchart of OMASGAN which generates minimum-anomaly-score OoD samples and then uses them for retraining.

of the support of the data distribution, which is the set of data points such that they are OoD and have a minimum anomaly score, measured as the f-divergence. OMASGAN generates OoD data samples, and we incorporate these OoD Minimum Anomaly Score (OMAS) samples in our algorithm. We perform negative retraining using the OMAS samples and the implicit distributions of G and B, and we train the generator $G'(\mathbf{z})$ using a discriminator, $C(\mathbf{x})$. The samples $G'(\mathbf{z})$ lie in $\mathcal{X}$. For AD, we train the discriminator $J(\mathbf{x})$ using the OMAS samples. During inference, for AD, we compute our proposed anomaly score and detect abnormal samples using J and G'. Specifically:

Establishing a distribution metric. We train a f-divergence GAN to learn the data, $\mathbf{x}$. Using $\mathbf{z} \sim p_{\mathbf{z}}$, $\mathbf{x} \sim p_{\mathbf{x}}$, and $G(\mathbf{z}) \sim p_g$,

$$\arg\min_G \max_D \ \mathbb{E}_{\mathbf{x}} \log(D(\mathbf{x})) + \mathbb{E}_{\mathbf{z}} \log(1 - D(G(\mathbf{z}))). \quad (1)$$

Using conjugate functions, f^*, [9], [10] and for example $D(\mathbf{x}) = 1/(1+\exp(-V_D(\mathbf{x})))$ and $g_f(\mathbf{v}) = -\log(1+\exp(-v))$, the first f-divergence based optimization objective of OMASGAN is

$$\arg\min_G \max_D \ \mathbb{E}_{\mathbf{x}} \, g_f(V_D(\mathbf{x})) \, - \, \mathbb{E}_{\mathbf{z}} \, f^*(g_f(V_D(G(\mathbf{z})))). \quad (2)$$

Formation of the distribution's boundary. To perform active negative sampling, we train the distribution boundary model, $B(\mathbf{z})$, with model parameters $\boldsymbol{\theta_b}$. The optimization is

$$\begin{aligned} \arg\ \min_{\boldsymbol{\theta_b}} & -m(B(\mathbf{z};\boldsymbol{\theta_b}), G(\mathbf{z})) + \mu \ d(B(\mathbf{z};\boldsymbol{\theta_b}), G(\mathbf{z})) \\ & + \nu \ s(B(\mathbf{z};\boldsymbol{\theta_b}), \mathbf{z}), \end{aligned} \quad (3)$$

where $m(B,G)$ is the distribution metric from (1), i.e. any f-divergence in its variational representation expressed in terms of the conjugate function, $f^*(t)$, as in (7) in [9], where t is a variational function taking as input a sample and returning a scalar. The special cases of Kullback–Leibler (KL) and Pearson are $f^*(t) = \exp(t-1)$ and $f^*(t) = 0.25t^2 + t$, respectively. The first term in (3) is a *decreasing function* of a distribution metric. This metric $m(B,G)$ is between the boundary and the data.

OMAS distribution: We generate minimum-anomaly-score OoD samples and perform learned negative data augmentation. We use a f-divergence GAN discriminator for the distribution metric, $m(B,G)$. The *first two terms* in (3) lead the $B(\mathbf{z})$ samples to the boundary of p_g. We use l_p-norm distance and dispersion regularization. We denote the l_2-norm distance between the point $B(\mathbf{z})$ and the set $\mathbf{x}$ by $d(B(\mathbf{z}),\mathbf{x})$. To avoid mode collapse [8], [11] and generate OMAS samples, we use the scattering measure $s(B(\mathbf{z}),\mathbf{z})$. The distance, $d(B(\mathbf{z}),G)$, and $s(B(\mathbf{z}),\mathbf{z})$ are

$$d(B(\mathbf{z}_i), G(\mathbf{z})) \, = \, \min_{j=1,\dots,Q} ||B(\mathbf{z}_i) - G(\mathbf{z}_j)||_2, \quad (4)$$

$$s(B(\mathbf{z}_i), \mathbf{z}_i) \, = \, \frac{1}{N-1} \sum_{j=1,\, j\neq i}^{N} \frac{||\mathbf{z}_i - \mathbf{z}_j||_2}{||B(\mathbf{z}_i) - B(\mathbf{z}_j)||_2}, \quad (5)$$

where N and Q are batch and inference sizes. A variation of the point-set distance, $d(B(\mathbf{z}),G)$, is the Chamfer distance [12]. In (3), μ and ν are hyperparameters. We find a reliable boundary between normal and abnormal data for classification. OMASGAN generates strong and specifically adversarial anomalies. Strong anomalies lie near the boundary, while *adversarial anomalies* are strong anomalies near high-probability data.

Active negative retraining. To address the learning-OoD-samples problem of G [2], [3], we retrain by including the OoD $B(\mathbf{z})$ self-generated on the boundary. Thus, for $G'(\mathbf{z})$,

$$\begin{aligned} \arg\min_{G'} \max_C \ \ & \mathbb{E}_{\mathbf{z}} \log(1 - C(G'(\mathbf{z}))) \, + \, \alpha \, \mathbb{E}_{\mathbf{x}} \log(C(\mathbf{x})) \\ & + \beta \, \mathbb{E}_{\mathbf{z}} \log(1 - C(B(\mathbf{z}))) \, + \, \gamma \mathbb{E}_{\mathbf{z}} \log(C(G(\mathbf{z}))), \end{aligned} \quad (6)$$

where $G'(\mathbf{z}) \sim p_{g'}$ lie in $\mathcal{X}$ and C is a discriminator for f-divergences [6], [13]. To compute distribution metrics in (6), i.e. between B and $(\mathbf{x}, G)$, we use a *weighted sum of f-divergences* [6], [13]. The optimization in (6) comprises four terms, and it outputs the learned mappings $C : \mathcal{X} \rightarrow \mathbb{R}$, and $G' : \mathscr{Z} \rightarrow \mathcal{X}$. The first and fourth terms enforce the generated samples to the data, as in Rumi-GAN [13]. The third term forces the samples away from our strong anomalies, which are near the support boundary of the data distribution and close to *high-probability* data. The discriminator, $C(\mathbf{x})$, is trained to separate B from $(\mathbf{x}, G)$. G' learns the data avoiding the generated OoD $B(\mathbf{z})$.

Detection of strong abnormal boundary samples. *Separation of generated and real normal from generated abnormal:* To address the learning-OoD-samples problem and to perform active negative training, we train the discriminator $J(\mathbf{x})$,

$$\begin{aligned} \arg\ \max_J \ \ & \mathbb{E}_{\mathbf{z}} \log(J(B(\mathbf{z}))) + \delta \, \mathbb{E}_{\mathbf{x}} \log(1 - J(\mathbf{x})) \\ & + (1-\delta) \, \mathbb{E}_{\mathbf{z}} \log(1 - J(G'(\mathbf{z}))). \end{aligned} \quad (7)$$

Inference mechanism. We use the Anomaly Discriminator, J, and the f-divergence distribution metric for AD, [9], [1]. The f-divergence is used for training, and we also use it during *inference*. The GAN discriminator computes f-divergences; for

Back to Contents

the distributions P and R, we write this metric as fD(P, R). We compute fD(G', $\delta_{\mathbf{x}}^{*}$) for a test sample $\mathbf{x}^{*}$ where G' is the learned distribution after retraining and $\delta_{\mathbf{x}}^{*}$ is a Dirac function centered at $\mathbf{x}^{*}$. For *any* $\mathbf{x}^{*} \in \mathcal{X}$, the Anomaly Score (AS) is

$$AS(\mathbf{x}^{*}) = J(\mathbf{x}^{*}) + \lambda \,\mathrm{fD}(G', \delta_{\mathbf{x}}^{*}). \tag{8}$$

The classification decision is the following. $\mathbf{x}^{*}$ is from the normal class if $J(\mathbf{x}^{*}) + \lambda\,\mathrm{fD}(G', \delta_{\mathbf{x}}^{*}) < \tau$, where τ is a threshold, and $\mathbf{x}^{*}$ is abnormal otherwise. By including negative samples, $J(\mathbf{x})$ learns to discriminate between the data distribution and *its complement*. We use $J(\mathbf{x})$ to detect OoD samples, [6].

III. Related Work

OMASGAN addresses the *rarity of anomalies* and provides negative data augmentation by creating strong OoD data on the distribution boundary, unlike [5], [14]. Our method performs sampling of negative data points and generates optimal points for negative training *closest to the data*, $\mathbf{x}$, in contrast to [4] which needs to have a process that programmatically creates the OoD examples using image transformations. OMASGAN *learns to generate* OoD samples. We perform retraining using active negative sampling, setting the boundary points as *strong anomalies*. This differs from creating OoD samples by using (i) low-epoch blurry reconstructions [6], [7], (ii) rotated features [4], and (iii) a CVAE [15]. Old is Gold (OGNet) creates weak anomalies far from the boundary, low-quality reconstructions, and pseudo-anomalies generated in an *ad hoc manner* without any guarantee of coverage of the OoD part of the data space. OGNet uses a pseudo-anomaly module to create OoD points. It uses a *restrictive definition of anomaly* as single-epoch blurry reconstructions. It also changes the discriminator (f-divergence distribution metric) by using an Autoencoder (AE). Anomalies that are far from the boundary are also created by [15].

Minimum Likelihood GAN (MinLGAN) and FenceGAN generate boundary samples to subsequently use the discriminator score for AD [16], [17], while our OMASGAN performs active negative retraining. In contrast to the Boundary of Distribution Support Generator (BDSG) [8], OMASGAN performs retraining of the normal class distribution by including negative samples self-generated on the boundary, uses any f-divergence distribution metric, no invertibility, and a discriminator for AD, [6]. Our self-supervised learning methodology involves model retraining by including the learned distribution boundary.

The *rarity of anomalies* is not addressed by [18] and [13], as they do not perform learned negative data augmentation. Rarity is addressed by GEOM, GOAD, and Deep Robust One Class Classification (DROCC) [19]. Here, GEOM trains a multi-class AD model to discern between geometric transformations, horizontal flipping, translations, and rotations. It learns feature detectors that identify anomalies based on the model's softmax statistics. The classification-based model GOAD generalizes transformation methods using affine and geometric transformations. In contrast, our OMASGAN does not use such data augmentation techniques. DeepSVDD minimizes the volume of a hypersphere to enclose the data using a deep kernel-based AD loss. However, DeepSVDD suffers from *representation collapse*. Representations richer than a hypersphere are needed. DROCC is robust to representation collapse and assumes that the data lie on a locally linear well-sampled low-dimensional manifold, but does not find the data distribution boundary.

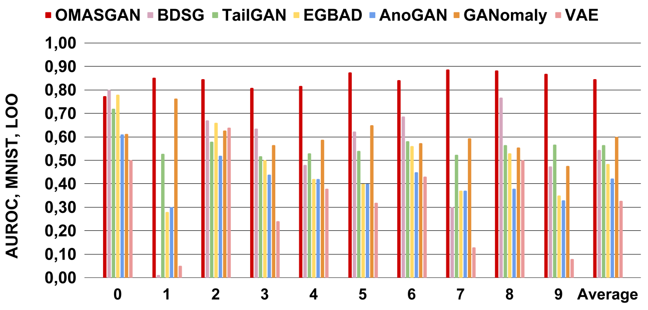


Fig. 2: Performance of OMASGAN on MNIST data in AUROC compared to GAN and AE baselines using LOO evaluation.

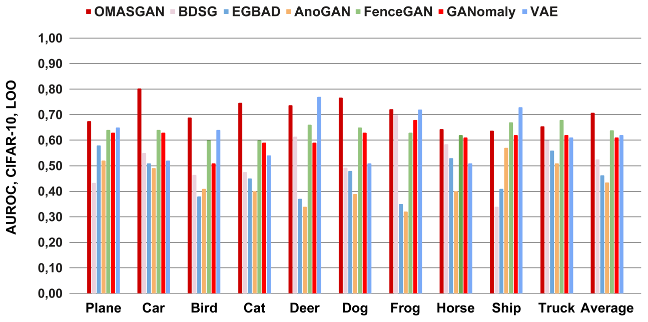


Fig. 3: Performance of OMASGAN in AUROC on CIFAR-10 data compared to GAN and AE baselines using LOO evaluation.

IV. Evaluation of OMASGAN

We evaluate OMASGAN using AUROC. The LOO evaluation is used; we set K classes of a dataset with $(K+1)$ classes as normal and the *leave-out class* as abnormal. LOO evaluation is more challenging than One Class Classification (OCC) used by MinLGAN, OGNet, and [20]–[22] which is setting a single class of a dataset as normal and all the remaining classes of the dataset as abnormal. LOO evaluation is *more realistic* and also closer to typical real-world scenarios than OCC [23].

Models. We train Convolutional Neural Networks (CNN) with batch normalization model architecture, and we use the f-divergence-based KL-Wasserstein GAN (KLWGAN) [10].

Baselines. We evaluate OMASGAN using LOO on MNIST and compare it to the GANs EGBAD [24] and AnoGAN [25], to the likelihood models BDSG and TailGAN [8], [26], and to the AEs GANomaly and VAE. We evaluate OMASGAN using LOO on CIFAR-10, and we compare it to EGBAD, AnoGAN, BDSG, GANomaly, and FenceGAN [16]. We compare OMASGAN to GOAD, GEOM, and DROCC using OCC on CIFAR-10.

Evaluation of OMASGAN on MNIST. *Setup:* We evaluate our f-divergence-based OMASGAN using LOO. We use $p_{\mathbf{z}} = \mathrm{N}_{128}(\mathbf{0}, \mathbf{1})$, $Q = 1024$, $N = 256$, $\mu = 0.2$, and $\nu = 0.25$. We train

Back to Contents

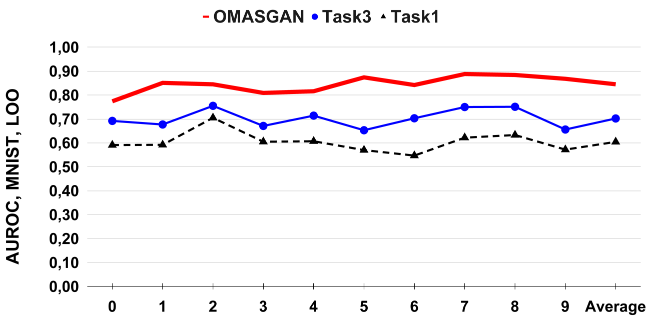


Fig. 4: Ablation study analysis of OMASGAN in AUROC on MNIST image data using the LOO evaluation methodology.

OMASGAN and generate $G(\mathbf{z})$, $B(\mathbf{z})$, and $G'(\mathbf{z})$. We evaluate OMASGAN and its discriminator $J(\mathbf{x})$ using histogram plots of the anomaly scores for normal and abnormal samples.

LOO Evaluation in AUROC: Figure 2 shows that on average and for all the MNIST digits, OMASGAN outperforms the GAN benchmarks EGBAD, AnoGAN, BDSG, and TailGAN. Also, Figure 2 shows that OMASGAN outperforms the AE-based models GANomaly and VAE. We evaluate OMASGAN and compare it with GANomaly when using the *same* inference conditions as those in OMASGAN: training set statistics rather than *test set* statistics for the batch normalization layers.

OMASGAN achieves on average an AUROC of 0.85 on MNIST and outperforms the benchmarks by at least 0.24 points in AUROC, by a percentage of 41%. Our OMASGAN model is robust and achieves the lowest standard deviation (SD) averaged over all the MNIST digits, i.e. 0.036, compared to EGBAD, AnoGAN, BDSG, TailGAN, GANomaly, and VAE. These AD benchmarks have SDs when averaged over all the MNIST digits 0.153, 0.093, 0.24, 0.059, 0.074, and 0.199, respectively.

Evaluation of OMASGAN on CIFAR-10. *Setup:* In (6) and (7), we use $\alpha + \gamma = 0.7$, $\beta = 0.7$, and $\delta = 0.5$, as in [6], [13]. *LOO Evaluation in AUROC:* Figure 3 shows that the performance of OMASGAN using LOO is better than that of the GAN models AnoGAN, EGBAD, FenceGAN, and BDSG both on average and for all the examined LOO AD tasks.

On average and for almost all classes, the proposed OMASGAN outperforms the AE AD benchmarks GANomaly, VAE, ADAE, and AED. OMASGAN *outperforms* the benchmarks in AUROC, averaged over all the classes. It is robust achieving the lowest SD, 0.056, compared to the AD benchmarks. It outperforms the benchmarks on average over all tasks by at least 0.07 AUROC points, by a percentage increase of at least 11%. OMASGAN, on average, achieves an AUROC of 0.71 (AUPRC 0.68) on CIFAR-10 data using LOO evaluation.

Ablation study of OMASGAN on MNIST. Figure 4 shows that, on average and for all the MNIST digits, OMASGAN improves the performance of the KLWGAN model implemented in (1) and (2) (i.e. Task 1) for OoD detection. Comparing the training objective in (1) to the loss in (6) (i.e. Task 3) and to the final loss in (7), OMASGAN improves the performance of the base model. The base model KLWGAN achieves an AUROC of 0.59 averaged over all the MNIST digits, increasing to 0.71 using the loss in (6), and then to AUROC 0.84 using our final OMASGAN model (AUPRC 0.82), and this is the contribution of the proposed negative training methodology.

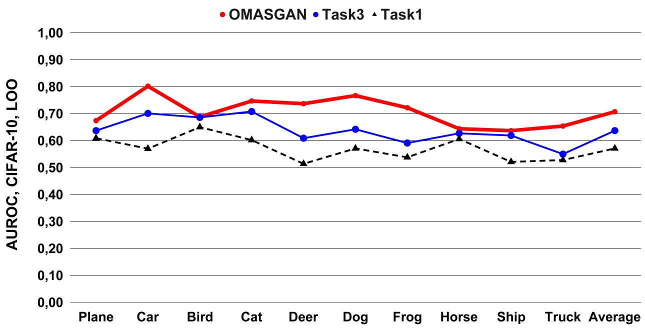


Fig. 5: Ablation study of OMASGAN in AUROC on CIFAR-10 data, and the impact of our losses, using LOO evaluation.

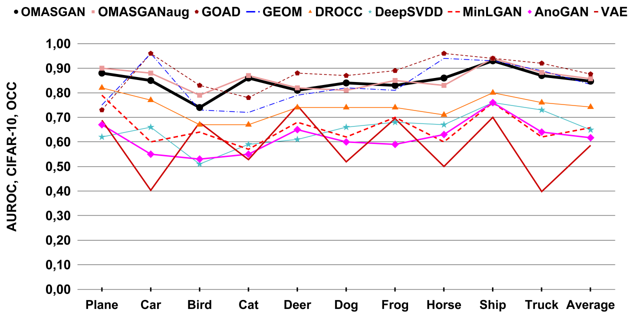


Fig. 6: Performance of OMASGAN in AUROC on CIFAR-10 data using OCC evaluation. Comparison to AD baselines.

Ablation study of OMASGAN on CIFAR-10. Figure 5 presents the ablation study on the losses of OMASGAN on CIFAR-10 in AUROC, using the LOO evaluation methodology. Our chosen base model, KLWGAN in (1), yields an AUROC of 0.57 averaged over all the LOO classes, and this increases to 0.64 using OMASGAN in (6) and to 0.71 using OMASGAN. The improvement of 0.14 points in AUROC is the contribution of our *retraining and negative data augmentation methodology*. The average SDs over all the AD LOO classes are 0.05, 0.05, and 0.06 for Task 1, Task 3, and OMASGAN, respectively.

Evaluation of OMASGAN on CIFAR-10 using OCC. In Figure 6, we evaluate OMASGAN, and we compare it to GAN and AE models, as well as to GOAD and GEOM. OMASGAN achieves an average AUROC of 0.85 (and AUPRC 0.88). It outperforms the GAN models AnoGAN and MinLGAN, as well as the AE AD models VAE and DeepSVDD. OMASGAN also *outperforms* the discriminator-based model DROCC [19]. OMASGAN also achieves robustness across the AD tasks.

OMASGANaug uses data augmentation comprising geometric image transformations, i.e. horizontal flipping and also color augmentation, during training and slightly improves the AD performance of OMASGAN. The performance of OMASGAN and OMASGANaug is comparable to that of the classification

AD model GEOM. GOAD uses data augmentation comprising geometric image transformations, such as flips and rotations, as well as affine transformations, and slightly outperforms OMASGANaug on average in OCC by approximately 2%.

The classification models GEOM and GOAD use features to discriminate between transformations. In contrast, OMASGAN does not use manual processes, human intervention, and feature engineering. This is desirable and strengthens scalability and applicability. It makes no assumptions about the underlying data distribution and is not ad hoc. AD is an automatic outcome of our *negative training*, which can improve upon many of the existing methods to more accurately and robustly learn $p_{\mathbf{x}}$.

V. Discussion and Conclusion

We have proposed OMASGAN, a retraining methodology for AD using active negative sampling and training, and self-supervised learning. OMASGAN performs negative retraining by including the generated boundary which has the effect of "pushing" the distribution away from the OoD samples to improve the learning of the data distribution. The evaluation outcome results on both MNIST and CIFAR-10 data, using the LOO methodology, show that OMASGAN achieves state-of-the-art performance and outperforms the GAN- and AE-based benchmarks, as illustrated in Figures 2 and 3. The ablation study analysis in Figures 4 and 5 shows that OMASGAN improves the base model for AD using LOO evaluation on MNIST and CIFAR-10. Using the AUROC, OMASGAN yields on average (i) an improvement of at least 0.24 points on MNIST over the benchmarks, achieving values of 0.85, and (ii) an improvement of at least 0.07 points on CIFAR-10, achieving values of 0.71 using the LOO methodology. LOO evaluation is more realistic than OCC; in real-world scenarios, we have a large number of items that are *not* rare, and we are interested in detecting objects that are significantly different from these items. Figure 6 shows that OMASGAN outperforms the GAN and AE benchmarks MinLGAN, AnoGAN, VAE, DeepSVDD, and the discriminator-based model DROCC on CIFAR-10 using OCC evaluation, and achieves an average AUROC of 0.85. OMASGAN also achieves a level of robustness across different AD tasks.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing.

References

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative Adversarial Nets," in Proceedings Advances in Neural Information Processing Systems (NeurIPS), p. 2672–2680, 2014.

[2] Nalisnick, E., Matsukawa, A., Teh, Y., Gorur, D., and Lakshminarayanan, B., "Do Deep Generative Models Know What They Don't Know?," in Proceedings Seventh International Conference on Learning Representations (ICLR), New Orleans, USA, 2019.

[3] Kirichenko, P., Izmailov, P., and Wilson, A., "Why Normalizing Flows Fail to Detect Out-of-Distribution Data," in Proceedings NeurIPS, 2020.

[4] Sinha, A., Ayush, K., Song, J., Uzkent, B., Jin, H., and Ermon, S., "Negative Data Augmentation," in Proceedings International Conference on Learning Representations (ICLR), 2021.

[5] Sipple, J., "Interpretable, Multidimensional, Multimodal Anomaly Detection with Negative Sampling for Detection of Device Failure," in Proceedings Thirty-seventh International Conference on Machine Learning (ICML), pp. 9016-9025, vol. 119, 2020.

[6] Zaheer, M., Lee, J., Astrid, M., and Lee, S., "Old is Gold: Redefining the Adversarially Learned One-Class Classifier Training Paradigm," in Proceedings IEEE/CVF Conference Computer Vision and Pattern Recognition (CVPR), 2020.

[7] Pourreza, M., Mohammadi, B., Khaki, M., Bouindour, S., Snoussi, H., and Sabokrou, M., "G2D: Generate to Detect Anomaly," in Proceedings IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 2003-2012, Virtual Conference, 2021.

[8] Dionelis, N., Yaghoobi, M., and Tsaftaris, S., "Boundary of Distribution Support Generator (BDSG): Sample Generation on the Boundary," in Proceedings IEEE International Conference on Image Processing (ICIP), pp. 803-807, October 2020.

[9] Nowozin, S., Cseke, B., and Tomioka, R., "f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization," in Proc. Conference on Neural Information Processing Systems (NeurIPS), 2016.

[10] Song, J. and Ermon, S., "Bridging the Gap Between f-GANs and Wasserstein GANs," in Proceedings International Conference on Machine Learning (ICML), pp. 9078-9087, vol. 119, 2020.

[11] Dionelis, N., Yaghoobi, M., and Tsaftaris, S., "Few-Shot Adaptive Detection of Objects of Concern Using Generative Models with Negative Retraining," in Proceedings International Conference on Tools with Artificial Intelligence (ICTAI), 2021.

[12] Nguyen, T., Pham, Q., Le, T., et al., "Point-set Distances for Learning Representations of 3D Point Clouds," arXiv:2102.04014, 2021.

[13] Asokan, S. and Seelamantula, C., "Teaching a GAN What Not to Learn," in Proceedings 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds), Vancouver, Canada, December 2020.

[14] Sung, Y., Pei, S., Hsieh, S., and Lu, C., "Difference-Seeking GAN - Unseen Sample Generation," in Proceedings Eighth International Conference on Learning Representations (ICLR), 2020.

[15] Bian, J., Hui, X., Sun, S., et al., "A Novel and Efficient CVAE-GAN-Based Approach With Informative Manifold for Semi-Supervised Anomaly Detection," IEEE Access, v. 7, p. 88903-88916, 2019.

[16] Ngo, P., Winarto, A., Li Kou, C., Park, S., Akram, F., and Lee, H., "Fence GAN: Towards Better Anomaly Detection", in Proceedings IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), pp. 141-148, Portland, Oregon, USA, 2019.

[17] Wang, C., Zhang, Y., and Liu, C., "Anomaly Detection via Minimum Likelihood Generative Adversarial Networks," in Proceedings International Conference on Pattern Recognition (ICPR), p. 1121-1126, 2018.

[18] Ruff, L., Vandermeulen, R., Görnitz, N., Binder, A., Müller, E., Müller, K., and Kloft, M., "Deep Semi-Supervised Anomaly Detection," in Proceedings International Conference on Learning Representations (ICLR), Virtual Conference, Formerly Addis Ababa, Ethiopia, 2020.

[19] Goyal, S., Raghunathan, A., Jain, M., Simhadri, H., and Jain, P., "DROCC: Deep Robust One-Class Classification," in Proceedings International Conference on Machine Learning (ICML), 2020.

[20] Kim, Y., et al., "Lipschitz Continuous Autoencoders in Application to Anomaly Detection," in Proc. International Conference on Artificial Intelligence and Statistics (AISTATS), p. 2507-2517, v. 108, 2020.

[21] Nguyen, D., Lou, Z., Klar, M., and Brox, T., "Anomaly Detection with Multiple-Hypotheses Predictions," in Proceedings International Conference on Machine Learning (ICML), p. 4800-4809, v. 97, 2019.

[22] Sohn, K., Li, C., Yoon, J., Jin, M., and Pfister, T., "Learning and Evaluating Representations for Deep One-Class Classification," in Proceedings International Conference on Learning Representations (ICLR), 2021.

[23] Ahmed, F. and Courville, A., "Detecting Semantic Anomalies," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34 (04), pp. 3154-3162, 2020.

[24] Zenati, H., Foo, C., Lecouat, B., Manek, G., and Chandrasekhar, V., "Efficient GAN-Based Anomaly Detection," Workshop Track in Sixth International Conference on Learning Representations (ICLR), 2018.

[25] Schlegl, T., et al., "Unsupervised Anomaly Detection with GANs to Guide Marker Discovery," in Proceedings International Conference on Information Processing in Medical Imaging (IPMI), 2017.

[26] Dionelis, N., Yaghoobi, M., and Tsaftaris, S., "Tail of Distribution GAN (TailGAN): Generative-Adversarial-Network-Based Boundary Formation," in Proceedings Sensor Signal Processing for Defence (SSPD) IEEE Conference, pp. 41-45, November 2020.

