

Wave parameter extraction from SAR data

Preliminary investigation into the development of a parameter extraction pipeline for sea ice characteristics from SAR



Ryan Jones
(JNSRYA006)

Department of Electrical Engineering
University of Cape Town
South Africa

Supervisor: Robyn Verrinder

October 2023

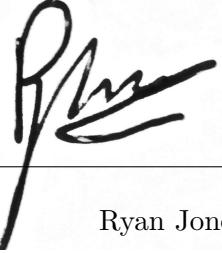
EEE4022S final year project report submitted in partial fulfilment of the requirements for the degree of **Bachelor of Science in Mechatronics** in the Department of Electrical Engineering at the University of Cape Town

Keywords: Synthetic Aperture Radar; Ocean Waves; Parameter Extraction; Marginal Ice Zone; Southern Ocean

Declaration

I, Ryan Jones, hereby:

1. grant the University of Cape Town free licence to reproduce the above thesis in whole or in part, for the purpose of research;
2. declare that:
 - (a) this thesis is my own unaided work, both in concept and execution, and apart from the normal guidance from my supervisor, I have received no assistance except as stated below:
 - (b) neither the substance nor any part of the above thesis has been submitted in the past, or is being, or is to be submitted for a degree at this University or at any other university, except as stated below.



Ryan Jones
Department of Electrical Engineering
University of Cape Town

Monday 30th October, 2023

Acknowledgements

I would like to sincerely thank my supervisor, Robyn Verrinder, for her continued support and guidance throughout this project. Her knowledge and expertise have been a great help in solving the problems faced in this project. She has given me an understanding, and an appreciation for ocean waves which I never expected to gain going into this project. I have really valued her approach to research and hope to be able to follow her advice moving forward. Secondly, my co-supervisor, Dr Stephen Paine has also been a great help throughout this project and has helped me to understand and appreciate the subject of radar, specifically SAR and for this I thank him.

Additionally, I would also like to thank Francesca De Santi, for helping me at various stages of my project through email correspondence. She was also kind enough to provide information when I began this project through Dylan White. Thank you to you both, for helping me get this project off the ground and pointing me in the right direction from the get-go.

I would not be here if it were not for my parents, Rob and Dawn. I would like to thank them for all the sacrifices they have made over the entirety of my educational journey as well as being my biggest supporters throughout. Without my brother, Glen, I would not have been able to constantly try to be the best version of myself - whether it be competitive, or not, you are still my biggest fan and I yours. So thank you for showing me how to be a good brother, and keeping me in line. Dad, thank you so much for all your proof reading of multiple chapters - I really appreciate it and it made the world of difference having another set of eyes look over everything. I love and appreciate you all, and this project is for you - particularly you, Mom.

Finally, I would like to thank all of my friends - new and old. They have all provided invaluable support to me throughout this degree. They have all been a source of motivation and have made me push myself to achieve and for that, I am eternally grateful. Tristyn, your continued support, encouragement, countless proofreads, late-night grafts, and sanity checks throughout the course of this project have been a huge factor in getting it to where it is as well as kept me sane. Thank you.

Abstract

Wave parameter extraction from SAR data

Ryan Jones

Monday 30th October, 2023

The composition and distribution of sea ice within the Marginal Ice Zone ([MIZ](#)) of Antarctica, along with sea ice-ocean interactions with the Southern Ocean ([SO](#)), play an important role in global climate. However, current sea ice and, wave attenuation and dispersion models of the Antarctic region significantly lag behind those of the Arctic region. The lack of year-round in-situ data of the [MIZ](#) necessitates improved data collection methods. This project aims to design and implement a wave parameter extraction pipeline using Synthetic Aperture Radar ([SAR](#)) data; a fundamental building block in the development of an Antarctic sea ice wave parameter pipeline. Whilst prior work has been done on wave and sea ice parameter extraction pipelines, the investigated literature inadequately defines the design of such pipelines. Although the literature on wave parameter extraction algorithms is well-defined and tested, the same cannot be said for wave attenuation and dispersion models in the Antarctic region. These models are defined and tested using data from the Arctic; a vastly different region with dissimilar sea ice-ocean interactions. Therefore, there is a need for the development of a sea ice parameter extraction pipeline tailored to the Antarctic region.