Robust DOA Estimation Based on Deep Neural Networks in Presence of Array Phase Errors

Xuyu Gao
College of Underwater Acoustic Engineering
Harbin Engineering University
Harbin, China
1325888546@qq.com

Aifei Liu*
the School of Software
Northwestern Polytechnical University
Xi'an, China
liuaifei@nwpu.edu.cn

Yutao Xiong
the School of Software
Northwestern Polytechnical University
Xi'an, China
2019302924@mail.nwpu.edu.cn

***Abstract*—Deep learning (DL) framework is gradually applied to solve the problem of DOA estimation in array signal processing. DL-based DOA estimation methods are much more efficient than conventional model-based methods in the testing stage. However, the generalization of DL-based methods is limited in the presence of array phase errors, because array phase errors may change in different environments, leading to the difference between the phase errors in the training and the ones in testing. In this paper, we explore the magnitude property of array received signal to develop robust deep neural network (DNN)-based framework for DOA estimation, named as magnitude-based DNN method (shorten as MDNN). The proposed MDNN method performs independently of array phase errors and enjoys a simpler network than the original DNN method. Simulation results in different scenarios demonstrate that the MDNN method behaves much more robust to array phase errors than the original DNN-based method.**

Keywords—Direction of arrival (DOA) estimation ; Deep neural network; Phase-error independence

I. Introduction

[1]Direction-of-Arrival (DOA) estimation in array signal processing is an important topic in wireless communication, radar, sonar and so on [1],[2]. According to the procedure how the DOAs are obtained, most of existing DOA estimation methods can be categorized as parametric methods [3-6], spectral-based methods[7]-[9], and sparse representation (SR) method [10],[11]. In the aforementioned methods, it is assumed that the array manifold is known in prior. However, this assumption is rarely guaranteed in practice. The array errors such as gain and phase errors, sensor position errors, and mutual coupling seriously degrade the performance of most DOA estimation algorithms [12],[13]. Among array errors, array phase errors significantly degrade the performance of most DOA estimation methods. In addition, array phase errors are more difficult to be calibrated to small values due to inherent hardware impairments, especially for extremely high frequency carriers such as millimeter wave [14]. Therefore, in this paper, we investigate robust DOA estimation in the presence of array phase errors.

In order to address robust DOA estimation in the presence of array errors, the parametric methods were developed by taking array errors as array unknown parameters and estimating them, which are classified as two categories. The first category [15],[16] employs the sources with known DOAs to estimate and compensate the array errors in the calibration stage. Afterwards, the unknown DOAs of the wanted source signals are estimated. In most of cases, the first category can calibrate the array well. However, the DOAs of source signals in the calibration stage must be precisely known, which is difficult to be satisfied during operation of the array. Therefore, the second category [17-20] was proposed to self-calibrate the array, which estimates array errors together with the DOAs of source signals. Most of self-calibration methods suffer from the suboptimal convergence in the case of large array errors [17],[18], limited to particular geometries [19], or the increment of spatial spectral searching load because the spatial spectrum becomes two-dimensional [20]. The robust DOA estimation methods in [21] and [22] provides robust DOA estimation in the presence of array errors, with the assumption the statistics of the array model errors are known.

In recent years, the deep learning (DL) framework[23-29] has been applied to solve the problem of DOA estimation, which provides better generalization ability than the conventional machine learning methods such as RBF- and SVR-based DOA estimation methods [30],[31]. Moreover, after the training of the DL network is finished, the calculations in the testing only involve additions and multiplications which are much more efficient than the high-dimensional nonlinear searching in conventional parametric methods, and the eigendecomposition and spectral searching in spectral-based methods. Among the methods in [24-29], the deep neural network(DNN)-based method in [25] and the CNN-based method in [26] investigate the performance of the DL-based DOA estimation in the presence of array errors. It was illustrated that with the array imperfections embedded in the training datasets, the trained DNN network gains the robustness to array imperfections. However, they did not consider the more general case that array imperfections in the testing stage may be different from those in the offline training stage due to the varying environment.

In this paper, we aim to further increase the generalization of the DL-based DOA estimation in the presence of array phase errors. We propose a magnitude-based DNN method (Named as MDNN). In particular, in order to eliminate the effect of array phase errors, we explore the magnitude of the received array signal and construct a phase-error independent signal vector. Afterwards, the phase-error independent signal vector is used to form an input vector of the next DNN of which the output vector is the labeled spatial spectrum. At the end, by searching the peaks of the spatial spectrum, the DOA estimation is accomplished. Due to the phase-error independence, the MDNN method benefits the robustness to array phase errors and a simplified network.

In this paper, the superscripts *, T, and H represent the conjugate, transpose, and conjugate transpose, respectively; E represents expectation operations and j is the imaginary unit.

[1] This work was supported by the Fundamental Research Funds for the Central Universities (Grant No. D5000210641) and the National Natural Science Foundations of China (Grant No. 61701133)

Back to Contents

II. Background

Consider an array with M omni-directional sensors deployed on the same plane and numbered 1 through M, where Sensor 1 is used as reference. Assume that there are K narrow-band, far-field, zero-mean, stationary signals $\left\{s_k\left(t_i\right)\right\}_{k=1}^{K}$ and the DOAs $\left\{\theta_k\right\}_{k=1}^{K}$. In addition, the source signals and the array sensors are assumed in the same plane. The array output can then be expressed as

$$\mathbf{r}\left(t\right)=\sum_{k=1}^{K}\mathbf{a}\left(\theta_k\right)s_k\left(t\right)+\mathbf{n}\left(t\right), \tag{1}$$

where $\mathbf{n}\left(t\right)$ is the zero-mean Gaussian noise vector and it is independent of the source signals, and $\mathbf{a}\left(\theta_k\right)$ is the steering vector of the k-th source, denoted as

$$\mathbf{a}\left(\theta_k\right)=\left[1,e^{-j\frac{2\pi d_{2,k}}{\lambda}},\cdots,e^{-j\frac{2\pi d_{M,k}}{\lambda}}\right]^{\mathrm{T}}, \tag{2}$$

where λ is the center wavelength of source signals, $d_{m,k}$ is the distance from Sensor 1 to Sensor m along θ_k, that is, $d_{m,k}=x_m\cos\theta_k+y_m\sin\theta_k$, x_m and y_m are the values of x axis and y axis of the m-th sensor, respectively. It is noted that since the first sensor is taken as references, we have $x_m=y_m=0$.

Eq. (1) can be written in a compact form as

$$\mathbf{r}\left(t\right)=\mathbf{A}\mathbf{s}\left(t\right)+\mathbf{n}\left(t\right), \tag{3}$$

where $\mathbf{A}$ is an $M\times K$ matrix with $\mathbf{A}=\left[\mathbf{a}\left(\theta_1\right),\mathbf{a}\left(\theta_2\right),\cdots,\mathbf{a}\left(\theta_K\right)\right]$ and $\mathbf{s}\left(t\right)$ is a K-dimensional vector with $\mathbf{s}\left(t\right)=\left[s_1\left(t\right),s_2\left(t\right),\cdots,s_K\left(t\right)\right]^{\mathrm{T}}$.

For simplicity, in the following, we omit the time variable. The covariance matrix of the array output vector can be written as

$$\mathbf{R}=\mathrm{E}\left[\mathbf{r}\mathbf{r}^{\mathrm{H}}\right]=\mathbf{A}\mathbf{R}_s\mathbf{A}^{\mathrm{H}}+\sigma_n^2\mathbf{I}_M, \tag{4}$$

where

$$\mathbf{R}_s=\mathrm{E}\left[\mathbf{s}\mathbf{s}^{\mathrm{H}}\right].$$

Denote the phase error of the m-th sensor as φ_m. As the first sensor is taken as reference, $\varphi_1=0$. In the presence of array phase errors, Eqs.(3) and (4) can then be rewritten as

$$\mathbf{r}=\mathbf{\Phi}\mathbf{A}\mathbf{s}+\mathbf{n} \tag{5}$$

$$\mathbf{R}=\mathbf{\Phi}\mathbf{A}\mathbf{R}_s\mathbf{A}^{\mathrm{H}}\mathbf{\Phi}^{\mathrm{H}}+\sigma_n^2\mathbf{I}_M, \tag{6}$$

where $\mathbf{\Phi}$ are diagonal matrices with their m-th diagonal elements Φ_{mm} equal to $e^{j\varphi_m}$.

In practice, the number of samples is finite. Thus, the covariance matrix $\mathbf{R}$ need be estimated by

$$\widehat{\mathbf{R}}=\frac{1}{N}\sum_{t=1}^{N}\mathbf{r}\left(t\right)\mathbf{r}^{\mathrm{H}}\left(t\right), \tag{7}$$

where N is the number of snapshots, $\hat{\bullet}$ denotes the estimate of the quantity over which it appears.

Thus, the problem addressed here is to develop the DOA estimation method with robustness to array phase errors.

III. Magnitude-based DNN Method

In order to obtain DOA estimation which is robust to array phase errors, we propose a magnitude-based DNN method, shorten as MDNN. The scheme of MDNN method is given in Fig.1, which mainly contains two parts: construction of phase-error independent signal vector and DNN.

A. Construction of phase-error independent signal vector

We construct the element-wise product of the array output vector and its conjugate as

$$\mathbf{r}_{ew}=\mathbf{r}\odot\mathbf{r}^{*}, \tag{8}$$

where $\odot$ denotes element-wise multiplication.

It is noted that $\mathbf{r}_{ew}$ is composed of the square of the element-wise magnitude of the array output vector and thus it eliminates the effect of array phase errors. Based on $\mathbf{r}_{ew}$, we construct a new covariance matrix below

$$\mathbf{R}_{ew}=\mathrm{E}\left[\mathbf{r}_{ew}\mathbf{r}_{ew}^{\mathrm{T}}\right], \tag{9}$$

It is noted that $\mathbf{R}_{ew}$ is independent of array phase errors since it only contains the information of the squared-magnitude of array output vector. Moreover, it is noted that $\mathbf{R}_{ew}$ is real-valued. Thus, $\mathbf{R}_{ew}=\mathbf{R}_{ew}^{*}$ and $\mathbf{R}_{ew}=\mathbf{R}_{ew}^{\mathrm{T}}$. We define $\widehat{\mathbf{R}}_{ew}$ as the estimate of $\mathbf{R}_{ew}$ under limited snapshots. Therefore, we only need to take the off-diagonal upper right elements of the matrix $\widehat{\mathbf{R}}_{ew}$ as a vector $\mathbf{z}$, that is

$$\mathbf{z}=\left[\widehat{\mathrm{R}}_{ew}\left(1,2\right),\cdots,\widehat{\mathrm{R}}_{ew}\left(1,M\right),\cdots,\widehat{\mathrm{R}}_{ew}\left(M-1,M\right)\right] \tag{10}$$

where $\widehat{\mathrm{R}}_{ew}\left(i,l\right)$ represents the i-th row and j-th column element of the matrix $\widehat{\mathbf{R}}_{ew}$.

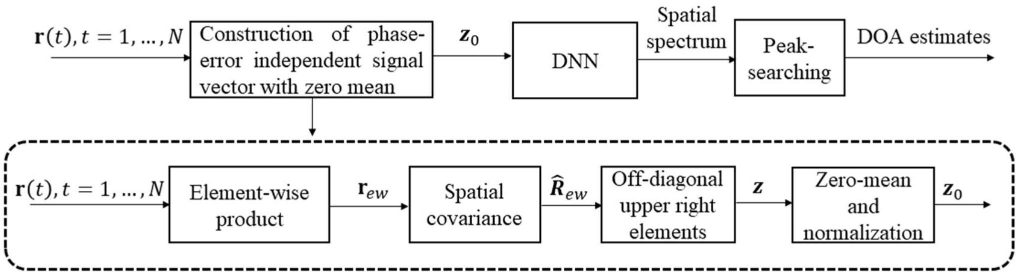

$\mathbf{r}(t), t=1,\ldots,N$
Construction of phase-error independent signal vector with zero mean
$\mathbf{z}_0$
DNN
Spatial spectrum
Peak-searching
DOA estimates
$\mathbf{r}(t), t=1,\ldots,N$
Element-wise product
$\mathbf{r}_{ew}$
Spatial covariance
$\widehat{\mathbf{R}}_{ew}$
Off-diagonal upper right elements
$\mathbf{z}$
Zero-mean and normalization
$\mathbf{z}_0$

Fig. 1 Scheme of MDNN

Back to Contents

We notice the vector $\mathbf{z}$ is always positive and thus its mean is larger than zero. In order to increase the convergence speed of the following deep neural network, we obtain a new vector $\mathbf{z}_0$ by making the vector $\mathbf{z}$ with zero-mean and normalize it below

$$\mathbf{z}_0 = \left(\mathbf{z} - mean(\mathbf{z})\right) / \left\|\mathbf{z} - mean(\mathbf{z})\right\|_2 \qquad (11)$$

where $mean(\mathbf{z})$ stands for taking the mean of the vector $\mathbf{z}$.

B. DNN

After getting the phase-error independent signal vector, we employ the fully connected DNN framework developed in [25] for DOA estimation. The DNN framework is composed of a multitask autoencoder and a series of parallel multilayer classifiers. The autoencoder performs as a group of spatial filters, which helps to reduce the burden of subsequent classifiers. The outputs of the classifiers are concatenated to reconstruct a spatial spectrum for DOA estimation. For clarity, we name the method in [25] as the original DNN method.

In the DNN, the multitask autoencoder is composed of one input layer, one hidden layer, and one output layer. On the other hand, each classifier contains one input layer, two hidden layers, and one output layer. The size of each layer of the MDNN and original DNN is given in Table 1.

In Table 1, IL, HL, and OL stand for input layer, hidden layer, and output layer, respectively; J_0 is the size of the input layer (i.e., the dimension of $\mathbf{z}_0$) in the MDNN method; J_1 is the size of the input layer in the original DNN method. According to Eqs.(8)-(11), we obtain

$$J_0 = \frac{M^2 - M}{2}. \qquad (12)$$

Considering that the original DNN method takes the real- and imaginary-part of off-diagonal upper right matrix elements as input of the DNN, we have

$$J_1 = M^2 - M. \qquad (13)$$

Table 1. Size of each layer of MDNN and original DNN methods

Proposed MDNN (J=J_0); Original DNN (J=J_1)	Multitask autoencoder			Each Classifier			
	IL	HL	OL	IL	HL1	HL2	OL
	J	$\left\lfloor \frac{J}{2} \right\rfloor$	$P \times J$	J	$\left\lfloor \frac{2J}{3} \right\rfloor$	$\left\lfloor \frac{4J}{9} \right\rfloor$	I_0

It is noted that in Table 1, $\lfloor a \rfloor$ represents the largest integer not larger than a ; P is the number of parallel multilayer classifiers, which is equal to the number of parallel classifiers; I_0 is the size of the output layer of each classifier. We assume that the angle-searching range of the targets is $\left[G_{\min}, G_{\max}\right)$ and define the searching grid as η , we then have

$$I_0 = \frac{G_{\max} - G_{\min}}{\eta P}. \qquad (14)$$

Suppose $M = 11$, $G_{\min} = -60^{\circ}$, $G_{\max} = 60^{\circ}$, $\eta = 1^{\circ}$, and $P = 6$, we have $J_0 = 55$, $J_1 = 110$, and $I_0 = 20$ according to Eqs.(12)-(14). In this case, from Table 1, we obtain that the number of neural nodes in the proposed MDNN method is approximately half of that in the original DNN method, which greatly reduces memory requirement and computational load as well.

In addition, considering that in practice, the SNR in the testing stage is commonly unknown, different from the original DNN method, we train the DNN with the data at multiple SNRs together instead of training the DNN at each individual SNR. Afterwards, the trained DNN is applicable to the cases of SNRs from low to high SNRs without knowing the specific SNR in the testing stage.

IV. Simulation Results

This section demonstrates the performance of the proposed MDNN method in different scenarios. The comparisons of the proposed MDNN methods and the original DNN method [25], MUSIC method [9], and the WF method [17] are also provided.

In the simulations, we use an array of M=11 sensors with a configuration same as that in [20]. In addition, we consider $G_{\min} = -60^{\circ}$, $G_{\max} = 60^{\circ}$, $\eta = 1^{\circ}$, and $P = 6$, then we obtain $J_0 = 55$, $J_1 = 110$, and $I_0 = 20$. The MDNN method and original DNN method assign the DNN structure according to Table 1. Moreover, the nonlinear activation involved in the aforementioned two methods is an elementwise hyperbolic tangent function. The Pytorch is used to implement the MDNN network, and the gradients are computed using its embedded tools.

The training dataset consists of two equal-power source signals. The intersignal angle Δ is within the set of $\left\{3^{\circ}, 6^{\circ}, \ldots, 60^{\circ}\right\}$. We assume the DOA of the first source signal θ_1 is sampled with an interval of 1° and $\theta_1 \in \left[-60^{\circ}, 60^{\circ} - \Delta\right)$. Then, we have the DOA of the second source signal $\theta_2 = \theta_1 + \Delta$. In addition, we consider the data at multiple SNRs {0, 5, 10, 20}dB for training the network simultaneously and set the number of snapshots for estimating the covariance matrix as 400 in the training stage. Furthermore, 10 groups of snapshots are collected for each direction setting with randomly generated noise. For one group of snapshots and each SNR, we have $\theta_1 \in \left\{-60^{\circ}, -59^{\circ}, \ldots, 60^{\circ} - \Delta\right\}$ and $\theta_2 = \theta_1 + \Delta$. We have $\theta_1 \in \left\{-60^{\circ}, -59^{\circ}, \ldots, 57^{\circ}\right\}$ which contains 117 elements, when $\Delta = 3^{\circ}$. Therefore, the number of covariance vectors is equal to 117. Similarly, when $\Delta = 6^{\circ}$, the number of covariance vectors becomes 114. As a result, $\left(117 + 114 + \cdots + 60\right) \times 10 \times 4 = 70800$ covariance vectors are collected as the training dataset, where 4 is the number of SNRs and 10 is the number of groups of snapshots. For the MDNN and original DNN methods, when training the spatial filter, the minibatch training strategy is used with a batch size of 32 and learning rate of μ_1 = 0.001, and 1000 epochs are taken for the training with the data set shuffled in each epoch. Afterwards, when training classifier, the minibatch training strategy remain the same expect that 300 epochs are taken for the training.

In practice, array phase errors may not have the specific values as given in [25]. Referring to [17], we consider the more general case for array phase errors, that is, the phase errors $\left\{\varphi_m\right\}_{m=1}^{M}$ of the sensors are randomly generated by

Back to Contents

$$\varphi_m = \sqrt{12}\sigma_\varphi \gamma_m , \tag{15}$$

where γ_m is independent and identically distributed random variables distributed uniformly over [-0.5, 0.5], and σ_φ is the standard deviation of φ_m.

Since array phase errors cannot be known in prior, we set $\sigma_\varphi = 10^\circ$ and train the original DNN network and MDNN network with randomly-generated array phase errors. In testing data, the array phase errors are generated randomly according to Eq.(15) once and then used for 100 times Monte-Carlo simulations to obtain the RSME of the DOA estimates. Note that in the testing stage, the randomly-generated array phase errors are ensured to be different from those in the training dataset.

In addition, considering that the noises and source signals are generated by Gaussian random process, we make sure that the noises and source signals used in the testing stage are different from those used in the training stage. In order to further test the generalization ability of the trained network when the DOAs of targets are off-grid, the testing dataset consists of two equal-power source signals with the DOAs of 10.2° and 50.5°, respectively. The following results are obtained in the testing stage.

A. Spatial Spectrum

We set $\sigma_\varphi = 20^\circ$, SNR = 10dB, and keep other simulation parameters the same as mentioned above. The spatial spectrum of aforementioned methods is given in Fig. 2. From Fig. 2, we can see that the MDNN method has two sharp peaks at the angles very close to the true DOAs of targets. The WF method performs similarly. In contrast, the two largest peaks of the original DNN method and MUSIC method both deviate from the true DOAs of targets obviously. The failure of the original DNN method is caused by the fact that the array phase errors in the testing stage is different from the training stage. In addition, the cause of the poor performance of the MUSIC method is that it was developed with assumption of the absence of array phase errors.

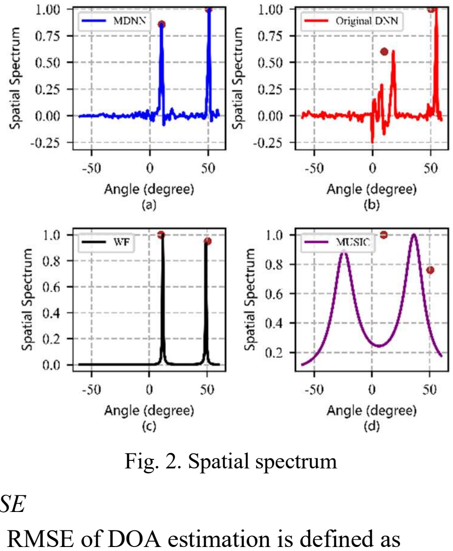


Fig. 2. Spatial spectrum

B. RMSE

The RMSE of DOA estimation is defined as

$$RMSE=\sqrt{\frac{1}{LK}\sum_{l=1}^{L}\sum_{k=1}^{K}(\hat{\theta}_{k,l}\text{-}\theta_k)^2} \tag{16}$$

where $L = 100$ is the number of Monte Carlo simulation experiments; $K = 2$ is the number of source signals. $\hat{\theta}_{k,l}$ denotes the DOA estimation value of the *k*-th source in the *L*-th experiment, and θ_k is the true DOA of the *k*-th source.

In the following, we investigate the RMSE of DOA estimation versus different variables.

1) Phase errors

In the testing, we remain the other parameters and change the phase-error standard deviation σ_φ. We verify the performance of the MDNN, original DNN, WF, and MUSIC methods versus different σ_φ, which is given in Fig. 3. From Fig. 3, we can see that the MDNN method performs regardless of array phase errors as its input of the network is independent of array phase errors. In contrast, the MUSIC method significantly degrades in the presence of array phase errors. The WF and original DNN methods have certain robustness to array phase errors. However, they fail in large phase errors.

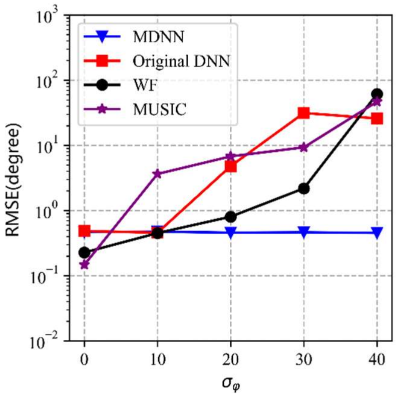


Fig. 3. RMSE versus σ_φ of array phase errors

2) SNRs

In the testing, we set $\sigma_\varphi = 20^\circ$, vary SNRs, and remain the other parameters. We provide the performance of the MDNN, original DNN, WF, and MUSIC methods versus SNRs, as shown in Fig. 4. From Fig. 4, it is illustrated that the performance of all aforementioned methods is getting better as the SNR increase. Among them, the MDNN performs the best. In addition, the MUSIC and original DNN methods have an estimation accuracy almost unchanged when the SNR is larger than -5dB, because of array phase errors.

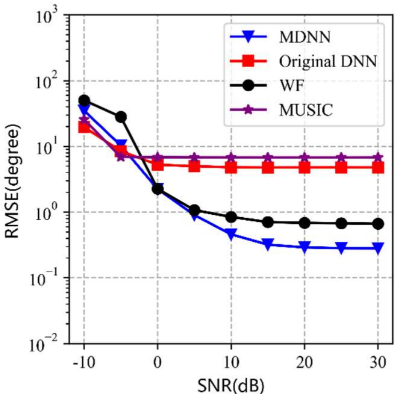


Fig. 4. RMSE versus SNR

3) Number of snapshots

Back to Contents

In the testing, we set $\sigma_{\varphi} = 20^{\circ}$, SNR = 10dB, vary the number of snapshots, and remain the other parameters. We provide the performance of the MDNN, original DNN, WF, and MUSIC methods versus the number of snapshots, as given in Fig. 5. From Fig. 5, it is illustrated that the performance of MDNN and WF methods behave better as the increment of the number of snapshots. However, the MUSIC and original DNN methods remain the same performance when the number of snapshots becomes larger, due to array phase errors.

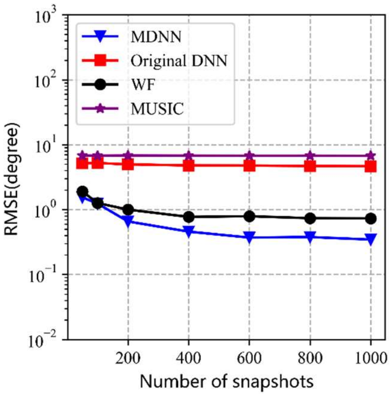


Fig. 5. RMSE versus number of snapshots

V. Conclusion

In this paper, the problem of DOA estimation in the presence of array phase errors is addressed. The proposed MDNN method explores the magnitude of array data to remove array phase errors from the input of the following DNN network. As a result, the MDNN method is independent of array phase errors. Furthermore, the network input vector of the MDNN method is half-length of that in the original DNN method, and thus reduces the number of neural nodes to nearly half of that in the original DNN method. This implies that the MDNN method can significantly reduce memory requirement and computational load. Simulation results under different scenarios illustrate that the MDNN method is always more robust than the DNN method in the presence of array phase errors. In addition, it is demonstrated that the MDNN method has good generalization in different number of snapshots and SNRs when the DOAs of targets are off-grid.

References

[1] D. H. Johnson and D. E. Dudgeon, Array Signal Processing: Concepts and Techniques. New Y ork, NY , USA: Simon & Schuster, 1992.

[2] H. Krim and M. Viberg, “Two decades of array signal processing research: The parametric approach,” IEEE Signal Process. Mag., vol. 13, no. 4, pp. 67–94, Jul. 1996.

[3] I. Ziskind and M. Wax, “Maximum likelihood localization of multiple sources by alternating projection,” IEEE Trans. Acoust. Speech Signal Process. vol. 36, no.10, pp.1553–1560, 1988.

[4] P. Stoica and K.C. Sharman, “Maximum likelihood methods for direction of arrival estimation,” IEEE Trans. Acoust. Speech Signal Process. vol. 38, no.7, pp.1132–1143, 1990.

[5] B. Ottersten, M. Viberg and T. Kailath, “Analysis of subspace fitting and ML techniques for parameter estimation from sensor array data,” IEEE Trans. Signal Process., vol. 40, no. 3, pp. 590-600, Mar. 1992.

[6] R. Roy and T. Kailath, “ESPRIT-estimation of signal parameters via rotational invariance techniques,” IEEE Trans. Acoust., Speech Signal Process., vol. 37, no. 7, pp. 984-995, July 1989.

[7] M. Bartlett, “Smoothing periodograms from time-series with continuous spectra,” Nature , vol. 161, pp. 686–687, 1948.

[8] J. Capon, “High-resolution frequency-wavenumber spectrum analysis,” in Proc. IEEE Proc. IRE*. vol. 57, no. 8, pp. 1408-1418, Aug. 1968.

[9] R. Schmidt, “Multiple emitter location and signal parameter estimation,” IEEE Trans. Antennas Propag., vol. 34, no. 3, pp. 276-280, March 1986.

[10] D. Malioutov, M. Cetin, and A. S. Willsky, “A sparse signal reconstruction perspective for source localization with sensor arrays,” IEEE Trans. Signal Process, 2005, 53(8):3010-3022.

[11] Z. Yang, J. Li, P. Stoica, and L. Xie, “Sparse Methods for Direction-of-Arrival Estimation,” Signal Processing, vol. 7, pp. 509–581, 2018.

[12] M. Wylie, S. Roy, and H. Messer, “Joint DOA estimation and phase calibration of linear equispaced (LES) arrays,” IEEE Trans. Signal Process., vol. 42, no. 12, pp. 3449–3459, 1994.

[13] A. Ferréol, P. Larzabal, and M. Viberg, “Statistical analysis of the MUSIC algorithm in the presence of modeling errors, taking into account the resolution probability,” IEEE Trans. Signal Process., vol. 58, no. 8, pp. 4156–4166, 2010.

[14] J. Zhang, X. Hu and C. Zhong, “Phase Calibration for Intelligent Reflecting Surfaces Assisted Millimeter Wave Communications,” IEEE Trans. Signal Process., vol. 70, pp. 1026-1040, 2022.

[15] B. C. Ng and C. M. S. See, “Sensor-array calibration using a maximumlikelihood approach,” IEEE Trans. Antennas Propag., vol. 44, no. 6, pp.827–835, 1996.

[16] N. Fistas and A. Manikas, “A new general global array calibration method,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 1994, vol. 4, pp. 73–76.

[17] A. J. Weiss and B. Friedlander, “Eigenstructure methods for direction finding with sensor gain and phase uncertainties,” Circuits Syst. Signal Process., vol. 9, no. 3, pp. 271–300, 1990.

[18] A. Paulraj and T. Kailath, “Direction of arrival estimation by eigenstructure methods with unknown sensor gain and phase,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 1985, pp. 640–643.

[19] Y. Li and M. H. Er, “Theoretical analyses of gain and phase error calibration with optimal implementation for linear equispaced array,” IEEE Trans. Signal Process., vol. 54, no. 2, pp. 712–723, 2006.

[20] A. Liu, G. Liao, C. Zeng, Z. Yang and Q. Xu, “An Eigenstructure Method for Estimating DOA and Sensor Gain-Phase Errors,” IEEE Trans. Signal Process., vol. 59, no. 12, pp. 5944-5956, Dec. 2011.

[21] J. Li, P. Stoica, and Z. Wang, “On robust Capon beamforming and diagonal loading,” IEEE Trans. Signal Process., vol. 51, no. 7, pp. 1702–1715, 2003.

[22] B. Wahlberg, B. Ottersten, and M. Viberg, “Robust signal parameter estimation in the presence of array perturbations,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 1991, pp.3277–3280.

[23] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, pp.436–444, 2015.

[24] Y. Kase, T. Nishimura, T. Ohgane, Y. Ogawa, D. Kitayama and Y. Kishiyama, “DOA Estimation of Two Targets with Deep Learning,” Workshop on Positioning, Navigation and Communications, 2018, pp. 1-5.

[25] Z. Liu, C. Zhang and P. S. Yu, “Direction-of-Arrival Estimation Based on Deep Neural Networks With Robustness to Array Imperfections,” IEEE Trans. Antennas Propag., vol. 66, no. 12, pp. 7315-7327, 2018.

[26] G. K. Papageorgiou, M. Sellathurai and Y. C. Eldar, “Deep Networks for Direction-of-Arrival Estimation in Low SNR,” IEEE Trans. Signal Process., vol. 69, pp. 3714-3729, 2021.

[27] A. M. Elbir, “DeepMUSIC: Multiple Signal Classification via Deep Learning,” IEEE Sens. Lett., vol. 4, no. 4, pp. 1-4, April 2020.

[28] L. Wu, Z. Liu and Z. Huang, “Deep Convolution Network for Direction of Arrival Estimation With Sparse Prior,” IEEE Signal Process. Lett., vol. 26, no. 11, pp. 1688-1692, Nov. 2019.

[29] G. K. Papageorgiou and M. Sellathurai, “Direction-of-Arrival Estimation in the Low-SNR Regime via a Denoising Autoencoder,” 2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2020, pp. 1-5.

[30] H. L. Southall, J. A. Simmers, and T. H. Donnell, “Direction finding in phased arrays with a neural network beamformer,” IEEE Trans. Antennas Propag., vol. 43, no. 12, pp. 1369–1374, 1995.

[31] M. Pastorino and A. Randazzo, “A smart antenna system for direction of arrival estimation based on a support vector regression,” IEEE Trans. Antennas Propag., vol. 53, no. 7, pp. 2161–2168, 2005.

A Polynomial Subspace Projection Approach for the Detection of Weak Voice Activity

Vincent W. Neo[†], Stephan Weiss[*], Patrick A. Naylor[†]
[†]Department of Electrical and Electronic Engineering, Imperial College London, UK
[*]Department of Electronic and Electrical Engineering, University of Strathclyde, Scotland
{vincent.neo09, p.naylor}@imperial.ac.uk, stephan.weiss@strath.ac.uk

***Abstract*—A voice activity detection (VAD) algorithm identifies whether or not time frames contain speech. It is essential for many military and commercial speech processing applications, including speech enhancement, speech coding, speaker identification, and automatic speech recognition. In this work, we adopt earlier work on detecting weak transient signals and propose a polynomial subspace projection pre-processor to improve an existing VAD algorithm. The proposed multi-channel pre-processor projects the microphone signals onto a lower dimensional subspace which attempts to remove the interferer components and thus eases the detection of the speech target. Compared to applying the same VAD to the microphone signal, the proposed approach almost always improves the F1 and balanced accuracy scores even in adverse environments, e.g. -30 dB SIR, which may be typical of operations involving noisy machinery and signal jamming scenarios.**

***Index Terms*—Voice activity detection, polynomial matrix eigenvalue decomposition, multi-channel signal processing**

I. Introduction

A voice activity detection (VAD) algorithm identifies whether or not time frames contain speech. VAD is essential for many military and commercial speech processing applications such as speech enhancement [1], speech coding [2], speaker identification [3], [4], and automatic speech recognition (ASR) systems [5]. For example, speech enhancement algorithms may facilitate communication among operators in military operations where the acoustic environment is very challenging, e.g., very noisy machinery and signal jamming scenarios. Such algorithms, however, usually rely on noise estimators, which can be derived from the VAD pre-processing.

Classical statistics-based VAD approaches such as [2], [6]–[8] exploit the statistics of speech and noise. These approaches compute the model parameters based on the assumptions of the speech and noise distributions. However, the performance of these algorithms degrades when the assumed signal statistics are violated and the speech presence probability, which the VAD algorithms usually exploit, is difficult to deduce analytically [9]. Furthermore, during noise-only segments, rapidly changing noise can result in transient interference [10].

Machine learning-based VAD methods have also been proposed to implicitly model the data without using an explicit noisy signal model [10]–[12]. Amongst many, a VAD algorithm, which uses a Gaussian mixture model (GMM) trained in recognizing speech features, has been widely adopted for real-time applications in the WebRTC system [13]. The algorithm cannot cope with noisy environments where it becomes challenging to extract speech features, severely degrading its performance [9], [11].

In [14], a broadband subspace-based approach is used to detect weak transient signals. The approach applies a polynomial matrix eigenvalue decomposition (PEVD), which is iteratively approximated by algorithms such as the second-order sequential best rotation (SBR2) [15], [16] and sequential matrix diagonalization (SMD) [17], [18] in the time-domain or [19], [20] in the discrete Fourier transform (DFT)-domain, to generate the eigenvectors and eigenvalues. Filtering the signal through the eigenvector filterbank yields a syndrome vector, which is more discriminative towards detecting a transient signal [14].

In this work, we adapt [14] and investigate the idea of weak transient signal detection for multi-microphone VAD. The novel contributions of this paper are (i) a subspace-projection approach for VAD instead of the syndrome vector approach used for weak transient signal detection in [14], (ii) the use of realistic speech signals and measured room impulse responses (RIRs) and (iii) a comparison of the proposed approach against benchmark VAD algorithms in adverse environments. We first describe the goal of a VAD algorithm and provide a review of PEVD in Section II. The proposed method based on a multi-channel polynomial subspace projection is presented in Section III. Simulations and results are discussed in Section IV and Section V concludes our findings.

II. Problem Formulation

A. Signal Model

The received signal at the q-th microphone is

$$x_q(n) = \sum_{p=1}^{P} \mathbf{h}_{p,q}^T(n)\mathbf{s}_p(n) \ , \qquad (1)$$

where $\mathbf{h}_{p,q} = [h_{p,q}(0), \ldots, h_{p,q}(J)]^T$ represents the RIR from the p-th source to the q-th microphone, modelled as a J-th order finite impulse response filter, $\mathbf{s}_p(n) = [s_p(n), \ldots, s_p(n-J)]^T$ is a tap delay line vector formed from the p-th source signal, n is the sample index, and $[\cdot]^T$ is the

The work of S. Weiss was supported by the Engineering and Physical Sciences Research Council (EPSRC) grant no. EP/S000631/1, and the UK MOD University Defence Research Collaboration in Signal Processing. The work of P. A. Naylor was funded through the UK EPSRC grant no. EP/S035842/1.

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

transpose operator. The data vector over Q microphones is $\mathbf{x}(n) = [x_1(n), \ldots, x_Q(n)]^T$.

Since the P source signals are not simultaneously excited all the time, the goal of a VAD algorithm is to identify time segments when the p-th source is active.

B. Polynomial Matrix Eigenvalue Decomposition

The space-time covariance matrix, parameterized by time lag $\tau \in \mathbb{Z}$, is computed using [21]

$$\mathbf{R}(\tau) = \mathbb{E}\{\mathbf{x}(n)\mathbf{x}^T(n-\tau)\} \; , \tag{2}$$

where $\mathbb{E}\{\cdot\}$ is the expectation operator over n. Each element, $r_{p,q}(\tau)$, is the correlation sequence between the p-th and q-th microphone signals. This produces auto- and cross-correlation sequences on the diagonals and off-diagonals, respectively.

The z-transform of (2),

$$\boldsymbol{\mathcal{R}}(z) = \sum_{\tau=-\infty}^{\infty} \mathbf{R}(\tau) z^{-\tau} \; , \tag{3}$$

denoted by $\mathbf{R}(\tau) \;\circ\!\!-\!\!\bullet\; \boldsymbol{\mathcal{R}}(z)$, is a para-Hermitian polynomial matrix satisfying $\boldsymbol{\mathcal{R}}(z) = \boldsymbol{\mathcal{R}}^P(z) = \boldsymbol{\mathcal{R}}^H(1/z^*)$, where $[\cdot]^*$, $[\cdot]^H$, $[\cdot]^P$ are the complex conjugate, Hermitian and para-Hermitian operators respectively. The para-Hermitian eigenvalue decomposition (EVD) of (3) is [21], [22]

$$\boldsymbol{\mathcal{R}}(z) = \boldsymbol{\mathcal{U}}(z)\, \boldsymbol{\Lambda}(z) \boldsymbol{\mathcal{U}}^P(z) \; , \tag{4}$$

where the columns of $\boldsymbol{\mathcal{U}}(z)$ are the polynomial eigenvectors and the elements on the diagonal matrix $\boldsymbol{\Lambda}(z)$ are the polynomial eigenvalues. Iterative PEVD algorithms based on the SBR2 [15], [16] and SMD [18], [23] are used to approximate (4) by Laurent polynomial factors.

Exploiting the orthogonality between subspaces and assuming L signal components, (4) can be partitioned into

$$\boldsymbol{\mathcal{R}}(z) = \begin{bmatrix} \boldsymbol{\mathcal{U}}_s(z) & \boldsymbol{\mathcal{U}}_\perp(z) \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_s(z) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_{\bar{s}}(z) \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{U}}_s^P(z) \\ \boldsymbol{\mathcal{U}}_\perp^P(z) \end{bmatrix} , \tag{5}$$

where $\mathbf{0}$ is a zero matrix, $\boldsymbol{\Lambda}_s : \mathbb{C} \to \mathbb{C}^{L \times L}$ contains the L principal eigenvalues of the signal-related components with its eigenvectors on the columns of $\boldsymbol{\mathcal{U}}_s(z) : \mathbb{C} \to \mathbb{C}^{Q \times L}$ while the eigenvalues $\boldsymbol{\Lambda}_{\bar{s}} : \mathbb{C} \to \mathbb{C}^{(Q-L)\times(Q-L)}$ defines the noise floor along with the orthogonal complement or noise-only subspace spanned by the columns of $\boldsymbol{\mathcal{U}}_\perp(z) : \mathbb{C} \to \mathbb{C}^{Q \times (Q-L)}$.

III. Polynomial Subspace Projection Approach for Voice Activity Detection

A. Polynomial Subspace Projection

Typically, VAD algorithms operate directly on the microphone signals. In the presence of strong interfering signals, however, the performance of these algorithms degrades, as will be investigated in Section IV.

Assuming that the first few frames contain only the interferer components, the space-time covariance matrix in (2) can be estimated without bias using [24], [25]

$$\mathbf{R}(\tau) \approx \frac{1}{N - |\tau|} \sum_{n=0}^{N-1} \mathbf{x}(n)\mathbf{x}^T(n-\tau) \; . \tag{6}$$

Whenever we have assurance that only the stronger interfering signals are present, $\mathbf{R}(\tau)$ can be re-estimated using (6) based on appropriate interference-only segments in $\mathbf{x}(n)$. The PEVD is computed on the z-transform of (6) to generate the orthogonal complement subspace $\boldsymbol{\mathcal{U}}_\perp(z)$ based on (5).

In [14], a syndrome vector is obtained by filtering the microphone signals through the eigenvector $\boldsymbol{\mathcal{U}}_\perp(z) \;\bullet\!\!-\!\!\circ\; \mathbf{U}_\perp(n)$. This syndrome vector is used to detect the entry of a new target source that may be weaker in power than the L interferers, assumed to be stationary for a period of time. The syndrome energy increases in the presence of a new source which is likely to protrude into the subspace $\boldsymbol{\mathcal{U}}_\perp(z)$. Furthermore, since $\mathbf{U}_\perp(n)$ is designed to be causal [26] and may introduce bulk delays to the microphone signals for signal alignment, the syndrome vector may no longer be temporally aligned with the microphone signals. Hence, the syndrome vector cannot be directly used to generate a VAD mask for the microphone signals.

Instead of generating a syndrome vector in [14], a polynomial subspace projection $\boldsymbol{\mathcal{P}}(z) = \boldsymbol{\mathcal{U}}_\perp(z)\boldsymbol{\mathcal{U}}_\perp^P(z) \in \mathbb{C}^{Q \times Q}$ is performed on the microphone signals $\mathbf{x}(n)$ to project them onto a reduced $(Q-L)$ dimensional subspace. This will generate time signals $\mathbf{y}(n)$ with a reduction in energy contributions of the estimated L interferer components using

$$\mathbf{y}(n) = \sum_k \sum_m \mathbf{U}_\perp(k)\, \mathbf{U}_\perp^H(k-m)\, \mathbf{x}(n-m) \; . \tag{7}$$

Note that L is the estimated rank of the interferer components. In general, because of errors incurred in estimating (2) and because PEVD algorithms such as SBR2 and SMD encourage spectral majorization of the extracted eigenvalues, leakage occurs across the subspace, i.e., some signal components leak into $\boldsymbol{\mathcal{U}}_\perp(z)$ [27]. More notably, in the context of dereverberation [28], the direct-path and early reflections are captured by the subspace associated with the first principal eigenvalue while the late reverberant components are observed in the other subspaces [29]. While an over-estimation of L may be advantageous in minimizing the energy spread of the interferer components, the projection of the target signal onto a lower $(Q-L)$ dimensional subspace may not yield significant components in $\mathbf{y}(n)$.

B. Voice Activity Detection on Projected Component

In order to detect a change point due to an emerging target speaker in the syndrome vector, a VAD algorithm [2] can be applied to the q-th processed signal $y_q(n)$ to generate a more reliable binary mask $m_q(n)$ than the microphone signal $x_q(n)$ which contains some interferer components. The segments containing the target source are then extracted using

$$\hat{s}_q(n) = m_q(n) \cdot y_q(n) \; , \tag{8}$$

where $\hat{s}(n)$ is the estimated target speech in the q-th processed signal, and $m_q(n)$ takes on the value 0 or 1 since it is binary. The proposed method is summarized in Algorithm 1.

Algorithm 1 Polynomial Subspace Projection-Based VAD.

```
Inputs: $\mathbf{x}(n) \in \mathbb{R}^Q, L$.
    $\mathbf{R}(\tau) \leftarrow E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$ // interferer-only frames, see (2)
    $\boldsymbol{\mathcal{R}}(z) \leftarrow \mathcal{Z}\{\mathbf{R}(\tau)\}$ // see (3)
    $\boldsymbol{\mathcal{U}}(z), \boldsymbol{\Lambda}(z) \leftarrow$ PEVD $\{\boldsymbol{\mathcal{R}}(z)\}$ // use SMD [18]
    $\mathbf{y}(n) \leftarrow$ project$\{\mathbf{U}_\perp(n), \mathbf{x}(n)\}$ // see (7)
    $m_q(n) \leftarrow$ VAD$\{y_q(n)\}$ // apply VAD [2] on q-th signal
    $\hat{s}_q(n) \leftarrow m_q(n) \cdot x_q(n)$ // extract target activity, see (8)
    return $\hat{s}_q(n)$.
```

IV. Simulation and Results

A. Setup

Measurements of the speech signals and $Q = 8$ channel cafeteria RIRs were taken from the VCTK corpus [30] and Kayser database [31], respectively. The interferer signals comprising F16 cockpit and destroyer engine room noise were extracted from the Noisex database [32]. If necessary, signals were resampled to match the sampling rates of 48 kHz. The speech and interferer signals were separately convolved with the RIRs before being added together at each microphone. The source-to-interferences ratio (SIR) [33] at the first microphone, taken to be the reference, was varied from -30 dB to 20 dB. The target speaker and directional interferer are respectively 1.02 m in front (along the y-axis) and 1.62 m to the right (along the x-axis) of the listener, at positions A and D in Fig. 1 [31].

The VAD algorithms used include Sohn's approach [2] and the approach used by WebRTC [13]. WebRTC operates at modes 0–3 from the least to the most aggressive setting. The microphone signals were processed in 30 ms frames. The first 15 frames were assumed to contain only the interferer signals and were therefore, used for calculating (5). We also applied [2] to the projected signal $\mathbf{y}(n)$ to investigate if there is any advantage of pre-processing with (7) using different rank estimates, $L = 1, 2, 5, 7$ (R1, R2, R3, R7).

B. Ground Truth Labels

A similar procedure described in [34] is used to establish the ground truth (GT) labels. The RIR from the target to the first microphone, chosen as the reference, is truncated approximately 5 ms after the direct-path peak. The truncation is necessary to ensure that the target speech is time aligned with the microphone signals while minimizing reverberation. The anechoic target speech signal is then convolved with the truncated RIR to generate the target speech in $x_1(n)$. The VAD algorithm Mode 3 [13] is applied to the target signal to generate the ground truth VAD labels as shown in Fig. 2(a). For the short target speech used later in Experiment 2 shown in Fig. 2(a)(ii), the positive label '1' at approximately 2.8 s corresponds to a bilabial sound made with both lips [5], as also observed in informal listening examples [35]. In this paper, results for only the first microphone are presented.

C. Evaluation Measures

The counts for the ground truth and predicted labels are tabulated using a confusion matrix [36]. The absence or presence of speech is indicated by the label '0' or '1'. True positive (TP) and true negative (TN) are obtained when both labels are '1' and '0' respectively. False negative (FN) occurs when the predicted label is '0' but the ground truth is '1' while false positive (FP) happens when the predicted label is '1' but the ground truth is '0'. This allows the use of F1, true positive rate (TPR), true negative rate (TNR), and balanced accuracy (BACC) scores defined as [36]

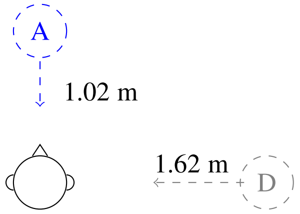


Fig. 1: Experiment setup in the cafeteria from [31].

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FN}},$$
$$\text{F1} = \frac{\text{TP}}{\text{TP} + 0.5 \times (\text{FP} + \text{FN})}, \text{BACC} = \frac{\text{TPR} + \text{TNR}}{2}. \quad (9)$$

D. Results and Discussions

1) Experiment 1: Comparison of VAD on Destroyer Noise. The results are summarized in Table I. At 20 dB SIR, G0 and G3 outperform the other approaches. Slight improvement in F1 and BACC scores arising from an increase in TP is observed when we apply Sohn to the signals projected onto the lower-dimensional subspace (R1, R2, R3, R7) over the microphone signal.

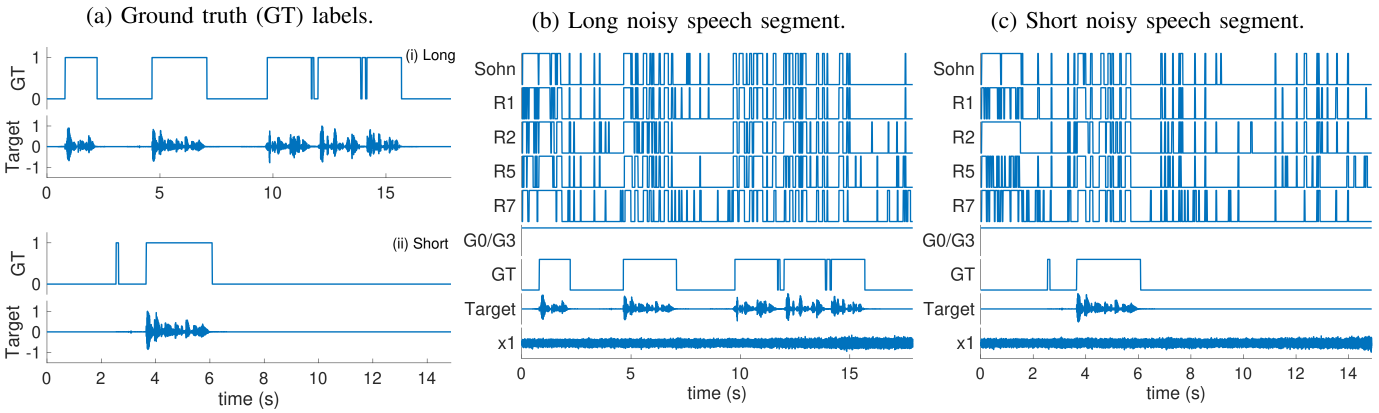
As shown in Table I(b) at 10 dB SIR, the VAD outputs of G0 and G3 are consistently 1, resulting in very high TP and FP. This gives a F1 score of 0.866 but poor BACC score of 0.500 arising from zero negative labels. The proposed approach to perform Sohn [2] on the projected signals shows a slight improvement in F1 score over direct processing on the microphone signal.

At -30 dB SIR where the target signal is significantly weaker, Table I(c) highlights the more significant improvement in the proposed approach over the baseline Sohn. The subspace projection approach increases TP by up to 57 for R7, although this was traded against a drop in TN by 12.

At a high SIR of 20 dB, subspace leakage into the orthogonal complement subspace from the interferer-only subspace is less likely. Hence, R1, R2, R5 and R7 performed similarly. However, at low SIR, e.g. -30 dB, the interferer-only subspace is likely to have leaked into the complement subspace. This promotes high-rank, e.g. R7, so that the microphone signal can be projected into a 1-dimensional subspace where interferer-only components are mostly removed. Note that this small dimensional subspace projection will likely contain only a fraction of the target signal, and hence, the selection of the rank L represents a trade-off.

2) Experiment 2: Different Target Speech Durations. The target speech is corrupted by -20 dB SIR directional F16 cockpit noise. The VAD outputs are shown in Fig. 2(b) for the

Back to Contents


(a) Ground truth (GT) labels.

(b) Long noisy speech segment.

(c) Short noisy speech segment.

Fig. 2: Comparison of VAD binary outputs $m_1(n)$ using Sohn VAD [2] on the microphone signal $x_1(n)$ (Sohn), proposed approach by applying [2] on projected signal $y_1(n)$ using different estimated ranks (R1-R7), WebRTC using modes 0 and 3 (G0, G3) [13]. The plots show (a) the ground truth (GT) labels for (i) long and (ii) short target signal component in $x_1(n)$; (b) long noisy and (c) short noisy segments of speech corrupted by -20 dB SIR F16 cockpit noise from Noisex database [32].

TABLE I: Confusion matrix and scores for VAD output on target speech in directional destroyer noise at various SIR.

(a) SIR = 20 dB

Method	TP	TN	FP	FN	F1	BACC
Sohn	283	175	104	32	0.806	0.763
R1	287	174	105	28	0.812	0.767
R2	286	175	104	29	0.811	0.768
R5	287	173	106	28	0.811	0.766
R7	291	171	108	24	0.815	0.768
G0	311	249	30	4	0.948	0.940
G3	293	273	6	22	**0.954**	**0.954**

(b) SIR = 10 dB

Method	TP	TN	FP	FN	F1	BACC
Sohn	271	238	41	44	0.864	**0.857**
R1	275	233	46	40	0.865	0.854
R2	275	224	55	40	0.853	0.838
R5	280	227	52	35	**0.866**	0.851
R7	277	231	48	38	**0.866**	0.854
G0	315	0	279	0	0.693	0.500
G3	315	0	279	0	0.693	0.500

(c) SIR= -30 dB

Method	TP	TN	FP	FN	F1	BACC
Sohn	94	226	53	221	0.407	0.628
R1	111	227	52	204	0.464	0.651
R2	102	235	44	213	0.443	0.642
R5	138	226	53	177	0.545	0.688
R7	151	214	65	164	0.569	**0.699**
G0	315	0	279	0	**0.693**	0.500
G3	315	0	279	0	**0.693**	0.500

TABLE II: Confusion matrix and scores for VAD output on long target speech in directional F16 noise at -20 dB SIR.

Method	TP	TN	FP	FN	F1	BACC
Sohn	130	241	38	185	0.538	0.638
R1	136	249	30	179	0.565	0.662
R2	158	244	35	157	0.622	**0.688**
R5	148	247	32	167	0.598	0.678
R7	136	224	55	179	0.538	0.617
G0	315	0	279	0	**0.693**	0.500
G3	315	0	279	0	**0.693**	0.500

TABLE III: Confusion matrix and scores for VAD output on short target speech in directional F16 noise at -20 dB SIR.

Method	TP	TN	FP	FN	F1	BACC
Sohn	28	334	76	56	0.298	0.574
R1	30	342	68	54	0.330	0.596
R2	45	349	61	39	**0.474**	**0.693**
R5	32	344	66	52	0.352	0.610
R7	32	325	85	52	0.318	0.587
G0	84	0	410	0	0.291	0.500
G3	84	0	410	0	0.291	0.500

same long speech segment as Experiment 1. The target signal and the GT labels are shown along with the other VAD outputs. As described in the earlier experiment, the G3 VAD output is always 1, which implies that it always predicts the presence of speech. This results in a high TP and, subsequently, good F1 score but is penalized by the poor BACC score arising from high FP, as shown in Table II.

When the target speech segment is short, as shown in Fig. 2(c), the G0 and G3 VAD outputs are also always 1. However, this time, the FP tremendously increases to 410 and this severely affects the F1 score. The proposed approach demonstrates that pre-processing the microphone with the subspace projection almost always improves the F1 and BACC scores. In this case, R2 provides an improvement over [2] in F1 and BACC scores by 0.176 and 0.119, respectively.

V. Conclusion

In this work, a polynomial subspace projection approach has been proposed as a pre-processor to improve VAD performance. We have shown that performing this multi-channel pre-processor prior to applying the single-channel Sohn VAD algorithm [2] almost always improves the F1 and balanced accuracy (BACC) scores even in adverse environments, e.g., -30 dB SIR. This improvement over the baseline of applying VAD to the microphone signal is less significant at high SIRs and more significant at low SIRs. Note that it is particularly in the low SIR regime, i.e., for weak speaker signals, where we set out to boost VAD performance. We have also shown that the rank estimate of the interferer-only subspace directly impacts the orthogonal complement subspace used for the projection and, subsequently, the VAD performance. Informal listening examples are available [35]. An end-to-end PEVD-based VAD algorithm has also been proposed recently [37].

Back to Contents

REFERENCES

[1] S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2230–2244, Sep. 2002.

[2] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, 1999.

[3] M. W. Mak and H. B. Yu, "A study of voice activity detection techniques for NIST speaker recognition evaluation," *Comput. Speech and Language*, vol. 28, no. 1, pp. 295–313, 2014.

[4] S. W. McKnight, A. O. T. Hogg, V. W. Neo, and P. A. Naylor, "A study of salient modulation domain features for speaker identification," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Dec. 2021, pp. 705–712.

[5] L. Rabiner and B. H. Juang, *Fundamentals of speech recognition*, ser. Signal Processing Series. New Jersey: Prentice Hall, 1993.

[6] ITU-T, "Recommendation G.729 Annex B: a silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," Int. Telecommun. Union (ITU-T), Recommendation, Jun. 2012.

[7] S. Gazor and W. Zhang, "A soft voice activity detector based on a Laplacian–Gaussian model," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 498–505, Sep. 2003.

[8] I. Cohen, "Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 5, pp. 466–475, Sep. 2003.

[9] J. H. Ko, J. Fromm, M. Philipose, I. Tashev, and S. Zarar, "Limiting numerical precision of neural networks to achieve real-time voice activity detection," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2018, pp. 2236–2240.

[10] A. Ivry, B. Berdugo, and I. Cohen, "Voice activity detection for transient noisy environment based on diffusion nets," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 2, pp. 254–264, May 2019.

[11] Z.-L. Zhang and D. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 2, pp. 252–264, Feb. 2016.

[12] J. W. Shin, J.-H. Chang, and N. S. Kim, "Voice activity detection based on statistical and machine learning approaches," *Comput. Speech and Language*, vol. 24, no. 2010, pp. 515–530, Mar. 2009.

[13] Google, "WebRTC Voice Activity Detector," 2021. [Online]. Available: https://github.com/wiseman/py-webrtcvad

[14] S. Weiss, C. Delaosa, J. Matthews, I. K. Proudler, and B. A. Jackson, "Detection of weak transient signals using a broadband subspace approach," in *Sensor Signal Process. for Defence Conf. (SSPD)*, Sep. 2021.

[15] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD algorithm for para-hermitian polynomial matrices," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 2158–2169, May 2007.

[16] Z. Wang, J. G. McWhirter, J. Corr, and S. Weiss, "Multiple shift second order sequential best rotation algorithm for polynomial matrix EVD," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2015, pp. 844–848.

[17] V. W. Neo, C. Evers, and P. A. Naylor, "Speech enhancement using polynomial eigenvalue decomposition," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2019, pp. 125–129.

[18] S. Redif, S. Weiss, and J. G. McWhirter, "Sequential matrix diagonalisation algorithms for polynomial EVD of parahermitian matrices," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 81–89, Jan. 2015.

[19] F. K. Coutts, K. Thompson, I. K. Proudler, and S. Weiss, "An iterative DFT-based approach to the polynomial matrix eigenvalue decomposition," in *Proc. Asilomar Conf. on Signals, Syst. & Comput.*, 2018, pp. 1011–1015.

[20] S. Weiss, I. K. Proudler, and F. K. Coutts, "Eigenvalue decomposition of a parahermitian matrix: extraction of analytic eigenvalues," *IEEE Trans. Signal Process.*, vol. 69, pp. 722–737, 2021.

[21] S. Weiss, J. Pestana, and I. K. Proudler, "On the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2659–2672, May 2018.

[22] S. Weiss, J. Pestana, I. K. Proudler, and F. K. Coutts, "Corrections to "On the Existence and Uniqueness of the Eigenvalue Decomposition of a Parahermitian Matrix"," *IEEE Trans. Signal Process.*, vol. 66, no. 23, pp. 6325–6327, Dec. 2018.

[23] V. W. Neo and P. A. Naylor, "Second order sequential best rotation algorithm with Householder transformation for polynomial matrix eigenvalue decomposition," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2019, pp. 8043–8047.

[24] C. Delaosa, J. Pestana, N. J. Goddard, S. Somasundaram, and S. Weiss, "Sample space-time covariance matrix estimation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2019, pp. 8033–8037.

[25] C. Delaosa, J. Pestana, N. J. Goddard, S. Somasundaram, and S. Weiss, "Support estimation of a sample space-time covariance matrix," in *Sensor Signal Process. for Defence Conf. (SSPD)*, 2019.

[26] J. Corr, K. Thompson, S. Weiss, J. G. McWhirter, and I. K. Proudler, "Causality-constrained multiple shift sequential matrix diagonalisation for parahermitian matrices," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2014, pp. 1277–1281.

[27] C. Delaosa, F. K. Coutts, J. Pestana, and S. Weiss, "Impact of space-time covariance estimation errors on a parahermitian matrix EVD," in *Proc. IEEE Sensor Array and Multichannel Signal Process. Workshop (SAM)*, 2018, pp. 164–168.

[28] P. A. Naylor and N. D. Gaubitch, Eds., *Speech dereverberation*. Springer-Verlag, 2010.

[29] V. W. Neo, C. Evers, and P. A. Naylor, "Enhancement of noisy reverberant speech using polynomial matrix eigenvalue decomposition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 3255–3266, Oct. 2021.

[30] C. Veaux, J. Yamagishi, and S. King, "The voice bank corpus design, collection and data analysis of a large regional accent speech database," in *Conf. Asian Spoken Language Research and Evaluation*, Nov. 2013.

[31] H. Kayser, S. D. Ewert, J. Anemüller, T. Rohdenburg, V. Hohmann, and B. Kollmeier, "Database of multichannel in-ear and behind-the-ear head-related and binaural room impulse responses," *EURASIP J. on Advances in Signal Process.*, vol. 2009, no. 1, p. 298605, Jul. 2009.

[32] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition II: NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 3, no. 3, pp. 247–251, Jul. 1993.

[33] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.

[34] S. Braun and I. Tashev, "On training targets for noise-robust voice activity detection," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2021, pp. 421–425.

[35] V. W. Neo, "VAD exploiting a polynomial subspace projection approach," Apr. 2022. [Online]. Available: https://vwn09.github.io/research/pevd-vad

[36] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, Aug. 2018.

[37] V. W. Neo, S. Weiss, S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Polynomial eigenvalue decomposition-based target speaker voice activity detection in the presence of competing talkers," in *Proc. Int. Workshop on Acoust. Signal Enhancement (IWAENC)*, Sep. 2022.

Optimizing sonobuoy placement using multiobjective machine learning

Christopher M Taylor
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, U.K.
Email: christopher.taylor2@liverpool.ac.uk

Simon Maskell
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, U.K.
Email: smaskell@liverpool.ac.uk

Jason F Ralph
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, U.K.
Email: jfralph@liverpool.ac.uk

***Abstract*—We present a new approach to finding optimal patterns for the placement of fields of sonobuoys in a complex undersea environment. The problem is modelled as a biobjective one, where the aim is to both minimize uncertainty over target localization and minimize sensor placement time. We develop a two-phase algorithm, where an offline multiobjective evolutionary phase finds initial Pareto-nondominated solutions to a static problem, and then an online multiobjective reinforcement learning phase finds improved solutions using updated information. Initial results show that our approach generates significant improvements over standard grid patterns.**

I. Introduction

Consider a theatre of operations into which a number of sensors are to be placed for target localization; for example, a field of passive DIFAR sonobuoys in an anti-submarine warfare (ASW) context. Fixed and regular deployment patterns could be selected based on mission objectives, including curved screening patterns [1], surrounding circles or ovals [2], [3], chevrons, and variations on grid patterns [4], but it may not be clear which if any of these are optimal, especially in a complex undersea environment where patterns optimized to current local conditions might perform better.

We consider a scenario in which two possibly conflicting objectives must be satisfied simultaneously: localization of the target of interest (TOI) with minimum uncertainty, and minimization of the time taken to place a pattern of N sensors. In a discretized two-dimensional $l \times w$ lattice of hexagons of equal side length b, there are $\begin{pmatrix} l \cdot w \\ N-1 \end{pmatrix}$ possible valid placement patterns, so for example using 12 sensors and a 100×100 lattice, there are approximately $2{\times}10^{39}$ such patterns. In general, it is impossible to exhaustively calculate the objective values associated with all possible patterns even before factoring in the varying effects of noise and clutter on the localization uncertainty objective. Hence efficient heuristic optimization methods are required that can produce an acceptable set of Pareto-nondominated solutions with respect to the two objectives within reasonable computation time.

Evolutionary algorithms (EAs) are a well-studied and widely used iterative technique to solve high-dimensional, nonlinear combinatorial optimization problems with complex constraints. In [5], an enhanced genetic algorithm (GA) is used for optimal placement of irregular patterns of passive and active-bistatic sonobuoys, and [6] extends the use case to multistatic-active sonobuoy fields using a combination of coherent and incoherent processing. In [7], a multiobjective EA (MOEA) is used to produce a Pareto front (PF) of nondominated solutions for deployment of drifting acoustic sensor networks for cooperative track detection.

EAs can have long run times and do not necessarily cope well with dynamically updating information. Reinforcement learning (RL) algorithms on the other hand can be designed to indicate optimal actions, or sequences of actions, in the presence of dynamic information, and have been applied to sensor management and scheduling problems [8] as well as to tracking in an underwater environment [9]. In recent years, multiobjective reinforcement learning (MORL) has gained popularity in addressing Markov decision processes (MDPs) with more than one objective [10], [11]. However, RL approaches can produce poor results and/or converge very slowly when the state space is very large, as is the case with the sonobuoy placement problem.

Recently, researchers have attempted to combine RL and EA approaches to leverage the strengths and mitigate the weaknesses of each [12], [13]. We extend this hybrid approach to a multiobjective setting in a two stage process, in which an MOEA initially addresses a static offline sensor placement problem and then an online MORL algorithm updates the Pareto front (PF) after updated information is received.

The remainder of this paper is organized as follows. Section II explains our methodology for modelling the constrained biobjective optimization problem. Section III details the two-stage machine learning algorithm. Section IV presents experimental results; concluding remarks follow.

II. Modelling

To represent uncertainty over localization of the TOI, measurements are taken from $3 \cdot r \cdot (r-1)$ possible contact positions in the hexes surrounding the real position (but excluding ground truth), where r is the number of surrounding concentric rings of hexes. The contacts have the same bearing and speed but different current positions. The chosen coordinate system uses complex numbers to express the position of objects in

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

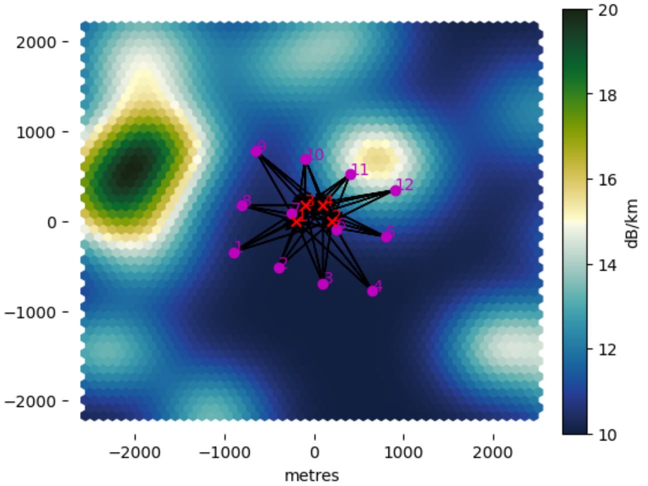


Fig. 1. Sonobuoy and contact locations
Purple circles represent locations of a grid pattern of 12 sonobuoys, numbered by order of placement. Red crosses represent contact locations. Black lines represent the paths for measurement from contact to sensor. The colour intensity on the hexagonal grid indicates the amount of noise and clutter.

the hex lattice, with the origin at the centre of the grid. A complex number presentation of positions is used principally for convenience of calculation; compared to using a vector to represent position in the x, y plane, any array involving these data has at least one dimension less than would otherwise be the case. This makes vectorization of the code more efficient, reducing computation time.

Let there be P different patterns of sensor placement locations to be evaluated. We specify each pattern $k \in \{1, \ldots, P\}$ as a complex-valued offset $p_{k,n},\ n = 1 \ldots N$ to the centroid of the contact positions $\widehat{c_{i,n}},\ i \in \{1, \ldots, C\}$ at the time the n-th sensor is placed; see Figure 1. By expressing sensor locations as offsets in the complex plane, rather than absolute positions, we generate patterns that can be moved and reassessed, so that they may remain valid as the contact locations are perturbed. Sensor n is placed at location $\widehat{\widehat{c_n}} + p_{k,n}$, where:

$$\widehat{\widehat{c_n}} = \frac{1}{C} \sum_{i=1}^{C} \widehat{c_{i,n}}. \tag{1}$$

We wish to simultaneously minimize both total sensor placement time and localization uncertainty. We formulate the biobjective optimization problem as follows:

Minimize

$$\mathbf{\Pi}^* = \arg\min \left[\mathbf{g}\left(\mathbf{\Gamma}\right), \mathbf{h}\left(\mathbf{\Upsilon}\right)\right], \tag{2}$$

subject to:

$$\tau_{i,n} \leq d_{\max}\ \forall i \in \{1 \ldots P\}, n \in \{1 \ldots N\}, \tag{3}$$

$$\tau_{i,n} \geq d_{\min}\ \forall i, n, \tag{4}$$

where:

- $\mathbf{\Pi}^*$ is the optimal policy matrix of non-dominated sensor patterns;
- $\mathbf{\Gamma}$ is the $P \times N - 1$ matrix of times $\tau_n = s_a \cdot d\left(p_{k,n}, p_{k,n-1}\right),\ k = 1 \ldots P,\ n = 2 \ldots N$ taken in seconds between placements at each successive position where P is the number of sampled schedules considered, s_a is the speed of the agent in m/s and $d\left(p_{k,n}, p_{k,n-1}\right)$ is the Euclidean metric;
- $\mathbf{g}\left(\mathbf{\Gamma}\right)$ is a vector-valued function, the output of which is a $P \times 1$ vector, the k-th entry of which gives the total placement times in seconds for pattern p_k;
- $\mathbf{\Upsilon}$ is a $P \times C$ matrix of azimuth measurements for each sensor in each pattern with respect to each contact;
- $\mathbf{h}\left(\mathbf{\Upsilon}\right)$ is a vector-valued function, the output of which is a $P \times 1$ vector, the k-th entry of which represents an aggregate measure of uncertainty over the localization of the TOI for pattern p_k after sensor placement is complete;
- $d_{\max}$ and $d_{\min}$ are maximum and minimum allowable placement distances between sensors, respectively.

For the first objective, the total placement time for sensor pattern k can be expressed as:

$$g_k(\mathbf{\Gamma}) = \widehat{t_{k,1}} + \chi \cdot (N-1) + \sum_{n=2}^{N} \tau_{k,n}, \tag{5}$$

where $\widehat{t_{k,1}}$ is the approach time in seconds, that is, the time taken for the agent (for example, an ASW helicopter) to arrive at the first placement location $\widehat{\widehat{c_1}} + p_{k,1}$ from the base (for example, a static ship) at b_0, and χ is a constant representing time taken in seconds for the placement procedure for each sensor. Placing the base at the origin so that $b_0 = 0 + 0 \cdot I,\ I \equiv \sqrt{-1}$, the distance from the base to the first placement location is $\left|\widehat{\widehat{c_0}} + \widehat{v_c} \cdot \widehat{t_{k,1}} + p_{k,1}\right|$, where $\widehat{\widehat{c_0}}$ is the centroid of the initial contact positions, and $\widehat{v_c}$ is a complex-valued vector representing the movement of the contact :

$$\widehat{v_c} = \widehat{s_c} \cdot \left[\cos\left(\widehat{\theta_c}\right) + I \cdot \sin\left(\widehat{\theta_c}\right)\right], \tag{6}$$

where $\widehat{s_c}$ and $\widehat{\theta_c}$ are the speed and bearing of the contact, respectively. We thus have:

$$\begin{aligned} |v_a| \cdot \widehat{t_{k,1}} &= \left|\widehat{\widehat{c_0}} + p_{k,1} + \widehat{v_c} \cdot \widehat{t_{k,1}}\right|, \\ v_a &= s_a \cdot \left[\cos\left(\theta_a\right) + I \cdot \sin\left(\theta_a\right)\right], \end{aligned} \tag{7}$$

where v_a represents the movement of the agent and θ_a is the bearing of the agent. Squaring both sides of this expression, substituting terms and rearranging, we obtain a quadratic expression in $\widehat{t_{k,1}}$:

$$\begin{aligned} &\left(\widehat{t_{k,1}}\right)^2 \cdot \left(s_c^2 - s_a^2\right) + 2 \cdot \widehat{s_c} \cdot \widehat{t_{k,1}} \cdot \left[\cos\left(\widehat{\theta_c}\right) \cdot \Re\left(\widehat{\widehat{c_0}} + p_{k,1}\right)\right. \\ &\quad \left. + \sin\left(\widehat{\theta_c}\right) \cdot \Im\left(\widehat{\widehat{c_0}} + p_{k,1}\right)\right] + \left|\widehat{\widehat{c_0}}\right|^2 = 0, \end{aligned} \tag{8}$$

which we can solve for the approach time using standard means. Since we can assume $s_a > s_c$ (helicopters are faster than submarines, for example), $\left(s_c^2 - s_a^2\right) \cdot \left|\widehat{\widehat{c_0}}\right|^2 < 0$ and Equation 8 has one positive root.

Back to Contents

For the second objective, we need to approximate the uncertainty from the entire pattern of sensors, which we model as a function of the transmission losses, the noise and clutter on the paths from sensors to contacts, and the position of each pair of sensors with respect to each contact. A noise/clutter map is generated using D bivariate Gaussians with random means and covariance matrices. Under the assumption that detection ranges are comparatively small against likely ocean depth, we model the transmission loss in dB as spherical so that:

$$TL_{k,i,n} = 20 \cdot \log_{10} \left(\left| \widehat{c_{i,N}} - \widehat{\overline{c_N}} - p_{k,n} \right| \right), \tag{9}$$

and measure the noise and clutter level $NC_{k,i,n}$ between sensor n in pattern k and contact i by sampling over over the line $L_{k,i,n}$ and approximating a line integral using Simpson's rule.

To represent localization uncertainty with a pattern of N sensors and with C contacts, we introduce the following measure:

$$h_k(\boldsymbol{\Upsilon}) = \frac{1}{\sum_{n=1}^{N} \beta_{k.n,j}}, \; j \neq n, \tag{10}$$

where $\beta_{k,n,j}$ represents the mean expected information gain from triangulation with respect to a pair of sensors with offsets $p_{k,n}, p_{k,j}, \; j \neq n$ and the contacts, calculated as follows:

$$\beta_{k,n,j} = \frac{1}{C} \cdot \sum_{i=1}^{C} \frac{\left|\sin\left(\arg\left(p_{k,n}\right) - \arg\left(p_{k,j}\right)\right)\right|}{TL_{k,i,n} + TL_{k,i,j} + NC_{k,i,n} + NC_{k,i,j}}. \tag{11}$$

Note that the numerator in Equation 11 approaches 1 as $[\arg(p_{k,n}) - \arg(p_{k,j})] \rightarrow (1 + 2 \cdot k) \cdot \pi/2, \; k \in \mathbb{Z}$ and approaches 0 as $[\arg(p_{k,n}) - \arg(p_{k,j})] \rightarrow k \cdot \pi$, so that information gain is maximized for a given sensor pair as the placement of the pair approaches an orthogonal attitude to a contact and the distance to the contact decreases, whilst information gain approaches zero as the position of a pair lines up with the contact and the distance increases.

III. Machine learning algorithm

The algorithm operates in two phases. In the first phase, an offline MOEA determines a number of nondominated solutions. All valid patterns generated and evaluated at each generation are added to an archive. In the second phase, an online MORL uses and builds on the archive passed from the MOEA, recalculating objective values based on new information and using the archived objective values to obtain approximate biobjective Q-values.

A. MOEA phase

In the first phase, a specialized MOEA is implemented to generate the initial optimal policy matrix $\boldsymbol{\Pi}^*$, as well as an archive of unique assessed patterns. All stages of the MOEA must satisfy Inequalities 3 and 4. The following stages of the MOEA are iterated until a generation limit is reached, or all solutions are nondominated.

1) Initialization: The population consists at each generation of P patterns of N sensor locations $p_{k,n}$. Generated placement patterns are subsequently rotated to match the estimated bearings of the contacts. The patterns are stored in a single array of constant size $P \times N$. Initialization starts from the origin and for each sensor $n \in \{2 \ldots N\}$ generates a random hex within the discretized $l \times w$ hex lattice that excludes all previously generated locations. The sensor is presumed placed at the centre of the hex.

2) Fitness evaluation: Calculation of fitness values, especially for the second objective, is computationally intensive. However, the computational requirement at each generation during the MOEA phase (and each rollout of the MORL) can be substantially reduced by precalculating matrices of all possible values for $\beta_{k.n,j}$ for each possible contact position during the placement process.

3) Elitism: Elite individuals are passed unaltered to the next generation, but are also passed to genetic operators, so that new individuals can be generated that share some or all of the elite individuals' schemata.

4) Tournament selection: We follow [14] by using complete permutations of the population and finding a PF for each tournament, rather than using a ranking, in an efficient batched process that can be parallelized.

5) Mutation: We employ an adaptive mutation rate μ, described in more detail in Section IV. All individuals passed to genetic operators undergo either mutation or crossover. The algorithm chooses a sensor at random from each pattern allocated to mutation, then chooses at random one valid adjacent hex.

6) Crossover: Parent pairs are generated from complete permutations of the available population. Parent patterns are first concatenated twice, with one concatenation having the first parent pattern followed by the second, and the other concatenation the other way round. For each concatenation, duplicate sensor locations are discarded, and provided sufficient candidate sensors remain, two cutpoints are randomly selected and two children produced, one from each concatenation, by joining the sequences before the first and after the second cutpoints.

B. MORL phase

1) Initialization: Once the MOEA has completed, a PF can be presented to an operator who will decide which Pareto-optimal pattern to choose, based on operational preferences over the trade-off between length of placement time and accuracy of localization. To simulate the new information, the location of each contact is perturbed by randomly moving them to adjacent hexes after the placement of the second sensor and made available to the algorithm immediately after the third sensor is placed. A subset of the archive passed from the MOEA is identified that consists of all patterns that match the chosen PF pattern up to the third sensor location, and the uncertainty objective value is updated for each.

2) Learning: As the state space is too large to use an explicit Q-table, we use an approximation function. We consider final returns for each complete rollout, so $R_{n,j,N} = h_j(\boldsymbol{\Upsilon})$ for some pattern $[p_{j,1},\ldots,p_{j,N-1},p_{j,N}]$. Hence for the second objective, for any partial pattern $p = [p_{j,1},\ldots,p_{j,N-k}], k \in \{1, N-3\}$:

$$Q_p\left(S_{p,N-k-1}|S_{p,N-k}\right) = \frac{1}{m}\cdot\sum_{j=1}^{m} Q_j\left(S_{j,N-k}|S_{j,N-k+1}\right)$$
$$= \frac{1}{m}\cdot\sum_{j=1}^{m} h_j(\boldsymbol{\Upsilon}), \tag{12}$$

where m is the number of already assessed patterns for which $[p_{i,1},\ldots p_{i,K-k}] = [p_{j,1},\ldots p_{j,K-k}]$, $i \neq j$, so that Equation 12 becomes a simple average of the sensor uncertainties. Hence, given that the Q-value from first objective can likewise be modelled as a simple average of the total placement times of all the reachable patterns, we can approximate biobjective $Q-$values using means of the objective values of all reachable patterns which have already been assessed. As updated contact location information is received after each sensor placement, objective values for archived patterns must be recalculated.

There are four scenarios when approximated Q-values are calculated:

1) if an initial check against a hyperparameter $\epsilon \in (0,1)$ is made, the algorithm chooses all remaining sensor locations at random;
2) If there are no non-zero Q-values, i.e. no reachable pattern exists in the archive, again a random choice is made;
3) If all Q-values are non-zero, a PF is calculated and the next sensor location is chosen at random from those on the PF;
4) If there is a mix of zero and non-zero Q-values, then depending on the outcome of a second check against ϵ, the algorithm chooses at random amongst either the subset of available sensor locations with zero Q-values or the subset of available sensor locations with non-zero Q-values.

The reason for using this more complicated decision structure rather than the standard ε-greedy approach is that there is, at least initially, a large proportion of next sensor locations with no reachable associated pattern in the archive, so were we to employ the common RL strategy of allowing exploration of all the states with zero Q-values, there will be very little exploitation until the algorithm has run for a large number of episodes. Thus the MORL serves as a form of (partially) greedy local search, extending the more general search carried out initially by the MOEA, as well as being a framework for dynamically updating based on new information.

IV. Experimental results

We first conducted 30 runs with 100 generations in the MOEA phase and 1000 episodes in the MORL phase for $N = [8, 9, \ldots 12]$, with a starting population of 1000 and a different randomly generated noise/clutter map on a 50×50 hex lattice with b =100 m hex side lengths for each run. We set $r = 3$ and so consider contacts in 18 hexes which form one larger hexagonal pattern, excluding the central hex, which is ground truth. Noise/clutter maps are generated with $D = 20$ random Gaussians and values are then linearly rescaled to fall within 10dB/km and 30dB/km. To provide a baseline, since in practice grid patterns are commonly used for such problems, grid patterns with all combinations of spacings in $[5, 6, \ldots, 10]$ hexes were generated, with the orientation and order of sensor placement optimized according to the direction of approach of of the placing agent. These grid patterns were included in the initial population of the MOEA and in the pattern archive passed from the MOEA to the MORL, so that if nondominated, they would be included in the PFs.

We set the tournament size to 2 since binary tournaments have lowest selection pressure, maximizing exploration. The mutation rate is initially set to a low value, μ =0.0001, so that crossover has a chance to find good solutions without destruction of any alleles. However, at each generation in which the PF does not change, the value of μ doubles to a maximum of 0.4096, but falls immediately back to the initial value as soon as the PF changes at any generation. The value of ϵ in the MORL phase is fixed at 0.1.

For the purposes of the experiments, we assumed that the operator would choose the solution with lowest modulus, i.e. with normalized objective values closest to the origin when considered in the complex plane. Because the contacts are perturbed and objective values recalculated after the MOEA phase, there is no guarantee that the nondominated patterns found by the MOEA will have high fitness in the MORL phase. We analyse performance in terms of improvement in the recorded minimum normalized modulus at each generation of the MOEA phase and each episode of the MORL phase, starting with the PF of the initialized population in each phase. Results are shown in Table I.

MOEA improvement after 100 generations over the initial population, which includes the generated optimized grid patterns, peaks when 11 sensor patterns are used, whilst MORL improvement over the pattern archive passed from the MOEA peaks with 9 sensor patterns. It is also instructive to examine the percentage of runs that show any improvement; for the MOEA, this improves monotonically to 100% for 12 sensors, whilst for the MORL, the proportion peaks at 30% for 9 sensors but declines with larger numbers of sensors. This may be in part because of the increased difficulty of finding new nondominated patterns with larger numbers of sensors, but also because the higher success rate of the MOEA for larger numbers of sensors makes it more difficult for the MORL to improve on the results obtained by the MOEA on any run.

To investigate performance with longer runs, we then performed 30 experiments each for $N = 8$ and $N = 10$ using the same noise/clutter maps as before, with an initial population 1000 including grid patterns, 1000 generations of the MOEA so that the archive passed to the MORL approached 1 million patterns, and 5000 episodes of the RL. With $N = 8$, the mean improvement in the minimum modulus objective values over the initial PF including grid patterns for the MOEA phase,

Back to Contents

TABLE I
IMPROVEMENT STATISTICS: 100 MOEA GENERATIONS + 1000 MORL EPISODES

Sensors	MOEA mean	MORL mean	MOEA maximum	MORL maximum	MOEA improvement	MORL improvement
8	3.0%	1.1%	9.4%	11.3%	83.3%	20.0%
9	3.4%	1.3%	9.5%	11.9%	80.0%	30.0%
10	5.1%	0.8%	15.3%	8.8%	90.0%	20.0%
11	5.5%	0.8%	11.5%	6.2%	96.7%	20.0%
12	4.1%	0.3%	7.7%	3.5%	100.0%	16.7%

A summary of improvement statistics for the MOEA and MORL phases respectively, for $N = [8, 9, \ldots 12]$. The first two columns show for each N the mean improvement in the minimum normalized modulus solution across 30 runs for the MOEA and MORL respectively; the third and fourth columns show the maximum improvement for any run; the final two columns show the percentage of runs showing some improvement.

rose from 2.8% for 100 generations to 6.9%, with a maximum improvement for any run of 19.4%, and the percentage of runs showing no improvement fell from 50% to 5.7%. For the MORL phase, the mean improvement in the minimum modulus objective values rose from 0.1% to 7.03% with a maximum improvement of 27.6%, and the percentage of runs showing no improvement fell from 86.7% to 26.7%.

For $N = 10$, results for the MOEA phase showed a 9.1% average improvement, better than the performance with 8 sensors, with a maximum improvement of 14.8%; runs showing no improvement were just 3.3%. However, the MORL phase showed lower improvement of 0.4% average and 5.7% maximum, with 76.7% of runs showing no improvement, almost treble the figure for $N = 8$. The performance of the MORL suggests that the rate of improvement drops steeply with the number of sensors, and that a much greater number of runs is required to deal with the larger state space and dynamic updates; it is notable that this is not necessarily the case for the MOEA, which copes better with path dependency and local minima in the static problem.

V. CONCLUSIONS

The combination of an MOEA and MORL produces promising initial results for the problem of optimally placing sonobuoys in a complex environment with the dual objectives of minimizing placement time and minimizing localization uncertainty. The MOEA conducts a general search in a static environment, and is designed with promotion of diversity in the final pattern archive in mind; this archive is then passed to the MORL which performs local search around a chosen member of the PF from the MOEA, and also dynamically updates based on new information from the sensors.

In evaluation, the algorithm has shown good results with modest numbers of sonobuoys for both the MOEA and MORL phases in terms of improvement in localization and/or speed of placement of sonobuoys over generic grid patterns. For larger numbers of sensors the increased size of the search space may require a more sophisticated value approximation function for the MORL, as well as further measures to improve computational speed and efficiency. An improved approximation function and higher computational efficiency are areas for future research, as are the effects of complexities such as sensor failure, sensor drift, and more complex oceanographic simulation.

ACKNOWLEDGEMENT

The authors would like to thank DSTL Grant no. 1000143726 for financial support.

REFERENCES

[1] S. B. Richter and L. J. Fusillo, "Helicopter Navigation Algorithms for the Placement of Sonobuoys in an Antisubmarine Warfare Environment," in *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference*, Diagnostic/Retrieval Systems, Inc. Oakland, NJ: IEEE, 1988, pp. 280–286.

[2] M. A. Kouritzin, D. J. Ballantyne, H. Kim, and Y. Hu, "On sonobuoy placement for submarine tracking," *Signal Processing, Sensor Fusion, and Target Recognition XIV*, vol. 5809, p. 244, 2005.

[3] M. Karatas, E. Craparo, and G. G. Akman, "Bistatic sonobuoy deployment strategies for detecting stationary and mobile underwater targets," *Naval Research Logistics*, vol. 65, no. 4, pp. 331–346, 2018.

[4] S. Ozols and M. P. Fewell, "On the Design of Multistatic Sonobuoy Fields for Area Search," Defence Science and Technology Organisation, Edinburgh, Australia, Tech. Rep., 2011.

[5] D. R. DelBalzo, D. N. McNeal, and D. P. Kierstead, "Optimized multistatic sonobuoy fields," *Oceans 2005 - Europe*, vol. 2, pp. 1193–1198, 2005.

[6] D. R. DelBalzo and K. C. Stangl, "Design and performance of irregular sonobuoy patterns in complicated environments," in *OCEANS 2009*, 2009, pp. 1–4.

[7] R. Grasso, M. Cococcioni, B. Mourre, J. Osler, and J. Chiggiato, "A decision support system for optimal deployment of sonobuoy networks based on sea current forecasts and multi-objective evolutionary optimization," *Expert Systems with Applications*, vol. 40, no. 10, pp. 3886–3899, 2013.

[8] S. O. Obadan and Z. Wang, "A Multi-agent Approach to POMDPs Using Off-policy Reinforcement Learning and Genetic Algorithms," *International Journal of Computing*, vol. 19, no. 3, pp. 377–386, 2020.

[9] Y. Gong, X. Li, J. Yan, and X. Luo, "Asynchronous Localization with Stratification Effect for Underwater Target: A Reinforcement Learning-based Approach," *ISAS 2019*, pp. 91–96, 2019.

[10] F. Hoffmann, A. Charlish, M. Ritchie, and H. Griffiths, "Sensor path planning using reinforcement learning," in *FUSION 2020*, 2020.

[11] C. Liu, X. Xu, and D. Hu, "Multi-objective Reinforcement Learning: A Comprehensive Overview," *IEEE Transactions on Systems, Man, and Cybernetics:Systems*, vol. 45, no. 3, pp. 385–398, 2015.

[12] A. D. Martinez, J. Del Ser, E. Osaba, and F. Herrera, "Adaptive Multi-factorial Evolutionary Optimization for Multi-task Reinforcement Learning," *IEEE Transactions on Evolutionary Computation*, vol. X, no. X, pp. 1–15, 2021.

[13] I. Tariq, M. A. Sindhu, R. A. Abbasi, A. S. Khattak, O. Maqbool, and G. F. Siddiqui, "Resolving cross-site scripting attacks through genetic algorithm and reinforcement learning," *Expert Systems with Applications*, vol. 168, no. November 2020, p. 114386, 2021.

[14] C. M. Taylor and A. Salhi, "On Partitioning Multivariate Self-Affine Time Series," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 6, pp. 845–862, 2017.

Image Quality SAR Refocus of Moving Targets undergoing Complicated Rolling Maneuvers

David A. Garren
Electrical and Computer Engineering Department
Naval Postgraduate School
Monterey, California, USA
Email: dagarren@nps.edu

***Abstract*—Moving targets typically appear smeared in synthetic aperture radar (SAR) imagery, often making the task of target recognition more difficult. Recently, research has yielded an ability to perform Arbitrary Rigid Object Motion Autofocus (AROMA) to generate automatically focused target images for cases of arbitrary target rotation and translation during the coherent processing interval of SAR collections. The current research analyzes the efficacy of AROMA processing for targets undergoing complicated rolling maneuvers against a background of measured Ku-band image clutter. The results of this analysis reveal that AROMA can yield focused target imagery that correlates well with the true structure of the target.**

***Index Terms*—Synthetic aperture radar, radar imaging, moving target imaging**

I. Introduction

SAR collection systems measured radar data returns over a synthetic aperture to yield focused images of stationary scenes for general weather conditions, day or night. However, often moving targets exhibit smearing to an extent that target recognition is degraded. Many investigations have examined the properties of moving targets within SAR image data, including Raney [1], Barbarossa et al. [2]–[4] DiPietro et al. [5], Jakowatz, Wahl, and Eichel [6], Perry, Dipietro, and Fante [7], and Kirscht [8]. Additionally, moving target detection has been considered by Fienup [9] and Dias and Marques [10]. Also, attempts to refocus such moving targets have been examined as well [11]–[21].

Some investigations have included cases involving target rotation, including Chen and Ling [22], Chen and Martorella [23], Berizzi, Martorella, and Giusti [24]. Additionally, focusing for more general target motion has been examined using methods that track individual target scattering centers, including Weness et al. [25], Werness, Stuff, and Fienup [26], and Carrera, Goodman, Majewski [27]. Also, the investigation of Rigling [28] applies entropy optimization in the refocusing of rotating targets.

Recently, an investigation has yielded an Arbitrary Rigid Object Motion Autofocus (AROMA) capability that automatically generates refocused images of targets that are permitted to have arbitrary target rotation and translation during the SAR synthetic aperture collection time. In effect, AROMA is a three-dimensional (3-D) extension of the standard Phase Gradient Autofocus (PGA) methods of Wahl, Jakowatz, et al. [6], [29]–[31] that are applied to focus stationary scenes. In particular, AROMA applies a 3-D maximum likelihood methodology to estimate the defocus corrections arising from arbitrary target rotation and translation. The use of multidimensional maximum likelihood techniques is presented in several references [32]–[34]. In addition, methods similar to AROMA have been examined in two dimensions for the case of atmospheric bending and delay of radar waveforms [35]–[38]. Furthermore, the input of AROMA is comprised of the usual complex-valued images as input, so that various image formation methods [27], [30], [39]–[43] can be applied in tandem with AROMA.

This analysis considers the use of AROMA refocusing for cases of complicated target roll maneuvers during the SAR collection interval. In particular, known target roll motion is injected into complex SAR image data comprised of measured Ku-band SAR images. This investigation reveals that AROMA yields good target refocus for many cases of target rolling maneuvers by generating focused imager which correlates well with the true target shape.

II. AROMA Processing

The major processing steps of AROMA are given in Figure 1. Multiple iterations can be applied in the overall AROMA processing, such that the refocused target image from a given iteration becomes the input image at the successive iteration. This strategy can be applied by using either a fixed number of iterations or some optimization metric, such as sharpness or entropy.

AROMA applies maximum likelihood methods to yield estimates of the temporal profiles of three unknown phase difference error vectors $\{\Delta\zeta_n, \Delta\mu_n, \Delta\nu_n\}$ that quantify changes from one radar pulse to the next. Then, these phase error vectors are integrated along the radar pulses of the synthetic aperture to yield the required estimates of the phase error vectors $\{\zeta_n, \mu_n, \nu_n\}$. These phase error vectors are applied to generate the refocused image at a particular iteration in the overall AROMA processing. For the subject investigation, 15 iterations were applied to generate the final AROMA refocused target images.

III. Target Roll Maneuvers

AROMA focus quality is evaluated by using synthetic radar data that is generated from the reflection of radar

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

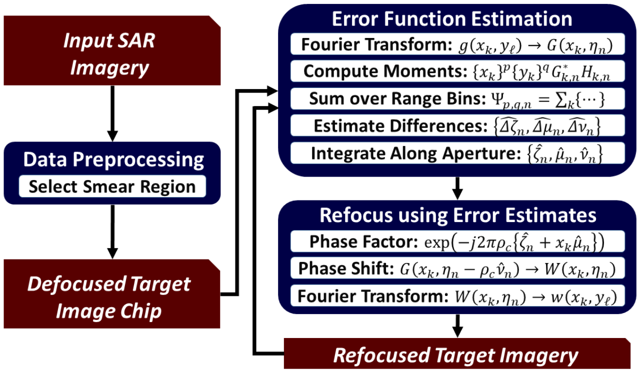

Input SAR Imagery
Data Preprocessing
Select Smear Region
Defocused Target Image Chip
Error Function Estimation
Fourier Transform: $g(x_k, y_\ell) \rightarrow G(x_k, \eta_n)$
Compute Moments: $\{x_k\}^p\{y_k\}^q G^*_{k,n} H_{k,n}$
Sum over Range Bins: $\Psi_{p,q,n} = \sum_k\{\cdots\}$
Estimate Differences: $\{\widehat{\Delta\zeta_n}, \widehat{\Delta\mu_n}, \widehat{\Delta\nu_n}\}$
Integrate Along Aperture: $\{\hat{\zeta}_n, \hat{\mu}_n, \hat{\nu}_n\}$
Refocus using Error Estimates
Phase Factor: $\exp(-j2\pi\rho_c\{\hat{\zeta}_n + x_k\hat{\mu}_n\})$
Phase Shift: $G(x_k, \eta_n - \rho_c\hat{\nu}_n) \rightarrow W(x_k, \eta_n)$
Fourier Transform: $W(x_k, \eta_n) \rightarrow w(x_k, y_\ell)$
Refocused Target Imagery

Fig. 1. A functional block diagram of AROMA [44].

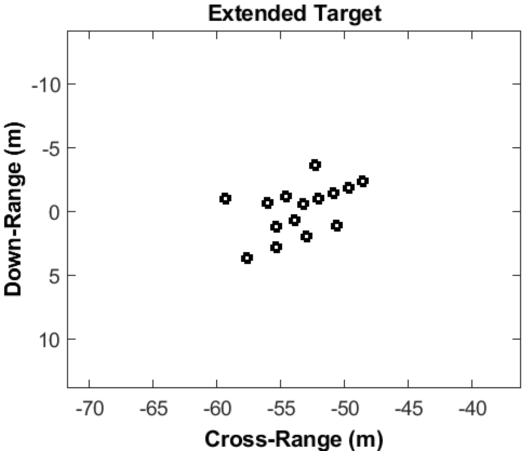
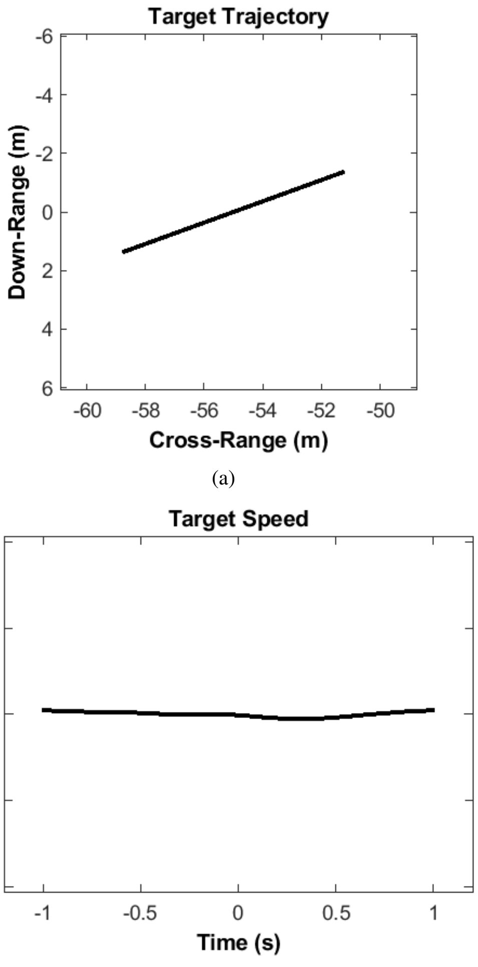


Fig. 2. The individual scattering centers corresponding to the emulated synthetic target has the outline of a mobile tank target.

waveforms from point scattering centers in combination with a background of Ku-band SAR imagery [45]. In generating the synthetic radar data, the same radar collection parameters as the measured Ku-band data are used. These SAR data were collected using a broadside imaging geometry wherein the radar main beam is orthogonal to the radar velocity vector and the radar travels at constant altitude, speed, and heading. For these data, the speed of the radar is $V_0 = 71.3763\,\mathrm{m/s}$, and the altitude of the radar is $Z_0 = 1.496\,\mathrm{km}$. Additionally, the mean value of the ground-range distance between the midpoint of the radar ground track and the imagery scene center is $X_0 = 2.914\,\mathrm{km}$. Furthermore, the coherent time interval of the SAR collection is $T_0 = 2.017\,\mathrm{sec}$. In these data, the radar bandwidth is given by $829.6\,\mathrm{MHz}$, and the radar center frequency is $f_c = 16.8\,\mathrm{GHz}$.

The individual scattering centers corresponding to the emulated synthetic target has the outline of a mobile tank target, as given in Figure 2. The various graphs showing the true motion of the injected synthetic target are given in Figures 3 and 4. In particular, Figure 3(a) presents the true target trajectory projected into the ground plane. Also, Figures 3(b), 4(a), and 4(b) show the true speed, heading, and roll of the injected synthetic target, respectively.



(a)

(b)

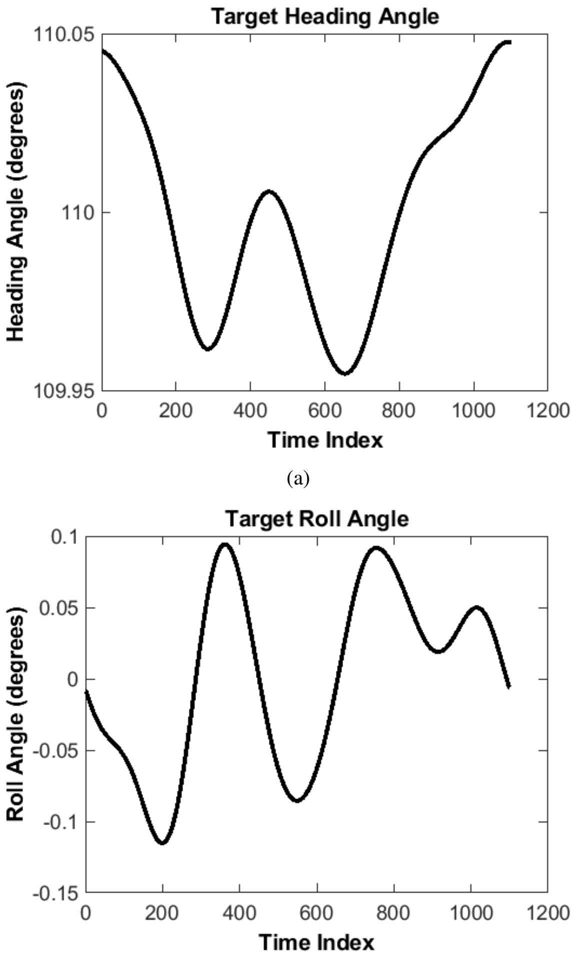
Fig. 3. The injected synthetic target has relatively light maneuver: (a) The true target trajectory projected into the ground plane; (b) True speed of the target.

The injected synthetic target having the relatively light maneuver of Figures 3 and 4 is combined with measured Ku-band SAR image data {Imagery available via Sandia National Lab}. The magnitude image of the combined data is presented in Figure 5(a). For processing within AROMA, a smaller rectangular region is selected about the moving target smear, as given in Figure 5(b).

The selected image chip of Figure 5(b) is applied as the input image for the AROMA processing of Figure 1. The use of 15 iterations of the AROMA process generates the estimates of the three phase error vectors of $\{\widehat{\zeta}_n, \widehat{\mu}_n, \widehat{\phi}_n\}$. The final AROMA refocused AROMA target image is given in Figure 6, revealing relatively good correspondence with the synthetic tank target of Figure 2.

As another example, the amplitude of the variations in roll, heading, and speed are increased significantly, as presented in the truth curves of Figures 7 and 8. Then, the selected image chip of Figure 9(b) is applied as the input image for the AROMA processing of Figure 1. The use of 15 iterations of

Back to Contents



(a)

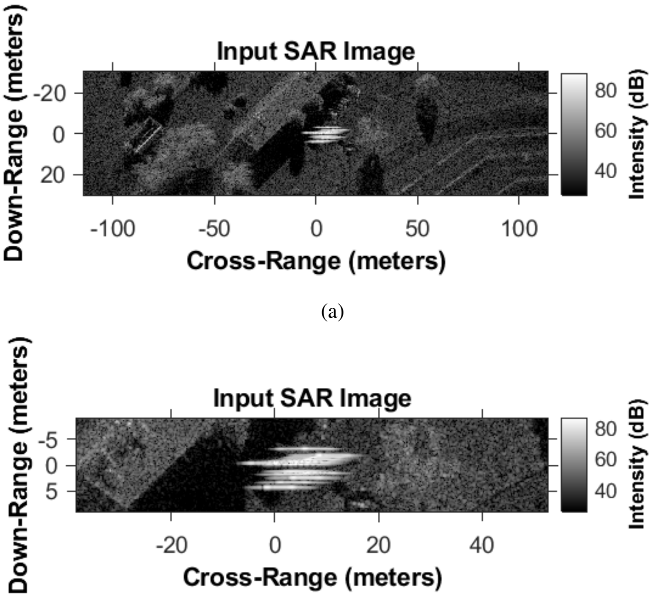
(b)

Fig. 4. The injected synthetic target has relatively light maneuver: (a) True heading of the target; ;(b) True roll of the target.

the AROMA process generates the estimates of the three phase error vectors of $\{\widehat{\zeta}_n, \widehat{\mu}_n, \widehat{\phi}_n\}$. The final AROMA refocused AROMA target image is given in Figure 10, revealing more degraded refocus in comparison with the synthetic tank target of Figure 2.

IV. Conclusions

The present investigation has examined the quality of AROMA refocus for cases of varying levels of target maneuver per variations in target roll, heading, and speed. In the case of a relatively small degree of target maneuver, relatively good target focus was obtained in comparison to the true outline of scattering centers for the injected synthetic target. However, the resulting target focus was degraded for the case of more moderate maneuver in terms of roll, heading, and speed. Future work will include a more comprehensive examination of AROMA performance for more general target maneuver.



(a)

(b)

Fig. 5. The injected synthetic target having the relatively light maneuver of Figures 3 and 4 is combined with measured Ku-band SAR image data {Imagery available via Sandia National Lab}: (a) The magnitude image of the combined data; (b) A selected smaller rectangular region about the moving target smear.

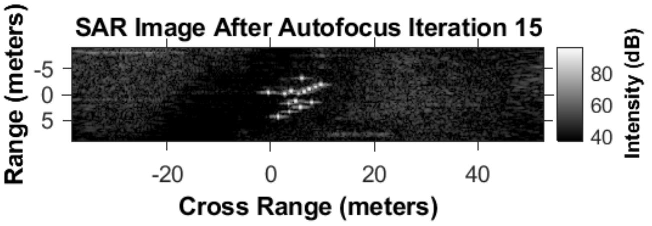


Fig. 6. AROMA processing is applied for 15 iterations to the input image data of Figure 5(b), revealing relatively good correspondence with the synthetic tank target of Figure 2.

References

[1] R. K. Raney, “Synthetic aperture imaging radar and moving targets,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 7, no. 3, pp. pp. 499–505, May 1971.

[2] S. Barbarossa and A. Farina, “A novel procedure for detecting and focusing moving objects with SAR based on the Wigner-Ville distribution,” *IEEE International Radar Conference held in Arlington, Virginia, U.S.A. on 7-10 May 1990*, p. 44, 1990.

[3] S. Barbarossa, “Detection and imaging of moving objects with synthetic aperture radar - Part 1: Optimal detection and parameter estimation theory,” *IEE Proceedings-F*, vol. 139, no. 1, pp. 79–88, Feb 1992.

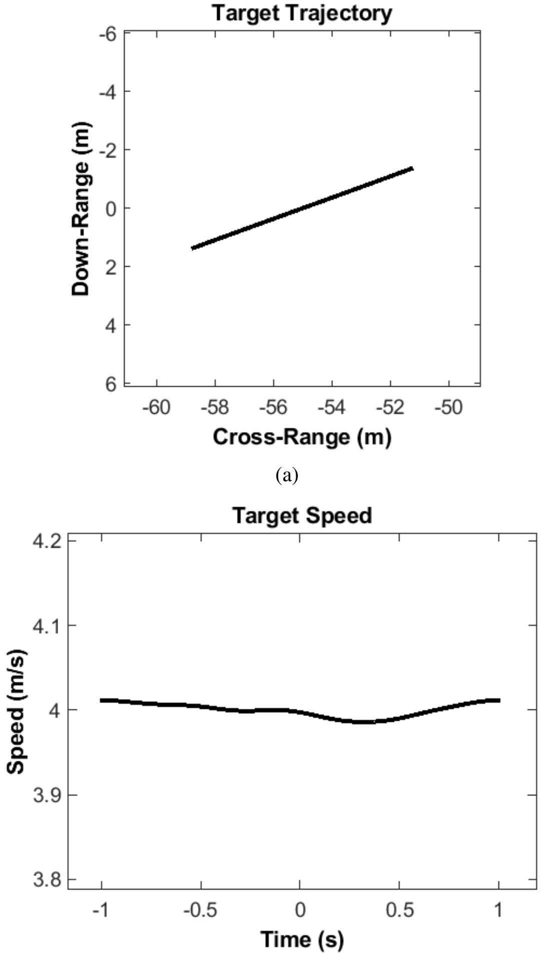
[4] S. Barbarossa and A. Farina, “Detection and imaging of moving objects with synthetic aperture radar - Part 2: Joint time-frequency analysis by Wigner-Ville distribution,” *IEE Proceedings-F*, vol. 139, no. 1, pp. 89–97, Feb 1992.

[5] R. C. DiPietro, R. L. Fante, and R. P. Perry, “Space-based bistatic GMTI using low resolution SAR,” *IEEE Aerospace Conference 1997 held in Aspen, Colorado, U.S.A. on 1-8 February 1997*, vol. 2, pp. 181–193, Feb 1997.

[6] C. V. Jakowatz Jr., D. E. Wahl, and P. H. Eichel, “Refocus of constant velocity moving targets in synthetic aperture radar imagery,” *Proc. SPIE: Algorithms for Synthetic Aperture Radar Imagery V, Edmund G. Zelnio, Editor, held in Orlando, Florida, USA, on 13-17 April 1998*, vol. 3370, pp. 85–95, Apr 1998.

[7] R. P. Perry, R. C. DiPietro, and R. L. Fante, “SAR imaging of moving

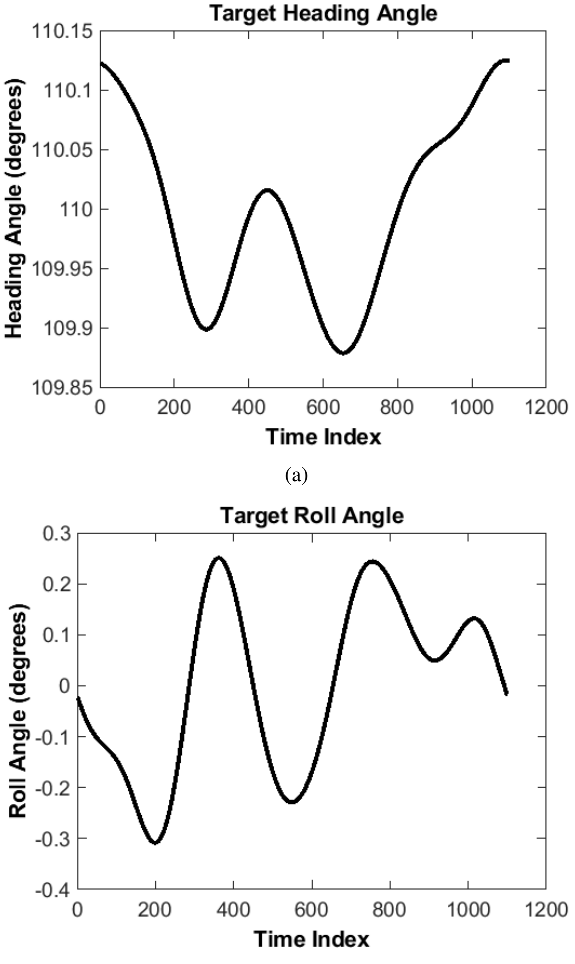
Back to Contents



(a)

(b)

Fig. 7. The injected synthetic target has moderate maneuver: (a) The true target trajectory projected into the ground plane; (b) True speed of the target.



(a)

(b)

Fig. 8. The injected synthetic target has moderate maneuver: (a) True heading of the target; ;(b) True roll of the target.

targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. pp. 188–200, Jan 1999.

[8] M. Kirscht, "Detection and imaging of arbitrarily moving targets with single-channel SAR," *IEE Proceedings - Radar, Sonar, and Navigation*, vol. 150, no. 1, pp. 7–11, Feb 2003.

[9] J. R. Fienup, "Detecting moving targets in SAR imagery by focusing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 3, pp. 794–809, Jul 2001.

[10] J. M. B. Dias and P. A. C. Marques, "Multiple moving target detection and trajectory estimation using a single SAR sensor," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 2, pp. 604–624, Apr 2003.

[11] J. K. Jao, "Theory of synthetic aperture radar imaging of a moving target," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 9, pp. 1984 –1992, Sep 2001.

[12] S. Rahman, "Focusing moving targets using range migration algorithm in ultra wideband low frequency synthetic aperture radar," *Blekinge Institue of Technology - Masters Thesis*, Jun 2010.

[13] P. A. C. Marques and J. M. B. Dias, "Moving targets processing in SAR spatial domain," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 864–874, Jul 2007.

[14] P. Leducq, L. Ferro-Famil, and E. Pottier, "Matching-pursuit-based analysis of moving objects in polarimetric SAR images," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 2, pp. 123–127, Apr 2008.

[15] D. A. Garren, "Method and system for developing and using an image reconstruction algorithm for detecting and imaging moving targets," *U.S. Patent: 7456780 B1; Filed: 26 July 2006; Awarded: 25 November 2008*, 2008.

[16] I. Stojanovic and W. C. Karl, "Imaging of moving targets with multistatic SAR using an overcomplete dictionary," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 1, pp. 164–176, Feb 2010.

[17] M. Cheney and B. Borden, "Waveform-diverse moving-target spotlight SAR," *Proceedings of the 2010 International Waveform Diversity and Design Conference held 8-13 August 2010 in Niagara Falls, Canada*, pp. 33–34, 2010.

[18] D. Cristallini, D. Pastina, F. Colone, and P. Lombardo, "Efficient detection and imaging of moving targets in SAR images based on chirp scaling," *IEEE Transactions on Geoscience and Reomote Sensing*, vol. 51, no. 4, pp. 2403–2416, Apr 2013.

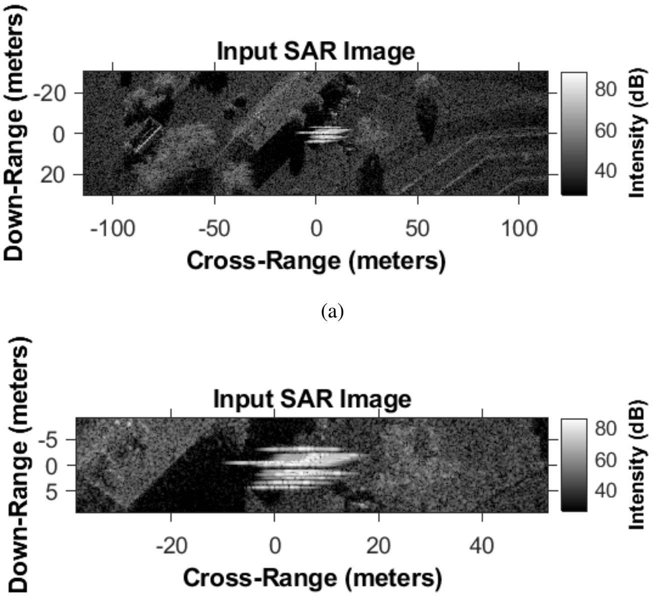
[19] D. A. Garren, "SAR focus theory of complicated range migration signatures due to moving targets," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 4, pp. 557–561, Apr 2018.

[20] ——, "Effects of region of interest selection on phase history based SAR moving target autofocus," *Proceedings of SPIE, Vol. 10987, Algorithms for Synthetic Aperture Radar Imagery XXVI, 14-18 April 2019, in Baltimore, Maryland, USA*, pp. 1 098 703–1 – 1 098 703–11, Apr 2019.

[21] ——, "Dependence of phase history based SAR moving target autofocus on signal-to-clutter ratio," *2019 IEEE International Radar Conference, 22-26 April 2019 in Boston, Massachusetts, USA*, Apr 2019.

[22] V. C. Chen and H. Ling, *Time-Frequency Transforms for Radar Imaging and Signal Analysis*. Norwood, MA, USA: Artech House, 2002.

[23] V. C. Chen and M. Martorella, *Inverse Synthetic Aperture Radar*



(a)

(b)

Fig. 9. The injected synthetic target having the moderate maneuver of Figures 7 and 8 is combined with measured Ku-band SAR image data {Imagery available via Sandia National Lab}: (a) The magnitude image of the combined data; (b) A selected smaller rectangular region about the moving target smear.

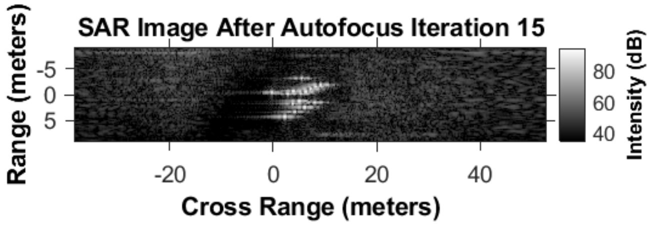


Fig. 10. AROMA processing is applied for 15 iterations to the input image data of Figure 9(b), revealing more degraded refocus in comparison with the synthetic tank target of Figure 2.

Imaging: Principles, Algorithms, and Applications. Edison, NJ, USA: SciTech Publishin, 2014.

[24] F. Berizzi, M. Martorella, and E. Giusti, *Radar Imaging for Maritime Observation*. Boca Raton, FL, USA: CRC Press, 2016.

[25] S. Werness, W. Carrara, L. Joyce, and D. Franczak, "Moving target imaging algorithm forSAR data," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 1, pp. 57–67, Jan 1990.

[26] S. A. Werness, M. A. Stuff, and J. R. Fienup, "Two-dimensional imaging of moving targets in SAR data," *IEEE 24th Asilomar Conference on Signals, Systems and Computers held in Pacific Grove, California, U.S.A. on 5-7 November 1990*, pp. 16–22.

[27] W. G. Carrara, R. S. Goodman, and R. M. Majewski, *Spotlight Synthetic Aperture Radar Signal Processing Algorithms*. Norwood, MA, USA: Artech House, 1995.

[28] B. D. Rigling, "Image-quality focusing of rotating SAR targets," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 750–754, Oct 2008.

[29] D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and C. V. Jackowatz, Jr., "Phase gradient autofocus - a robust tool for high resolution sar phase correction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 3, pp. 827–834, Jul 1994.

[30] C. V. Jakowatz Jr., D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and P. A.Thompson, *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach*. Norwell, MA, USA: Kluwer Academic Publishers, 1996.

[31] D. E. Wahl, D. A. Yocky, and C. V. Jackowatz, "An implementation of a fast backprojection image formation algorithm for spotlight-mode SAR," *Proc. of SPIE Vol. 6970, "Algorithms for Synthetic Aperture Radar Imagery XV" held in Orlando, Florida, U.S.A. on 12 May 2008*, vol. 6970, p. 69700H, 2008.

[32] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part 1: Detection, Estimation, and Linear Modulation Theory*. New York: Whiley and Sons, 1968.

[33] R. J. Schalkoff, *Pattern Recognition: Statistical, Structural and Neural Approaches*. New York: Whiley and Sons, 1992.

[34] A. U. Pillai and T. I. Shim, *Spectrum Estimation and System Identification*. New York: Springer-Verlag, 1993.

[35] D. A. Garren, "Theory of data-driven SAR autofocus to compensate for refraction effects," *IET Radar, Sonar and Navigation - Date of Publication as an IET E-First article: 17 September 2018*, vol. 13, no. 2, pp. 254–262, February 2019.

[36] ——, "Effects of polynomial plus power-law errors on sar refraction autofocus," *Proceedings of the 2019 Sensor Signal Processing for Defence Conference, 9-10 May 2019 in Brighton, UK*, May 2019.

[37] ——, "Perturbation amplitude effects of power law errors on refraction autofocus," *Proceedings of SPIE, Vol. 11393, Algorithms for Synthetic Aperture Radar Imagery XXVII, 24 April 2020, Online Only*, pp. 1 139 304–1 – 1 139 304–9, Apr 2020.

[38] ——, "Robustness of sar refraction autofocus to power-law errors," *2020 IEEE International Radar Conference, 28-30 April 2020; Online Only*, pp. 345–350, Apr 2020.

[39] D. C. Munson, Jr., J. D. O'Brien, and W. K. Jenkins, "A tomographic formulation of spotlight-mode synthetic aperture radar," *Proceedings of the IEEE*, vol. 71, no. 8, pp. 917–925, 1983.

[40] J. L. H. Webb and D. C. Munson, Jr., "SAR image reconstruction for an arbitrary radar path," *1995 International Conference on Acoustics, Speech, and Signal Processing in Detroit, Michigan, USA, on 09 May 1995 - 12 May 1995*, vol. 4, pp. 2285–2288, May 1995.

[41] S. Xiao and D. C. Munson, Jr., "Spotlight-mode SAR imaging of a three-dimensional scence using spectral estimation techniques," *Geoscience and Remote Sensing Symposium Proceedings, 1998. IGARSS '98. 1998 IEEE International, held in Seattle, Washington, USA, on 6-10 July 1998*, vol. 2, pp. 642–644, 1998.

[42] M. Soumekh, *Synthetic Aperture Radar Signal Processing with MATLAB Algorithms*. New York: Whiley and Sons, 1999.

[43] R. J. Sullivan, *Radar Foundations for Imaging and Advanced Concepts*. Raleigh, NC, USA: SciTech Publishing Inc., 2004.

[44] D. A. Garren, "Theory of arbitrary rigid object motion autofocus for non-uniform target rotation and translation," *IET Radar, Sonar and Navigation - Date of Publication as an IET E-First article: 20 September 2020*, September 2020.

[45] Sandia National Laboratories. (2017) Complex SAR data. [Online]. Available: http://www.sandia.gov/radar/complex-data/

Back to Contents

Learning Low-Rank Models From Compressive Measurements for Efficient Projection Design

Fraser K. Coutts, John Thompson, and Bernard Mulgrew
Institute for Digital Communications, University of Edinburgh, Edinburgh, EH9 3FG, UK
email: fraser.coutts@ed.ac.uk

***Abstract*—Recent research has uncovered information-theoretic means to design projection matrices in scenarios where one information source is compressively sampled in the presence of a secondary source. Furthermore, if both sources can be approximated by Gaussian mixture (GM) models, it has been shown that it is possible to learn the characteristics of the secondary source from compressive measurements only. In this work, we investigate techniques that exploit low-rank GM approximations to the true distributions to reduce computational complexity and memory requirements during source learning and projection design. Two novel alternative projection design strategies are also introduced. These are tested against an existing strategy to determine which approach is superior for low size, weight, and power (SWAP) applications. Experimental results validate the benefits of the proposed low-rank strategies and reveal that all projection design algorithms offer similar levels of performance.**

I. Introduction

Designing effective compression strategies is an important problem for both civilian and defence applications. In general, such strategies must dispose of some information to reduce complexity and memory requirements down the signal processing chain. In particular, when designing solutions that are constrained by low size, weight, and power (SWAP) requirements, data reduction is a primary step.

Linear compression strategies often make use of compressive sensing [1] (CS) for effective data reduction. This involves the projection of high-dimensional input data to lower dimensions via random projection matrices. In practice, a random projection might not be the best choice if we know the statistics of the source data [2]. For example, if one is able to effectively characterise source data via a Gaussian mixture (GM) model, information-theoretic methods can be utilised to design projection matrices that can prioritise signal reconstruction or classification [3]–[7]. Importantly, GM distributions can model source data up to an arbitrary level of precision if the number of parameters involved is unbounded [8]. Furthermore, GM models (GMMs) can outperform sparse signal models in some scenarios [9]. Projection matrices designed using GMMs have been shown to be effective in a number of applications, including in image [5] and radar processing [6].

Recent research in [5]–[7] has generalised the projection design approach for a single information source presented in [4] by considering the presence of multiple signals of interest prior to compression. By incorporating secondary information sources, these more recent works are better suited to more general signal processing scenarios. For example, in defence applications, new — potentially adversarial — secondary sources might appear; in this context, adequately extracting information from or mitigating such secondary sources could be vital. Work in [7] addressed this issue by giving specific attention to the learning of secondary information sources via compressive measurements — i.e., without accessing the source data directly. Following the source learning process, it was possible to deploy a more informed compression strategy.

While the adaptive projection design approach in [7] is capable of dealing with new or changing secondary sources, its memory and computational complexity requirements are not ideal for online, low SWAP implementations. This paper explores novel extensions of existing methods to test if lower complexity options are available for GMM-based source learning and information-theoretic projection design. Three novel contributions are provided. The first introduces techniques to learn low-rank GMM approximations to secondary source distributions from compressive measurements. These techniques are extended from the single-source case described in [10]. The second contribution provides some insight into the complexity reductions possible during projection design when incorporating low-rank GM distributions. Finally, we introduce two alternative projection design strategies and test their efficacy against the established strategy of [5] to determine if cost savings can be achieved via algorithms with faster convergence. These alternative strategies are adapted from the literature [5], [6], [12] to consider our specific signal model.

Below, Sec. II introduces our signal-plus-noise compressive sensing model. Sec. III then relates this signal model to our information-theoretic projection design framework. Sec. IV introduces a novel approach for learning low-rank GMM representations for secondary sources from compressive measurements. Sec. V reveals the key expressions required to implement alternative projection design strategies with the goal of reducing convergence time. Sec. VI and Sec. VII provide experimental results and conclusions, respectively.

Notation: Straight bold lowercase and uppercase symbols denote vectors and matrices, respectively, and $\mathbf{I}_n$ is an $n\times n$ identity matrix. Italicised uppercase letters denote random vectors and variables; their realisations are lowercase equivalents. Operators $\{\cdot\}^{\mathrm{H}}$, $\mathbb{E}[\cdot]$, diag$\{\cdot\}$, $\mathrm{vec}\{\cdot\}$, and tr$\{\cdot\}$ evaluate the Hermitian transpose, expectation, diagonal, vector form, and trace, respectively.

II. Signal Model

We utilise the following complex-valued signal-plus-noise compressive sensing model:

$$\boldsymbol{Y} = \mathbf{\Phi}(\boldsymbol{X}+\boldsymbol{N})+\boldsymbol{W}\,, \tag{1}$$

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

with $\boldsymbol{Y},\boldsymbol{W}\in\mathbb{C}^m$, $\boldsymbol{X},\boldsymbol{N}\in\mathbb{C}^n$, $\boldsymbol{\Phi}\in\mathbb{C}^{m\times n}$, and $m\ll n$. Such a model generalises typical compressive sensing scenarios where the secondary source $\boldsymbol{N}$ is not present. To facilitate the modelling of non-Gaussian $\boldsymbol{X}$ and $\boldsymbol{N}$, we assign the following complex-valued GM distributions:

$$\boldsymbol{X}\sim p_{\boldsymbol{x}}(\boldsymbol{x})=\sum\nolimits_{c=1}^{J_{\boldsymbol{x}}} z_c\sum\nolimits_{o=1}^{O}\pi_{c,o}\,\mathcal{CN}(\boldsymbol{x};\boldsymbol{\chi}_{c,o},\boldsymbol{\Omega}_{c,o})\,, \quad (2)$$

$$\boldsymbol{N}\sim p_{\boldsymbol{n}}(\boldsymbol{n})=\sum\nolimits_{g=1}^{J_{\boldsymbol{n}}} v_g\sum\nolimits_{k=1}^{K}s_{g,k}\,\mathcal{CN}(\boldsymbol{n};\boldsymbol{\mu}_{g,k},\boldsymbol{\Gamma}_{g,k})\,. \quad (3)$$

Here, $\boldsymbol{X}$ possesses $J_{\boldsymbol{x}}$ classes with probability z_c, $c=1,\ldots,J_{\boldsymbol{x}}$. Each class is represented by a sum of weighted Gaussians, with weights $\pi_{c,o}$ such that $\sum_o\pi_{c,o}=\sum_c z_c=1$, mean vectors $\boldsymbol{\chi}_{c,o}$, and covariance matrices $\boldsymbol{\Omega}_{c,o}$. In this paper, we consider Gaussian measurement noise; i.e., we have $\boldsymbol{W}\sim\mathcal{CN}(\boldsymbol{w};\boldsymbol{\nu},\boldsymbol{\Lambda})$.

III. Optimisation Framework

As in previous works [5]–[7], we iteratively seek the linear projection $\boldsymbol{\Phi}$ that maximises an information-theoretic objective function. In this case, we consider the following weighted sum of mutual information (MI) terms:

$$F(\boldsymbol{\Phi},\boldsymbol{\beta})=\beta_1 I(\boldsymbol{X};\boldsymbol{Y})+\beta_2 I(C;\boldsymbol{Y})+\beta_3 I(\boldsymbol{N};\boldsymbol{Y})+\beta_4 I(G;\boldsymbol{Y})\,. \quad (4)$$

Here, $I(\boldsymbol{X};\boldsymbol{Y})$ quantifies the MI between input $\boldsymbol{X}$ and output $\boldsymbol{Y}$, and $\boldsymbol{\beta}=[\beta_1,\beta_2,\beta_3,\beta_4]\in\mathbb{R}^4$ controls the relative importance of the MI terms. The classes of $\boldsymbol{X}$ and $\boldsymbol{N}$ are represented by random variables C and G, respectively. Note that negative elements of $\boldsymbol{\beta}$ will result in a $\boldsymbol{\Phi}$ that reduces the relevant information term. Research in [5] has illustrated that positive values for β_1 and β_3 will yield a projection matrix with lower reconstruction errors for $\boldsymbol{X}$ and $\boldsymbol{N}$. Also, choosing $\beta_2,\beta_4>0$ will improve classification for the two sources [6].

IV. Learning a Low-Rank Approximation to the Secondary Source Distribution

A. Adapted Expectation-Maximisation Approach

We have shown in [7] that it is possible to learn the GM distribution of a secondary source $\boldsymbol{N}$ under the signal model of (1) via an expectation-maximisation [11] (EM) approach. The computational complexity and memory requirements of this approach increase with the signal dimension n due to the required manipulation of the GMM covariance matrices. We therefore impose a near-low-rank structure on the covariance matrices of the learned secondary source $\boldsymbol{N}$ such that we have

$$\boldsymbol{\Gamma}_k=\mathbf{F}_k\mathbf{F}_k^{\mathrm{H}}+\eta\,\mathbf{I}_n\,,\quad k=1,\ldots,K\,, \quad (5)$$

where $\mathbf{F}_k\in\mathbb{C}^{n\times r_k}$, $r_k\ll n$, and $0<\eta\ll 1$. By manipulating the 'tall' matrix $\mathbf{F}_k$ in the below approach, we reduce our memory footprint and incur lower computational costs than if we were to use $\boldsymbol{\Gamma}_k$ directly. Here, we have considered a secondary source $\hat{\boldsymbol{N}}$ with only one class:

$$\hat{\boldsymbol{N}}\sim\sum\nolimits_{k=1}^{K}s_k\,\mathcal{CN}(\hat{\boldsymbol{n}};\boldsymbol{\mu}_k,\boldsymbol{\Gamma}_k)\,. \quad (6)$$

We use this single-class form throughout this section to simplify the description of the source learning process.

We learn the distribution of $\hat{\boldsymbol{N}}$ from N_s measurements

$$\{\boldsymbol{y}_i=\boldsymbol{\Phi}_i\hat{\boldsymbol{n}}_i+\hat{\boldsymbol{w}}_i\}\,,\quad i=1,\ldots,N_s\,, \quad (7)$$

where the matrices $\boldsymbol{\Phi}_i$ are randomly generated with elements drawn from $\mathcal{CN}(0,1)$. Each vector $\hat{\boldsymbol{w}}_i$ is an instance of

$$\hat{\boldsymbol{W}}_i\sim\sum\nolimits_{d=1}^{D}\tau_d\,\mathcal{CN}(\hat{\boldsymbol{w}}_i;\boldsymbol{\nu}_d^i,\boldsymbol{\Lambda}_d^i)\,, \quad (8)$$

$$\tau_d=z_{c'}\pi_{c',o'}\,,\quad \boldsymbol{\nu}_d^i=\boldsymbol{\Phi}_i\boldsymbol{\chi}_{c',o'}+\boldsymbol{\nu},\quad \boldsymbol{\Lambda}_d^i=\boldsymbol{\Phi}_i\boldsymbol{\Omega}_{c',o'}\boldsymbol{\Phi}_i^{\mathrm{H}}+\boldsymbol{\Lambda}\,,$$
$$D=J_{\boldsymbol{x}}O\,,\qquad c'=\left\lceil\tfrac{d}{O}\right\rceil\,,\qquad o'=((d-1)\bmod O)+1.$$

Learning the multiple classes described by the full GM distribution of $\boldsymbol{N}$ can be achieved by executing the source learning process for several batches of size N_s and recognising similar GM distributions via, e.g., the Kullback–Leibler divergence. Different classes will yield suitably different GM distributions.

In this low-rank form, we can express a sample from the kth GM component of $\hat{\boldsymbol{N}}$ as

$$\hat{\boldsymbol{n}}^k=\mathbf{F}_k\boldsymbol{b}+\boldsymbol{\mu}_k+\boldsymbol{u}\,, \quad (9)$$

where $p_{\boldsymbol{b}}(\boldsymbol{b})=\mathcal{CN}(\boldsymbol{b};\mathbf{0},\mathbf{I}_{r_k})$ and $p_{\boldsymbol{u}}(\boldsymbol{u})=\mathcal{CN}(\boldsymbol{u};\mathbf{0},\eta\mathbf{I}_n)$. We can therefore write

$$\hat{\boldsymbol{N}}\sim\sum\nolimits_{k=1}^{K}s_k\int\mathcal{CN}(\hat{\boldsymbol{n}};\boldsymbol{\mu}_k+\mathbf{F}_k\boldsymbol{b},\eta\mathbf{I}_n)\,\mathcal{CN}(\boldsymbol{b};\mathbf{0},\mathbf{I}_{r_k})\,\mathrm{d}\boldsymbol{b}. \quad (10)$$

We use an EM approach to find the system parameters θ that maximise the marginal log-likelihood

$$\log p_{\mathbf{y}|\theta}(\boldsymbol{y}|\theta)=\log\sum_{k,d}\iint p_{\mathbf{y},\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\theta}(\boldsymbol{y},\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\theta)\mathrm{d}\hat{\boldsymbol{n}}\mathrm{d}\boldsymbol{b}\,. \quad (11)$$

Unlike the procedure in [7], we iteratively update $\mathbf{F}_k$ instead of $\boldsymbol{\Gamma}_k$ by extending the single-source work of [10]. In iteration $(t+1)$, we maximise the expected value of the complete log-likelihood given access to the current system parameters $\theta^{(t)}$:

$$\ell_{\mathrm{EC}}\left(\theta\middle|\theta^{(t)}\right)=\sum\nolimits_{i=1}^{N_s}\underset{\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\boldsymbol{y}_i,\theta^{(t)}}{\mathbb{E}}\left[\log p^i_{\mathbf{y},\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\theta}(\boldsymbol{y}_i,\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\theta)\right]. \quad (12)$$

Here, the superscript i indicates the reliance of the distribution function on i via $\boldsymbol{\Phi}_i$. With θ omitted for brevity, we can write

$$p^i_{\mathbf{y},\hat{\boldsymbol{n}},k,d,\boldsymbol{b}}(\boldsymbol{y}_i,\hat{\boldsymbol{n}},k,d,\boldsymbol{b})=$$
$$p^i_{\mathbf{y}|\hat{\boldsymbol{n}},d}(\boldsymbol{y}_i|\hat{\boldsymbol{n}},d)p_{\hat{\boldsymbol{n}}|\boldsymbol{b},k}(\hat{\boldsymbol{n}}|\boldsymbol{b},k)p_{\boldsymbol{b}}(\boldsymbol{b})s_k\tau_d\,. \quad (13)$$

With $\tilde{\boldsymbol{b}}=[\boldsymbol{b}^{\mathrm{T}},1]^{\mathrm{T}}$ and $\tilde{\mathbf{F}}_k=[\mathbf{F}_k,\boldsymbol{\mu}_k]$, the gradient of $\ell_{\mathrm{EC}}(\theta|\theta^{(t)})$ with respect to $\tilde{\mathbf{F}}_k$ is

$$\nabla_{\tilde{\mathbf{F}}_k}\ell_{\mathrm{EC}}\left(\theta\middle|\theta^{(t)}\right)=\sum_{i=1}^{N_s}\underset{\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\boldsymbol{y}_i,\theta^{(t)}}{\mathbb{E}}\left[\nabla_{\tilde{\mathbf{F}}_k}\log p_{\hat{\boldsymbol{n}}|\boldsymbol{b},k}(\hat{\boldsymbol{n}}|\boldsymbol{b},k)\right]=$$
$$\sum_{i=1}^{N_s}\underset{\hat{\boldsymbol{n}},k,d,\boldsymbol{b}|\boldsymbol{y}_i,\theta^{(t)}}{\mathbb{E}}\left[-\nabla_{\tilde{\mathbf{F}}_k}\left(\hat{\boldsymbol{n}}-\tilde{\mathbf{F}}_k\tilde{\boldsymbol{b}}\right)^{\mathrm{H}}(\eta\mathbf{I}_n)^{-1}\left(\hat{\boldsymbol{n}}-\tilde{\mathbf{F}}_k\tilde{\boldsymbol{b}}\right)\right]. \quad (14)$$

Setting the gradient to zero yields the updated parameters

$$\left[\mathbf{F}_k^{(t+1)},\boldsymbol{\mu}_k^{(t+1)}\right]=\sum_{i,d}p^i_{k,d|\mathbf{y}}(k,d|\boldsymbol{y}_i)\left[\mathbf{A}^i_{k,d}+\tilde{\boldsymbol{\mu}}^i_{k,d}\tilde{\boldsymbol{\eta}}^i_{k,d}{}^{\mathrm{H}},\tilde{\boldsymbol{\mu}}^i_{k,d}\right]$$
$$\times\left(\sum_{i,d}p^i_{k,d|\mathbf{y}}(k,d|\boldsymbol{y}_i)\begin{bmatrix}\mathbf{B}^i_{k,d}+\tilde{\boldsymbol{\eta}}^i_{k,d}\tilde{\boldsymbol{\eta}}^i_{k,d}{}^{\mathrm{H}} & \tilde{\boldsymbol{\eta}}^i_{k,d}\\ \tilde{\boldsymbol{\eta}}^i_{k,d}{}^{\mathrm{H}} & 1\end{bmatrix}\right)^{-1}, \quad (15)$$

$$p^i_{k,d|\mathbf{y}}(k,d|\boldsymbol{y}_i)=s_k\,\tau_d\,p^i_{\mathbf{y}|k,d}(\boldsymbol{y}_i|k,d)\,/\,p^i_{\mathbf{y}}(\boldsymbol{y}_i)\,, \quad (16)$$

$$p^i_{\mathbf{y}|k,d}(\boldsymbol{y}_i|k,d)=\mathcal{CN}(\boldsymbol{y}_i;\boldsymbol{\Phi}_i\boldsymbol{\mu}_k+\boldsymbol{\nu}^i_d,\boldsymbol{\Phi}_i\boldsymbol{\Gamma}_k\boldsymbol{\Phi}_i^{\mathrm{H}}+\boldsymbol{\Lambda}^i_d)\,, \quad (17)$$

$$\tilde{\boldsymbol{\mu}}^i_{k,d}=\boldsymbol{\mu}_k+\mathbf{C}^i_{k,d}\boldsymbol{\Phi}_i^{\mathrm{H}}(\boldsymbol{\Lambda}^i_d)^{-1}\left(\boldsymbol{y}_i-\boldsymbol{\Phi}_i\boldsymbol{\mu}_k-\boldsymbol{\nu}^i_d\right)\,, \quad (18)$$

$$\tilde{\boldsymbol{\eta}}^{i}_{k,d} = \mathbf{F}_k^{\mathrm{H}}\mathbf{\Phi}_i^{\mathrm{H}}\left(\mathbf{\Lambda}_d^i + \mathbf{\Phi}_i\mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\right)^{-1}\left(\boldsymbol{y}_i - \mathbf{\Phi}_i\boldsymbol{\mu}_k - \boldsymbol{\nu}_d^i\right) , \quad (19)$$

$$\mathbf{A}^{i}_{k,d} = \mathbf{F}_k - \mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\left(\mathbf{\Lambda}_d^i + \mathbf{\Phi}_i\mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\right)^{-1}\mathbf{\Phi}_i\mathbf{F}_k\,, \quad (20)$$

$$\mathbf{B}^{i}_{k,d} = \mathbf{I}_{r_k} - \mathbf{F}_k^{\mathrm{H}}\mathbf{\Phi}_i^{\mathrm{H}}\left(\mathbf{\Lambda}_d^i + \mathbf{\Phi}_i\mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\right)^{-1}\mathbf{\Phi}_i\mathbf{F}_k\,, \quad (21)$$

$$\mathbf{C}^{i}_{k,d} = \mathbf{\Gamma}_k - \mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\left(\mathbf{\Lambda}_d^i + \mathbf{\Phi}_i\mathbf{\Gamma}_k\mathbf{\Phi}_i^{\mathrm{H}}\right)^{-1}\mathbf{\Phi}_i\mathbf{\Gamma}_k\,. \quad (22)$$

Similarly, under a constraint of $\sum_k s_k = 1$, we can obtain

$$s_k^{(t+1)} = \frac{\sum_{i=1}^{N_s} p_{k|\boldsymbol{y}}(k|\boldsymbol{y}_i)}{\sum_{i=1}^{N_s}\sum_{k'=1}^{K} p_{k|\boldsymbol{y}}(k'|\boldsymbol{y}_i)} = \frac{\sum_{i=1}^{N_s} p_{k|\boldsymbol{y}}(k|\boldsymbol{y}_i)}{N_s}\,. \quad (23)$$

The above equations enable the iterative update of the low-rank GM parameters for $\boldsymbol{N}$. The iterative process ceases if the log-likelihood in (11) is no longer significantly increasing, or if a preselected number of iterations have elapsed.

B. Benefits of Reduced Rank

As a simple, representative example of the benefits of a low-rank representation, consider the matrix multiplication operation $\mathbf{\Phi}\mathbf{\Gamma}_k\mathbf{\Phi}^{\mathrm{H}}$, which evaluates the covariance of component k of $\hat{\boldsymbol{N}}$ after projection via $\mathbf{\Phi}$. Due to the iterative update of $\mathbf{\Phi}$ in methods such as [5], this operation must be executed at each iteration of the information-theoretic projection design algorithm with a complexity of order $\mathcal{O}(mn^2 + m^2n)$. If we instead compute $\mathbf{\Phi}\mathbf{F}_k\mathbf{F}_k^{\mathrm{H}}\mathbf{\Phi}^{\mathrm{H}}$ to approximate this operation using our low-rank representation with $r_k = r\,\forall\,k$, we require a complexity of order $\mathcal{O}(mnr + m^2r)$. Since $r \leq n$, our approximation will generally have lower complexity requirements. Considering that such matrix multiplications form a significant cost at each iteration of the utilised projection design algorithms, reduced computation and increased algorithm speed should be expected. As r approaches n, the complexity of a low-rank projection design algorithm should approach that of a full-rank implementation. Additional benefits can be expected if $\boldsymbol{X}$ is also given a low-rank representation.

V. Two-Stage Information-Theoretic Algorithms

In this section, we derive two alternative strategies for the iterative design of projection matrix $\mathbf{\Phi}$ subject to the objective function in (4) with the goal of achieving faster convergence and therefore lower system complexity.

Applying the singular value decomposition and eigenvalue decomposition to the projection matrix and measurement noise covariance matrix, respectively, yields $\mathbf{\Phi} = \mathbf{U_\Phi}\mathbf{D_\Phi}\mathbf{V}_{\mathbf{\Phi}}^{\mathrm{H}}$ and $\mathbf{\Lambda} = \mathbf{U_\Lambda}\mathbf{D_\Lambda}\mathbf{U}_{\mathbf{\Lambda}}^{\mathrm{H}}$. If $\mathbf{U_\Phi} = \mathbf{U_\Lambda}$, we can choose

$$\bar{\boldsymbol{Y}} = \mathbf{D}_{\mathbf{\Lambda}}^{-1/2}\mathbf{U}_{\mathbf{\Lambda}}^{\mathrm{H}}\boldsymbol{Y} = \mathbf{D}_{\mathbf{\Lambda}}^{-1/2}\mathbf{D_\Phi}\mathbf{V}_{\mathbf{\Phi}}^{\mathrm{H}}(\boldsymbol{X} + \boldsymbol{N}) + \bar{\boldsymbol{W}}\,, \quad (24)$$

with $\bar{\boldsymbol{W}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_m)$ (assuming zero mean without loss of generality). Below, $\mathbf{E}_z$ is a weighted sum of minimum mean-square error (MMSE) matrices according to

$$\mathbf{E}_z = \beta_1\mathbf{E}_{z,\boldsymbol{x}} + \beta_2\mathbf{E}_{z,c} + \beta_3\mathbf{E}_{z,\boldsymbol{n}} + \beta_4\mathbf{E}_{z,g}\,, \quad (25)$$

where $\mathbf{E}_{z,\boldsymbol{x}}$ and $\mathbf{E}_{z,c}$ are complex-valued versions of the MMSE matrices for $\boldsymbol{X}$ and its classes C from [5] and $\mathbf{E}_{z,\boldsymbol{n}}$ and $\mathbf{E}_{z,g}$ are the MMSE matrices for $\boldsymbol{N}$ and its classes G.

Theorem 1 (Gradient expressions for $F(\mathbf{\Phi}, \boldsymbol{\beta})$)**.** With $\mathbf{G} = \mathbf{D_\Phi}\mathbf{\Theta}$, $\mathbf{\Theta} = \mathbf{V}_{\mathbf{\Phi}}^{\mathrm{H}}$, $\mathbf{H} = \mathbf{D}_{\mathbf{\Lambda}}^{-1/2}$, $\mathbf{P} = \mathbf{G}^{\mathrm{H}}\mathbf{H}^{\mathrm{H}}\mathbf{H}\mathbf{G}$, $\mathbf{D_\Phi} = \hat{\mathbf{D}}_{\mathbf{\Phi}}\hat{\mathbf{I}}$, and $\hat{\mathbf{I}} = [\mathbf{I}_m, \mathbf{0}]$, we can evaluate the following gradients of $F(\mathbf{\Phi}, \boldsymbol{\beta})$ for the signal model of (24):

$$\nabla_{\mathbf{G}}F(\mathbf{\Phi}, \boldsymbol{\beta}) = \mathbf{H}^{\mathrm{H}}\mathbf{H}\mathbf{G}\mathbf{E}_z\,, \quad (26)$$

$$\nabla_{\mathbf{\Theta}}F(\mathbf{\Phi}, \boldsymbol{\beta}) = \mathbf{D}_{\mathbf{\Phi}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\mathbf{D_\Phi}\mathbf{\Theta}\mathbf{E}_z\,, \quad (27)$$

$$\nabla_{\mathbf{P}}F(\mathbf{\Phi}, \boldsymbol{\beta}) = \mathbf{E}_z\,, \quad (28)$$

$$\nabla_{\hat{\mathbf{D}}_{\mathbf{\Phi}}^2}F(\mathbf{\Phi}, \boldsymbol{\beta}) = \mathrm{diag}\Big\{\mathrm{diag}\Big\{\hat{\mathbf{I}}\mathbf{\Theta}\mathbf{E}_z\mathbf{\Theta}^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\Big\}\Big\}\,. \quad (29)$$

Proof. See derivations in [12] and adapt to the signal model of (24) by using the gradient expressions in [5], [6]. ■

With the above expressions, our task is now to iteratively seek the best $\mathbf{G} = \mathbf{D_\Phi}\mathbf{\Theta}$ by updating $\mathbf{D_\Phi}$ and $\mathbf{\Theta}$ in sequence. Fortunately, we know $\nabla_{\hat{\mathbf{D}}_{\mathbf{\Phi}}^2}F(\mathbf{\Phi}, \boldsymbol{\beta})$ and at odd iteration numbers can update the squared singular values via $\hat{\mathbf{D}}_{\mathbf{\Phi}}^2 \leftarrow \hat{\mathbf{D}}_{\mathbf{\Phi}}^2 + \mu_D\nabla_{\hat{\mathbf{D}}_{\mathbf{\Phi}}^2}F(\mathbf{\Phi}, \boldsymbol{\beta})$ for some step size μ_D, subject to a power constraint.

We consider two options for the update of the unitary matrix $\mathbf{\Theta}$, which occurs at even iteration numbers. The first is a simple gradient ascent operation such that $\mathbf{\Theta} \leftarrow \mathrm{orth}\{\mathbf{\Theta} + \mu_{\mathbf{\Theta}}\nabla_{\mathbf{\Theta}}F(\mathbf{\Phi}, \boldsymbol{\beta})\}$, where $\mathrm{orth}\{\cdot\}$ identifies the nearest orthonormal matrix. The second approach extends the work of [12] and expresses $\mathbf{\Theta}$ as product of Givens rotations $\mathbf{U}_{pq}(\omega_{pq}, \nu_{pq})$ weighted by a diagonal matrix $\mathbf{D}_{\mathbf{\Theta}}$:

$$\mathbf{\Theta} = \mathbf{D}_{\mathbf{\Theta}}\prod_{p=n-1}^{1}\prod_{q=p+1}^{n}\mathbf{U}_{pq}(\omega_{pq}, \nu_{pq})\,. \quad (30)$$

Matrix $\mathbf{P}(\omega_{pq}, \nu_{pq})$ is therefore a function of ω_{pq} and ν_{pq}. We identify the parameter changes $(\delta\omega_{pq}, \delta\nu_{pq})$ required to enforce a change of $\mathbf{P}(\omega_{pq}+\delta\omega_{pq}, \nu_{pq}+\delta\nu_{pq}) - \mathbf{P}(\omega_{pq}, \nu_{pq}) = \delta\mathbf{P}(\omega_{pq}, \nu_{pq}) = \mu_P\nabla_{\mathbf{P}}F(\mathbf{\Phi}, \boldsymbol{\beta}) = \mu_P\mathbf{E}_z$, for some step size μ_P. A first-order approximation to this change is:

$$\delta\mathbf{P} \approx [\delta\mathbf{\Theta}]^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\hat{\mathbf{D}}_{\mathbf{\Phi}}^2\hat{\mathbf{I}}\mathbf{\Theta} + \mathbf{\Theta}^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\hat{\mathbf{D}}_{\mathbf{\Phi}}^2\hat{\mathbf{I}}[\delta\mathbf{\Theta}]\,, \quad (31)$$

$$\delta\mathbf{\Theta} = \sum_{p=n-1}^{1}\sum_{q=p+1}^{n}\frac{\partial\mathbf{\Theta}}{\partial\omega_{pq}}\delta\omega_{pq} + \sum_{p=n-1}^{1}\sum_{q=p+1}^{n}\frac{\partial\mathbf{\Theta}}{\partial\nu_{pq}}\delta\nu_{pq}\,. \quad (32)$$

Thus, if we reindex with (ω_j, ν_j), $j = 1, \ldots, N_G$, and $N_G = n(n-1)/2$, we can write

$$\mathrm{vec}(\mathbf{\Theta}^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\hat{\mathbf{D}}_{\mathbf{\Phi}}^2\hat{\mathbf{I}}[\delta\mathbf{\Theta}]) = \sum_{j=1}^{N_G}\Bigg[\mathrm{vec}(\mathbf{\Theta}^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\hat{\mathbf{D}}_{\mathbf{\Phi}}^2\hat{\mathbf{I}}\frac{\partial\mathbf{\Theta}}{\partial\omega_j}), \\ \mathrm{vec}(\mathbf{\Theta}^{\mathrm{H}}\hat{\mathbf{I}}^{\mathrm{H}}\mathbf{D}_{\mathbf{\Lambda}}^{-1}\hat{\mathbf{D}}_{\mathbf{\Phi}}^2\hat{\mathbf{I}}\frac{\partial\mathbf{\Theta}}{\partial\nu_j})\Bigg] \times \left[\delta\omega_j, \delta\nu_j\right]^{\mathrm{T}}, \quad (33)$$

$$\mathrm{vec}(\delta\mathbf{P}) \approx \sum\nolimits_{i=j}^{N_G}\mathbf{\Xi}_j\left[\delta\omega_j, \delta\nu_j\right]^{\mathrm{T}} = \mathbf{\Xi}\boldsymbol{\vartheta}\,, \quad (34)$$

where $\mathbf{\Xi}_j$, $\mathbf{\Xi}$, and $\boldsymbol{\vartheta}$ can be inferred from the preceding equations. To update $\mathbf{\Theta}$, we find the parameters $\boldsymbol{\vartheta}$ that minimise $\|\mathrm{vec}(\delta\mathbf{P}) - \mathbf{\Xi}\boldsymbol{\vartheta}\|_2^2$, subject to angle constraints on ω_{pq} and ν_{pq}, via constrained least-squares optimisation.

VI. Experimental Results

A. Quality of Low-Rank Approximations

1) Experiments with Synthetic Data : In this section, we conduct simulations to confirm that low-rank approximations

Back to Contents

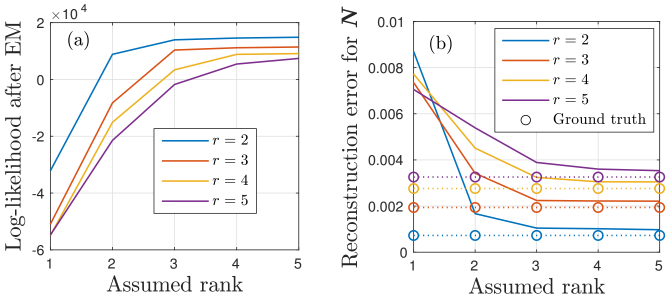


Fig. 1. (a) Log-likelihood of compressive measurements after source learning via EM and (b) reconstruction error for $\boldsymbol{N}$ versus assumed rank of covariance matrices $\boldsymbol{\Gamma}_k$ for actual ranks $r \in \{2, 3, 4, 5\}$.

TABLE I
SOURCE LEARNING RUN TIME VERSUS ASSUMED RANK

Rank	1	2	3	4	5
Time	61.6 s	69.6 s	76.0 s	82.4 s	90.3 s

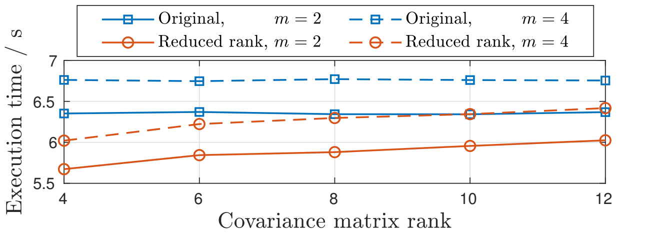


Fig. 2. GMM rank versus required projection design execution times. The original method of [5] is compared with a low-rank-optimised version.

to the GM distributions of secondary sources can be obtained from compressive measurements. Synthetic input data of dimension $n = 16$ is generated for single-class inputs $\boldsymbol{X}$ and $\boldsymbol{N}$ ($J_{\boldsymbol{x}} = J_{\boldsymbol{n}} = 1$) with $O = K = 3$. The weights $\pi_{c,o}$ and $s_{g,k}$ are drawn from the standard uniform distribution and normalised. The mean vectors $\boldsymbol{\chi}_{c,o}$ and $\boldsymbol{\mu}_{g,k}$ comprise elements drawn from $\mathcal{CN}(0, 3\sqrt{2}/10)$, and the covariance matrices $\boldsymbol{\Omega}_{c,o}$ and $\boldsymbol{\Gamma}_{g,k}$ are approximately low-rank and equal to instances of $\mathbf{F}\mathbf{F}^{\mathrm{H}} + \eta \mathbf{I}_n$, where $\mathbf{F} = \mathbf{Q}\mathbf{D}_r \in \mathbb{C}^{n \times r}$, $\mathbf{Q} \in \mathbb{C}^{n \times n}$ is a random unitary matrix, $\mathbf{D}_r = [\mathbf{D}, \mathbf{0}]^{\mathrm{T}}$, and $\mathbf{D} \in \mathbb{R}^{r \times r}$ is a diagonal matrix with elements drawn from $\mathcal{U}(10^{-6}, 10^{-2})$ and normalised such that $\mathrm{tr}\{\mathbf{D}^2\} = 0.1$ for $\boldsymbol{N}$ and $\mathrm{tr}\{\mathbf{D}^2\} = 0.01$ for $\boldsymbol{X}$. We demonstrated the impact of the relative powers between $\boldsymbol{X}$ and $\boldsymbol{N}$ on source learning performance in [7]. We use a value of $\eta = 10^{-4}$ over 10^2 simulation instances and average the results. We use $N_s = 1000$ random measurements and 250 iterations during the training of the GMM for $\boldsymbol{N}$. Our compressive measurements are of dimension $m = 8$. The measurement noise is characterised by $\boldsymbol{W} \sim \mathcal{CN}(\boldsymbol{w}; \mathbf{0}, 10^{-6}\mathbf{I}_m)$.

In Fig. 1, we demonstrate the impact of an incorrect assumption of the rank of $\boldsymbol{\Gamma}_k$. Fig. 1(a) highlights that as the assumed rank exceeds the true rank $r \in \{2, 3, 4, 5\}$, we reach a plateau in terms of distribution log-likelihood. The log-likelihood here quantifies the quality of fit of a distribution to the data, with a higher log-likelihood signifying a better fit. These results are bolstered by Fig. 1(b), which compares the mean-square reconstruction errors obtained for $\boldsymbol{N}$ when utilising ground truth or estimated distributions. Here, $\boldsymbol{N}$ is reconstructed from compressive measurements obtained using a fixed, random projection matrix and the Bayesian inference model described in [5]. A more accurate estimation of the distribution of $\boldsymbol{N}$ will yield a lower reconstruction error. In general, increasing the assumed rank improves performance in terms of log-likelihood and reconstruction error; however, there is an associated cost in terms of computational complexity. Table I illustrates this by comparing the assumed rank with the time taken for source learning to complete. An approximately linear relationship is observed, with each increase in assumed rank increasing the run time by 7.2 seconds on average.

To test how low-rank models benefit projection design complexity, we paired GMMs with known rank with a low-rank-optimised version of the method of [5]. Using this version, we were able to design projection matrices with $m \in \{2, 4\}$ that maximised the objective function of (4) with $\boldsymbol{\beta} = [1, 0, 0, 0]$. Fig. 2 compares the algorithm run times (averaged over 10^2 instantiations) after 100 iterations when using low-rank covariance matrices for $\boldsymbol{X}$ and $\boldsymbol{N}$ — with the distributions for each input as given above and known *a priori* during projection design. Here, we can observe an approximately linear relationship between rank and algorithm execution time for the optimised implementations, with an approximate 10% decrease in computation time for a rank of $r = 4$. As the rank increases, we return to the execution times demanded by a non-rank-optimised projection design implementation.

2) Experiments with Real Radar Data : In this section, we test the ability of the proposed low-rank source learning methodology to estimate the distribution of an unknown secondary source. For this, we use a radar dataset that emulates a test scenario in which two helicopters are present. The data is in the form of recorded micro-Doppler [13] radar returns from two static fans with rotating blades. Each fan had three potential speeds, which we consider as classes. The returns from the first and second fans are assigned to inputs $\boldsymbol{X}$ and $\boldsymbol{N}$ respectively, in a fashion that replicates the steps of [6], [7]. We refer the reader to these works for a full description of the data processing involved.

Our test scenario involves *a priori* knowledge of the GM distribution for $\boldsymbol{X}$, which is of dimension $n = 32$ and has $J_{\boldsymbol{x}} = 3$ classes with $O = 1$ component each. We deploy a low-rank version of the method from [7] on a vector-valued sequence of radar returns of length $N_T = 4250$ obtained when Fan 1 ($\boldsymbol{X}$) is rotating at its slowest speed (class 1). The middle 3000 samples of this sequence are corrupted by additive noise in the form of Fan 2 ($\boldsymbol{N}$) rotating at its slowest speed. Under the assumption of $\boldsymbol{N} \approx \mathbf{0}$, we have an initial $\boldsymbol{\Phi}_{\mathrm{opt}} \in \mathbb{C}^{4 \times n}$ that has been designed to maximise the classification accuracy for $\boldsymbol{X}$ according to Sec. III with $\boldsymbol{\beta} = [0, 1, 0, 0]$. We wish to use compressive measurements of dimension $m = 16$ obtained from the middle 3000 samples to approximate the distribution of the present class of $\boldsymbol{N}$ with $K = 1$, and to use this approximation to design a more effective $\boldsymbol{\Phi}_{\mathrm{opt}}$. Our measurement noise covariances during source learning and

Back to Contents

TABLE II
CLASSIFICATION ACCURACY (CA) FOR $\boldsymbol{X}$ VERSUS ASSUMED RANK

Rank	8	10	12	14	16
CA	60.3%	65.8%	70.6%	71.1%	71.2%

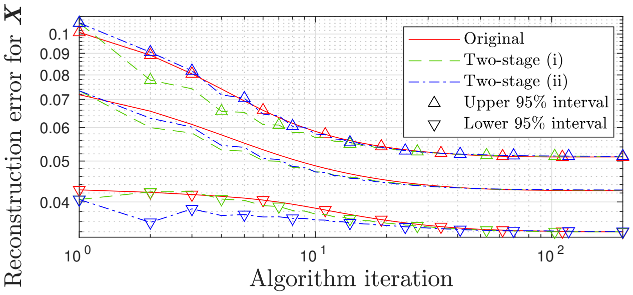


Fig. 3. Reconstruction error for $\boldsymbol{X}$ versus algorithm iteration for the original gradient ascent approach as implemented in [5] and the developed (i) dual gradient ascent and (ii) Given's rotation-based two-stage implementations.

projection design are $10^{-6}\mathbf{I}_m$ and $10^{-6}\mathbf{I}_4$, respectively.

Our low-rank learning approach is applied to the data sequence for assumed ranks $r \in \{8, 10, 12, 14, 16\}$. The source learning process was allowed to run for 2000 iterations or until the change in log-likelihood between iterations dropped below 1. Table II shows the resulting classification accuracies for $\boldsymbol{X}$ for an identically constructed, unseen sequence of data following the redesign of $\boldsymbol{\Phi}_{\text{opt}}$ for each assumed rank. As with the synthetic data, we see the performance increase with the assumed rank until the underlying rank is matched — at which point, the performance plateaus. From this, we can ascertain that this class of $\boldsymbol{N}$ can be adequately modelled with a covariance matrix of rank $r = 16$; however, $r = 12$ would reduce complexity without significantly impacting performance. The classification accuracy when using the original $\boldsymbol{\Phi}_{\text{opt}}$ — without accounting for the presence of $\boldsymbol{N}$ — was 29.7%.

B. Comparing Projection Design Strategies

In this section, we deploy the projection design strategies of [5] and Sec. V with $\boldsymbol{\beta} = [1, 0, 0, 0]$ on a full-rank version of the simulation scenario of Sec. VI-A1 with $m = 3$ and $n = 9$. Results are averaged over 10^2 randomised instances with source parameters as defined above. Here, we do not normalise $\text{tr}\{\mathbf{D}^2\}$ and we use $\boldsymbol{W} \sim \mathcal{CN}(\boldsymbol{w}; \mathbf{0}, 10^{-2}\mathbf{I}_m)$. All projection matrices were normalised such that $\text{tr}\{\boldsymbol{\Phi}\boldsymbol{\Phi}^{\text{H}}\} = m$.

By plotting the mean-square reconstruction error for input $\boldsymbol{X}$ over 200 projection design iterations, Fig. 3 demonstrates that all considered strategies perform similarly on average, subject to small deviations in algorithm convergence due to the respective gradient ascent step sizes, which were determined experimentally. Here, for a low number of algorithm iterations, the proposed two-stage implementations offer a slight advantage. While they do not allow us to concretely favour one algorithm over another, these simulations validate earlier research that utilised a more straightforward gradient ascent approach [5], [6], since performance upon convergence is approximately equal for all methods.

VII. CONCLUSIONS

In this paper, we have investigated techniques that exploit low-rank GMM approximations to source data distributions to reduce computational complexity and memory requirements during source learning and projection design. Simulations with both real and synthetic data have validated the benefits of the proposed low-rank strategies. Importantly, reducing the rank can decrease computational complexity for low SWAP applications while only slightly lowering performance. The proposed techniques can be extended to existing applications in imaging [5] and radar [6], and to additional scenarios in which unseen secondary sources of information might appear.

Two novel projection design strategies were introduced and tested against an existing method to determine which approach offers superior convergence and therefore could be more advantageous in low SWAP applications. Since all algorithms performed similarly, the simple gradient ascent approach proposed in [5] is likely to be the best choice.

ACKNOWLEDGEMENTS

This work was supported by the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing. Gratitude is extended to Dr Domenico Gaglione, Dr Christos Ilioudis, and Dr Carmine Clemente for capturing the radar data.

REFERENCES

[1] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[2] J. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1395–1408, July 2009.

[3] D. P. Palomar and S. Verdú, "Gradient of mutual information in linear vector Gaussian channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 141–154, Jan. 2006.

[4] L. Wang *et al.*, "Information-theoretic compressive measurement design," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1150–1164, June 2017.

[5] F. K. Coutts, J. Thompson, and B. Mulgrew, "Gradient of mutual information in linear vector Gaussian channels in the presence of input noise," in *Proc. 28th Eur. Signal Process. Conf.*, 2021, pp. 2264–2268.

[6] ——, "Information-theoretic compressive measurement design for micro-Doppler signatures," in *Proc. Sens. Signal Process. Defence*, 2020, pp. 1–5.

[7] ——, "Learning a secondary source from compressive measurements for adaptive projection design," in *Proc. Sens. Signal Process. Defence*, 2021, pp. 1–5.

[8] B. Paul, C. D. Chapman, A. R. Chiriyath, and D. W. Bliss, "Bridging mixture model estimation and information bounds using I-MMSE," *IEEE Trans. Signal Process.*, vol. 65, no. 18, pp. 4821–4832, Sept. 2017.

[9] G. Yu and G. Sapiro, "Statistical compressed sensing of Gaussian mixture models," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 5842–5858, Dec. 2011.

[10] J. Yang *et al.*, "Compressive sensing by learning a Gaussian mixture model from measurements," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 106–119, Jan 2015.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[12] C. Xiao, Y. R. Zheng, and Z. Ding, "Globally optimal linear precoders for finite alphabet signals over complex vector Gaussian channels," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3301–3314, July 2011.

[13] V. C. Chen, *The micro-Doppler effect in radar*, 1st ed. Norwood, MA: Artech House, 2011.

LoRaWAN Performance Evaluation and Resilience under Jamming Attacks

Vaia Kalokidou, Manish Nair and Mark A. Beach
Communication Systems and Networks Group
University of Bristol, UK,
{Vaia.Kalokidou, Manish.Nair, M.A.Beach}@bristol.ac.uk

***Abstract*— There is an increasing deployment of Internet-of-Things (IoT) networks, from smart meters and smart lighting to humidity soil sensors and medical wearable devices. Long Range (LoRa) is one such over-the-air (OTA) transmission IoT standard, having a wide range of applications in smart cities, agriculture and health. It facilitates the inter-connection of services and smooth exchange of information. However, owing to its wireless interface, it is susceptible, as all wireless networks are, to OTA attacks. In this paper, we initially obtain the Bit Error Rate (BER) and Packet Error Rate (PER) of LoRa, in order to investigate the impact of continuous and reactive jamming attacks on it. We show that overall, LoRa can achieve a good performance even under a jamming attack, subject to parameters such as the transmit power, the Spreading Factor (SF) and the Coding Rate (CR). Moreover, it is proven that the impact on BER and PER is similar irrespective of whether the attack occurs with total frame synchronization or is synchronized to after the preamble transmission. Lastly, we apply a detection scheme, based on previous values of Received Signal Strength Indicator (RSSI) and PER to successfully identify malicious attacks.**

Keywords—LoRa, LoRaWAN, PHY Security, Jamming.

I. INTRODUCTION

There is a wide deployment of Internet of Things (IoT) networks in smart cities/buildings, healthcare, and industrial applications. However, wireless networks in general are susceptible to cyber-attacks. Therefore, it is crucial to “build” secure and agile future networks by developing detection and defense mechanisms.

A well-known IoT technology is the Long Range (LoRa) standard, developed by Semtech. It has wide ranging use cases such as smart parking, waste management, smart meters, lighting, agriculture, healthcare, smart industrial control, supply chain and logistics [1]. In the UK, The Things Network (TTN) has been initially deployed in Cambridge and is expanding elsewhere. TTN is based on Long Range Wide Area Network (LoRaWAN) [2], a Low Power Wide Area Network (LPWAN) technology that operates on top of the proprietary LoRa protocol stack (originally developed to connect battery and low-power devices wirelessly to the internet) [2]. It constitutes a STAR network topology that uses gateway devices for receiving data from nodes and forwarding it onto LoRaWAN servers [3]. LoRaWAN allows geographically spread devices connectivity, securing bi-directional communication, mobility, and localisation services, and provides open-source software for hardware gateways and backend services [4].

Table I. LoRa Specifications (Europe).

Parameter	Values (approx.)
Frequency	868-870 MHz
Bandwidth	(UL) 125/250 kHz
	(DL) 125 kHz
EIRP	max 20dBm
Link Budget	155 dB
Spreading Factor	7-12
Data Rate	250bps – 50kbps
Battery Life	106 months (2000mAh)
Coverage	(urban) up to 5km
	(rural) up to 15km

LoRa features low-power operations, long range communications and low data rates. Table I provides an overview of LoRa specifications in Europe. Ten channels are defined in total, with eight having multi data-rate of 250bps-5.5Kbps, a single channel with high data rate (11Kbps), and a single Frequency Shift Keying (FSK) channel at 50kbps [3]. As LoRa is an over-the-air (OTA) transmission standard, it is susceptible to cyber-attacks. There are two levels of security in LoRa: (a) network level security (authentication of node, providing integrity between the device and the network server - NwkSKey), and (b) application layer security (confidentiality with end-to-end encryption between the device and the application server - AppSKey) [4]. Most important identified LoRa vulnerabilities are related to the encryption keys, which are the key to attack the network once compromised [3,4].

State-of-the-art research has shown that additional security can be attained by employing physical layer (PHY) security. In general, PHY security entails: information-theoretic security, artificial noise aided security, security-oriented beamforming techniques, diversity-assisted security approaches, and physical-layer secret key generation [1,2]. The latter has gained a lot of attention in the LoRa standard. In [3], authors investigate the employment of different algorithms based on PHY key generation to a LoRaWAN network, looking at both static and mobile scenarios, achieving 13Mbit/s and 21Mbits/s key establishment rates. Moreover, [4] presents indoor and outdoor LoRa network experiments on secure key generation achieving higher key establishment rate of 31Mbits/s in mobile scenarios. In [5], the authors show that wireless key refreshment is feasible even in cases where an eavesdropper is close to the legitimate

978-1-6654-8348-3/22/$31.00 ©2022 IEEE

Back to Contents

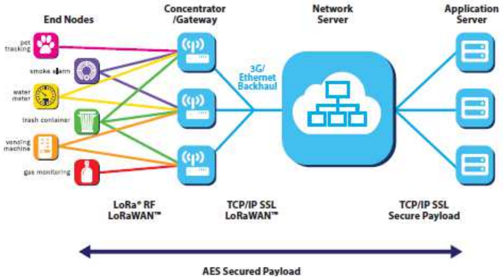


Fig. 1 LoRaWAN Architecture [6].

nodes. Interestingly enough, [3] presents a Machine Learning approach on generating security keys by converting wireless signals into structured datasets. In [4], PHY key generation is employed to LoRaWAN by using differential equations to achieve a great degree of randomness.

In this paper, we aim to initially evaluate the performance of LoRa, by building a LoRa-like Matlab simulator. Performance results are benchmarked to published results [5] to ensure the correct operation of our simulator. Then, we investigate the impact of various jamming attacks on the performance for different Spreading Factors (SF) and Coding Rates (CR). A detection mechanism is then applied, based on setting a threshold, related to Packet Error Rate (PER) and Received Signal Strength Indicator (RSSI), that provides the LoRa-like simulator with the opportunity to correctly identify a potential threat, i.e., jamming attack.

This paper is organised as follows: Section II presents the generic LoRa architecture, the PHY and the frame format, as well as the working specifications of the LoRa simulator developed in the University of Bristol. Section III gives an overview of performance results, starting from mean Bit Error Rate (BER) and PER under normal operation, and then analysing the performance impact of different jamming attacks. Finally, Section IV discusses the results of our research along with recommendations for future work.

II. LoRa PHY

A. Architecture

In a LoRa-LoRaWAN network, as depicted in Fig. 1, the end nodes, for e.g., smart meters, communicate with the gateways via the LoRa PHY. The gateways are connected to the network server via 3G/Backhaul Ethernet, and the network server communicates with the application server based on the TCP/IP SSL secure payload. Our focus in this paper falls on the connectivity between the end-nodes and the gateways, as we investigate LoRaWAN from the PHY layer perspective (LoRa).

LoRaWAN uses three different classes of devices to trade off network downlink (DL) communication latency against battery duration and optimise performance [3]. Class A entails bi-directional end-devices, whose UL transmission is followed by two short DL receive windows [3], based on ALOHA-type of protocol. This is the lowest required power class for applications that only require DL communication from the server shortly after the UL transmission. Class B comprises of bi-directional end-devices that require scheduled receive slots, allowing the server to identify active end-devices that are listening. Finally, bi-directional end-devices with maximal receive slots fall into the Class C category, with devices almost constantly opening receive windows [3]. In this Section, we focus on the LoRa PHY standard, discussing the frame format, the encoding and decoding process, and the modulation/demodulation employed by the standard. Since LoRa is a proprietary standard, the description of LoRa architecture and operation is based on research papers, online available material and reverse engineering results.

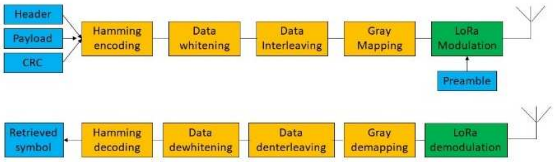


Fig. 2 LoRa PHY.

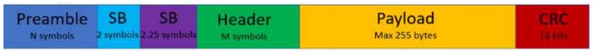


Fig. 3. LoRa frame format.

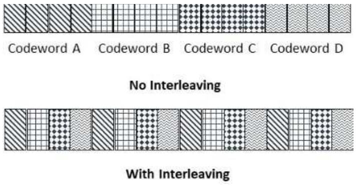


Fig. 4. Example of Bit Interleaving.

Overall, the LoRa PHY architecture is depicted in Fig. 2. It should be noted that some sources [6] define that data-whitening proceeds Hamming encoding. As shown in Fig. 2, there are four distinct operations comprising the LoRa encoding: (a) Hamming encoding, which adds parity bits, (b) data-whitening, which provides de-correlation of data, removing DC-bias in the transmitted data, (c) bit-interleaving, which scrambles bits to provide better immunity to burst errors (fading), and (d) gray-mapping, which reduces errors in adjacent bits by making adjacent symbols in the original representation only differ by one bit in the gray representation [6].

Encoding is followed by modulation. The LoRa standard uses Chirp Spread Spectrum (CSS) modulation. CSS modulation uses wideband frequency modulated chirp pulses to encode data. A chirp refers to a sinusoidal signal that increases/decreases in frequency over time.

The input symbol is spread on different frequencies and different time instances. The value of the SF, which takes values from 7 to 12, denotes the number of raw bits that can be encoded by the symbol and all the possible chip values (2^{SF}). The number of samples for every input symbol is given by the sampling frequency divided by the symbol rate, and for each sample the symbol value is cyclically shifted. To encode a LoRa symbol S in a chirp, a starting offset is added to the frequency sweep. The starting offset is given by [6]:

$$f_{offset} = \left({}^{Bwth}/_{2^{SF}}\right) S, \text{ where } \quad S \in [0, 2^{SF} - 1]. \qquad (1)$$

The bandwidth is restricted to $\left(f_c - {}^{Bwth}/_2, f_c + {}^{Bwth}/_2\right)$, and thus, the instantaneous frequency is linearly increased to the maximum frequency ($f_c + {}^{Bwth}/_2$), and then wrapped to the minimum frequency ($f_c - {}^{Bwth}/_2$). The instantaneous

Back to Contents

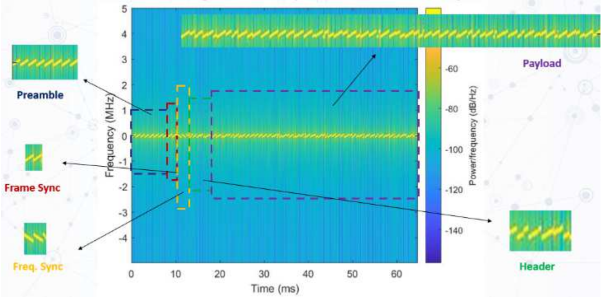


Fig 5. LoRa I/Q symbols for SF=7 and CR=4.

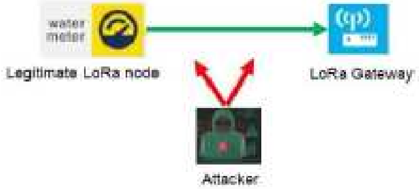


Fig. 6: LoRa network with one jammer attempting to attack the network.

frequency of the LoRa symbol S at time t, with $t \in [0\ T_s]$ and T_s denoting the symbol period, is given by [10]:

$$f_s(t) = S\frac{Bwth}{2^{SF}} + \mu\frac{Bwth}{T_s}t\ (mod\ Bwth), \qquad (2)$$

where μ defines if we have an upchirp ($\mu = 1$) or a downchirp ($\mu = -1$).

Demodulation and extracting symbols in a LoRa packet requires: (a) channelising and resampling the signal to the chirp bandwidth, (b) de-chirping with a locally generated signal, (c) taking the Fast Fourier Transform (FFT) of the de-chirped signals (where the number of FFT bins equals the spreading factor), and (d) extracting the maximum value from each FFT to obtain the symbol. Accurate synchronisation on the Start Frame Delimiter (SFD) is essential for demodulation. This is because incorrect synchronisation can spread the symbol energy between adjacent FFTs, resulting in incorrect demodulation. Lastly, the receiver performs synchronisation and frequency-offset estimation and compensation prior to demodulation. More details on the operation of the aforementioned blocks are given in Section II.B, with regards to the LoRa simulator developed in Matlab. Lastly, Fig. 3 depicts the LoRa frame format.

B. LoRa-Like Simulator

A LoRa-like simulator is developed using Matlab, partially based on the work presented in [5]. The frame consists of 8 symbols in the preamble, 2 symbols in the frame synchronisation field and 2.25 symbols in the frequency synchronisation field, 7 symbols in the header, variable length payload filed (depending on the simulation), and a 2-byte CRC field.

1) LoRa Encoding

The input data is randomly generated in binary format and converted to decimal (and back) depending on the stage of the encoding:

a) Hamming Encoding: Hamming codes (HC) belongs to the family of cyclic redundancy codes that check the integrity of the received message. A hamming encoder adds a number of parity bits that helps to detect and/or correct errors at the receiver during decoding. In LoRa, four CRs are available: a) 4/5 (simple parity check), b) 4/6 (shortened HC), c) 4/7 (common HC), and d) 4/8 (extended HC), with the first two CRs providing only error detection and the last two able to support error correction as well.

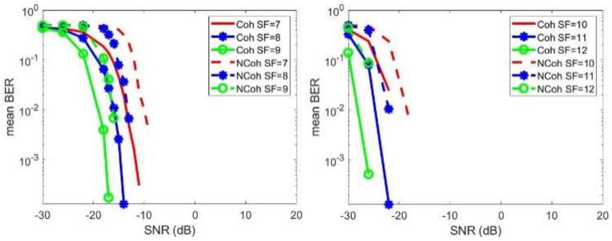


Fig. 7: Mean BER for CR=1 and (left) SF=7,8,9, (right) SF=10,11,12.

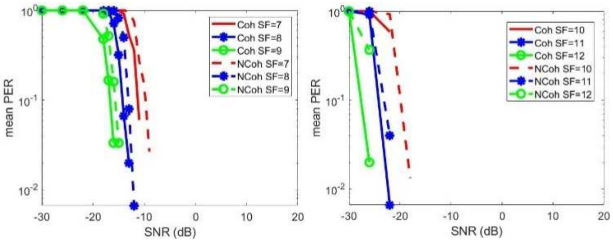


Fig. 8: Mean PER for CR=1 and (left) SF=7,8,9, and (right) SF=10,11,12.

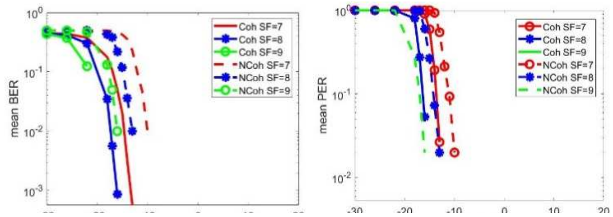


Fig. 9: Performance for SF=7,8,9 and CR=4 and (left) BER, (right) PER.

b) Data Whitening: During the data whitening, the transmitter XORs the transmit frame with a pseudorandom sequence, and the receiver XORs the received frame with the same sequence. Randomising data in this way attains receiver synchronisation similar to Manchester coding. However, unlike Manchester coding, it provides the advantage of keeping the same data rate at the cost of not having the guarantee of removing any DC-bias albeit with a very high probability of removing it [6].

c) Bit-Interleaving: Interleaving is a very-well known process in communications systems. The aim is to spread the bits comprising a codeword between multiple symbols. There are several ways of scrambling data during interleaving. Most sources in LoRa are not specific on the kind of interleaving employed. In our simulator, we perform simple interleaving by taking the transpose of the original data whitened matrix and mixing bits as shown in the Fig. 4. Reverse engineering work performed claims to have identified a special way of interleaving data in LoRa, based on using diagonals to scramble the bits [6].

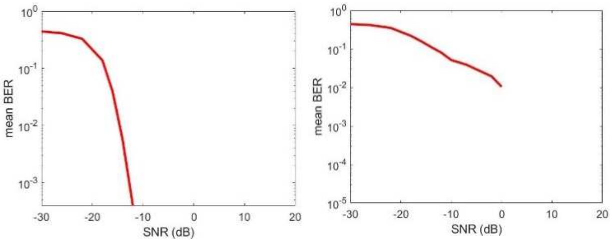


Fig. 10. BER (left) no jamming, and (right) with CW jamming.

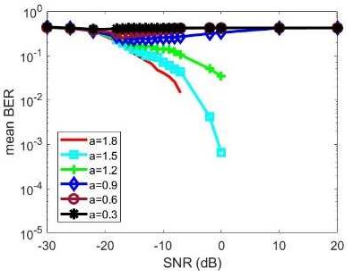


Fig. 11. Reactive jamming: Mean BERs for SF=7 and CR=1

d) Gray Mapping: In general, gray-mapping entails the mapping between a symbol, in any numeric representation, to a binary sequence. The input to the gray-mapper is XOR'd with a shifted version of itself. The Gray code we apply is given by $C_n = B_n\, XOR\, (B_n \gg 1)$ where B_n is the left most significant bit binary representation of n. On top of the mapping, a shift of -1 is used. At the receiver, a reverse to the encoding process is applied in order to retrieve the original symbols.

2) LoRa Modulation/Demodulation

The input to the modulator is a vector containing decimal values from $[0, \cdots, 2^{SF} - 1]$. The modulation process follows the steps defined in Section IIB. For every symbol (decimal value), there are N_s samples. Once the instantaneous frequency is chosen (2), the instantaneous phase of the LoRa symbol S at time t $(t \in [0\, Ts])$ is calculated:

$$\theta_s(t) = 2\pi f_s(t) t \tag{3}$$

Lastly, the complex LoRa symbol at time t $(t \in [0\, Ts])$ is given by:

$$s(t) = cos\theta_s(t) + jsin\theta_s(t) \tag{4}$$

At the demodulator, a default sequence of all zeros is CSS modulated and multiplied by the received sequence, separately for the preamble, and separately for the header/payload field. Then, having a choice between non-coherent and coherent detection, the data is demodulated. In the case of non-coherent detection, the maximum of the FFT window is taken. When coherent detection is active, then the resulting data is convolved with an ideal FSK signal, and the maximum real value is chosen. As shown in Section III, coherent detection offers a better performance. The I/Q LoRa, for the case of SF=7 and CR=4, are depicted in Fig. 5.

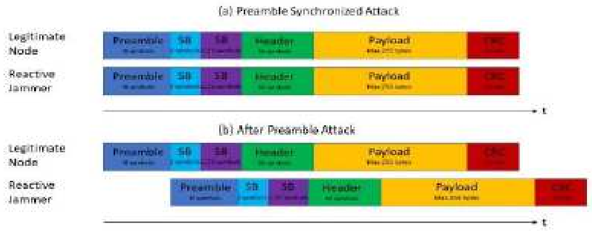


Fig. 12. Schematic of a preamble related attack.

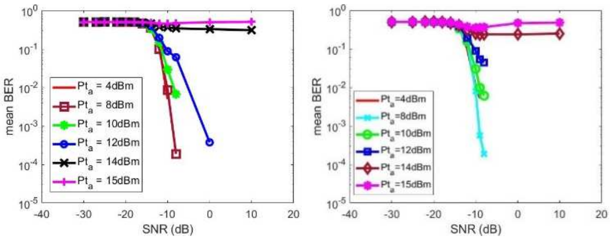


Fig. 13. Jamming Preamble BER (SF=7, CR=1), (left) total sync, (right) attack after preamble.

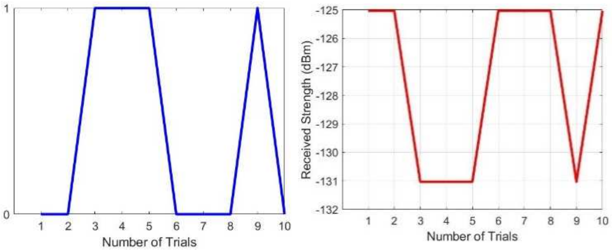


Fig. 14. Reactive jamming detection with RSSI and PER threshold. A set of attack-sessions denoted by 1 (left) mixed with no attack-sessions denoted by 0 (left) are simulated.

3) LoRa Cyclic Redundancy Code (CRC)

CRC is available only at the UL and has a size of 2 bytes. It belongs to the family of block codes and is applied to detect changes (errors) to the transmitted data. It entails a binary division of the actual data by a predetermined divisor, generated using polynomials. Based on [6], the polynomial used in LoRa is given by

$$x^{16} + x^{12} + x^5 + 1. \tag{5}$$

Moreover, findings in [3] show that the CRC bytes in the payload are not taken into account in the CRC calculation, but they are used as the final XOR value.

III. LoRa System Performance

LoRaWAN performance results are captured from several Matlab simulations in terms of the Bit Error Rate (BER) and Packet Error Rate (PER). The length of the payload is set to 17 bytes with all other fields in the frame having a pre-fixed length according to simulator definition parameters given in Section IIB. We consider one LoRa sensor transmitting to a gateway, and one jammer attempting to intrude the network, as depicted in Fig. 6. Unless stated otherwise, the transmit power at the legitimate node is set to 12dBm.

A. LoRa General Performance

We consider transmission over an Additive White Gaussian Noise (AWGN) channel

$$y(t) = s(t) + z(t) \tag{6}$$

where $y(t)$ is the received signal at the gateway, $s(t)$ is the CSS modulated LoRa signal transmitted by the LoRa sensor, and $z(t)$ represents the AWGN, with $z \in \mathcal{CN}(0, \sigma^2)$. We consider both coherent and non-coherent detection. Fig. 7 (left) presents the BER for a rate code ($RC = \frac{CR}{CR+1}$, where CR

is the coding rate) of 4/5 (i.e., CR=1), and all available SFs, i.e., SF=7-12. For the same specifications, Fig. 7 (right) depicts the respective PER. Fig. 8 depicts the mean BER and PER for a coding rate 4/8 (i.e., CR=4) and SF=7,8,9. Overall, it can be observed that the higher the SF, the better the BER and PER. This is because higher SFs attain higher symbol energy. Moreover, as we switch from CR=1 to CR=4, an improved BER performance can be observed, as expected. It should be noted that our results are aligned with published results in [5], thus validating the accuracy of our LoRa-like simulator, using similar parameters. It can be observed that Fig. 8 (left) attains similar PER as in Fig. 9 for SF=7,8,9 as in [5].

B. LoRa Performance Under Attack

There are various types of attacks that can be anticipated in a LoRa network. Investigation is performed on two types of jamming: (a) continuous jamming, where the jammer continuously transmits independently of whether a legitimate transmission takes place or not, and (b) reactive jamming, where the jammer attempts an attack only when they sense a legitimate transmission [2]. Initially, the case of having a continuous jammer is simulated. Assuming that the attacker transmits an asynchronous continuous wave (CW) signal at 868MHz over an AWGN channel for SF=7 and CR=1, the degradation in performance is depicted in Fig. 10. As compared to the case without jamming, for asynchronous jamming, a much higher SNR is required to maintain the same BER. For example, whilst a mean BER of 10^{-3} is attained at an SNR of -10dB under normal LoRa operation (Fig. 10 left), at similar SNRs, the BER degrades by two orders of magnitude to 10^{-1}) under asynchronous CW jamming (Fig. 10 right). Moreover, considering reactive jamming, the received signal at the gateway is given by

$$y(t) = s_l(t) + s_\alpha(t) + z(t) \tag{8}$$

where $y(t)$ is the received signal at the gateway, $s_l(t)$ is the CSS modulated LoRa signal transmitted by the legitimate LoRa sensor, $s_\alpha(t)$ is the CSS modulated LoRa signal transmitted by the attacking node and $z(t)$ represents the AWGN. For SF=7, Fig. 11 shows the mean BER for the case of CR=1. The ratio of the legitimate node's transmit power over the power of the attacker is denoted by α. It can be observed that for $\alpha < 0.9$, the system breaks, i.e., packets cannot be transmitted correctly.

Lastly, we consider the case that a reactive jamming attack is performed either in total frame synchronisation between the attacker and the legitimate node, or right after the end of the preamble transmission from the legitimate node's end, as described in Fig. 12. For SF=7 and CR=1, the comparison between the two cases is depicted in Fig. 13. Taking $Pt\alpha$ as the transmit power of the attacker varying from 4dBm to 15dBm, with the legitimate node having a transmit power of 12dBm, we can observe that when the attacker transmits at 13dBm or lower, a good BER can be achieved. Furthermore, no major difference, on the performance, is observed if there is no total synchronisation between the transmissions of the attacking and the legitimate node.

C. LoRa Detection of Attacks

One of the most popular detection mechanisms against cyber-threats is the establishment of a threshold, typically related to RSSI and PER, based on their values from previous observations. This method is particularly suitable for networks in environments that are highly static or with slow changes (e.g., static sensor in a rural area) where severe changes in the environment are not anticipated allowing the setting of a threshold to detect any threats on the network.

We have chosen to set both an RSSI and a PER threshold. For SF=7 and CR=1, the values of the thresholds, based on previous observations (i.e., extensive simulations), were taken as -126dBm for the RSSI case, and 0.001 for the PER case. Again, for a payload length of 17 bytes and reactive jamming on the network, a set of attack-sessions (denoted by 1) mixed with no attack-sessions (denoted by 0) are simulated (10 trials overall) to observe if attacks can be identified on both metrics. The sequence of events was 0011100010. As shown in Fig. 14, attacks were correctly detected. Moreover, for each event the corresponding RSSI value is depicted.

IV. Conclusions

In this paper, we modeled transmissions between LoRa nodes and gateways. BER/PER under normal operation is assessed. Multiple jamming attacks were performed to study the networks' performance under their impact. Asynchronous continuous jamming had an impact on the performance, however, transmissions were still possible. It was shown that the performance variation between attacking the network with total synchronisation and attacking it after the preamble transmission was not substantial. In the case of reactive jamming, if the transmit power of the jammer was not considerably higher than that of the legitimate node, good BERs were attained. An RSSI and PER threshold were employed to successfully detect any possible threats. For future investigation, we propose using RSSI values to 'train' the LoRa network and apply PHY key generation exchange between the legitimate nodes to secure the network against malicious attacks

Acknowledgment

This research is funded through the UKRI/EPSRC Prosperity Partnership in Secure Wireless Agile Networks (SWAN), (EP/T005572/1), https://www.swan-partnership.ac.uk/

References

[1] E. Aras, et. al., "Exploring The Security Vulnerabilities of LoRa", IEEE Conference on Cybernetics, June 2017

[2] Y. Zen, X. Wang, L. Hanzo, "A Survey on Wireless Security: Technical Challenges, Recent Advances and Future Trends", Proceedings of the IEEE, vol. 104, no. 9, pp. 1727-1765, Sept. 2016.

[3] T.M. Hoang, et. Al., "Physical Layer Security: Detection of Active Eavesdropping Attacks by Support Vector Machines" IEEE Access, vol. 9, pp. 31595-31607, Feb. 2021.

[4] J. Zhang, A. Marshall, L. Hanzo, "Channel-Envelope Differencing Eliminates Secret Key Correlation: LoRa-Based Key Generation in Low Power Wide Area Networks", IEEE Transactions on Vehicular Technology, vol. 67, no. 12, pp. 12462-12466, Dec. 2018.

[5] [B. Al Homssi, et. Al., "IoT Network Design using Open-Source LoRa Coverage Emulator", IEEE Access, vol. 9, pp. 53636-53646, April 2021.

[6] T, Joachim, "Complete Reverse Engineering of LoRa", EPFL, Telecommunications Circuits Laboratory, Lausanne.

[7] M. Chiani, A. Elzanaty, "On the LoRa Modulation for IoT: Waveform Properties and Spectral Analysis", IEEE Journal on Internet of Things, vol. 6, no. 5, pp. 8463-8470, May 201

Back to Contents

Author Index

Back to Contents

SSPD 2023

Sensor Signal Processing for Defence Conference

Important Dates:

Submission of Papers: 16th April 2023

Notification of Paper Acceptance: 30th June 2023

Final version of Paper Due: 30th July 2023

Date of conference: 12 to 13 September 2023

Online / Royal College of Physicians Edinburgh

International Conference in Sensor Signal Processing for Defence: from Sensor to Decision

Signal Processing for Defence Conference is organised by the University Defence Research Collaboration (UDRC) in Signal Processing. SSPD 2023 aims to bring together researchers from academia, industry and government organisations interested in Signal Processing for Defence.

Papers are solicited from the following areas:-

- Array Signal Processing
- Image Processing
- Radar, Sonar and Acoustic
- Multimodal Signal Processing
- Multi-Target Tracking
- Signal Acquisition and Sensor Management
- Multiple-input and multiple-output (MIMO)
-Deep Learning, Machine Learning

- Information/Data Analysis
- Data Fusion
- Source Separation
- Anomaly Detection
- Distributed Signal Processing
- Low Size Weight & Power Solutions
- Target Detection and Identification
- Electro-Optic Sensing

All submitted papers will be peer reviewed. Technical sponsorship is provided by the IEEE Signal Processing Society and proceedings will be submitted to the Xplore Digital Library.

www.sspdconference.org

UDRC

[dstl] The Science Inside

SSPD Conference

UKRI Engineering and Physical Sciences Research Council

IEEE Signal Processing Society TECHNICAL CO-SPONSOR