This report aims to provide insight into the design process undertaken, as well as the related results. This pipeline was developed in MATLAB and open-source data was used throughout this project from Sentinel-1A ([S1A](#)) and the National Oceanic and Atmospheric Administration ([NOAA](#)). The efficacy and accuracy of the pipeline were tested through two case studies surrounding the Cape Point region. Each case study investigated the reliability of the pipeline for different conditions. Regrettably, the pipeline had to be tested without ground-truth data. Instead, analysis was conducted using known relationships between significant wave height, period, and depth. The pipeline utilised the Hasselmann and Hasselmann ([HH](#)) procedure for wave parameter extraction from [SAR](#) data. The implementation of this procedure was successful, bar the maximisation of the [HH](#) cost function. This meant that the pipeline did not accurately extract significant wave height, although the relationship with depth was maintained. Significant wave period was correctly extracted, and the pipeline was found to have less than 5 % relative error for varying sea states.

Contents

Declaration	i
Acknowledgements	ii
Abstract	iii
Contents	vi
List of Figures	x
List of Tables	xi
Abbreviations	xii
Chapter 1 Introduction	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Project Objectives	2
1.4 Scope and Limitations	2
1.5 Report Outline	3
Chapter 2 Literature Review	4
2.1 Antarctic Climate	4
2.1.1 Antarctic Sea Ice	5
2.1.2 Marginal Ice Zone	7
2.1.3 Pancake Ice	8
2.2 Characterisation of Sea Ice	9
2.2.1 Microwave Radiometry and Scatterometry	9
2.2.2 Altimetry	10
2.2.3 Optical and Thermal Infrared Imaging	10
2.3 Sea Ice Characterisation using SAR	11
2.3.1 SAR Satellites	11
2.3.2 SAR Ocean Wave Inversion	13
2.3.3 Wave Propagation Model in Sea Ice	14
2.4 Literature Critique	16
Chapter 3 Theoretical Background	17
3.1 Ocean Waves	17
3.1.1 Description of Ocean Waves	17
3.1.2 Wave Spectra	18
3.1.3 Wave-number Spectra	20
3.1.4 Linear Wave Theory	21
3.1.5 Ocean Wave Dynamics	21
3.1.6 Wave Modelling for Idealised Cases	23
3.2 Synthetic Aperture Radar (SAR)	25
3.2.1 RAR vs. SAR	25
3.2.2 SAR Imaging Modes	27
3.2.3 Pre-processing Techniques	29

3.3	Hasselmann and Hasselmann Ocean Wave Inversion Technique	29
3.3.1	SAR Imaging of Ocean Waves	30
3.3.2	inversion technique	32
Chapter 4	Pipeline Design	33
4.1	Overview	33
4.2	Pre-processing	35
4.2.1	NetCDF Export	35
4.2.2	MATLAB Data Import	36
4.2.3	Data Transects	36
4.3	Metadata Extraction	38
4.3.1	Data Import	39
4.3.2	Filter Attributes	39
4.4	Wave Spectrum Generation	39
4.4.1	Wave Data Download	40
4.4.2	GRIB2 to NetCDF Conversion	40
4.4.3	Defining Location	41
4.4.4	Wave Parameter Extraction	42
4.4.5	One Dimensional Wave Spectrum	42
4.4.6	Two-dimensional Wave Spectrum	44
4.4.7	Wave-number Spectrum	45
4.5	SAR Spectra	46
4.6	SAR Spectrum of Ocean Waves	47
4.6.1	Additional Metadata Extraction	47
4.6.2	Modulation Transfer Functions (MTFs)	49
4.6.3	Co and Autocovariance Functions	49
4.6.4	Spectral Expansion	51
4.7	Inversion	52
4.7.1	Cost Function Calculation	52
4.7.2	Best-fit Wave Parameters	53
Chapter 5	Pipeline Verification Experiments	54
5.1	SAR Data Transects	54
5.2	Wave Spectra	55
5.2.1	Wave Evolution	55
5.2.2	Deep Water Approximation	57
5.2.3	Directional Distribution Function	57
5.2.4	Negative \mathbf{k} Wave-number Spectra	58
5.3	SAR Spectra of Ocean Waves	59
5.3.1	Power Spectral Expansions	59
5.4	Inversion	61
5.4.1	Single Iteration	62
5.4.2	Multiple Iterations	62
5.5	Summary	64
Chapter 6	Pipeline Case Study Testing	65
6.1	Testing Setup	65
6.2	Depth Investigation	66
6.2.1	Results	67
6.2.2	Discussion	68
6.3	Sea State Investigation	69
6.3.1	Discussion	71
Chapter 7	Conclusions	73

7.1	Recommendations for Future Work	74
7.1.1	Pre-processing	74
7.1.2	SAR Imaging of Ocean Waves	74
Chapter A	Additional Theory Definitions	75
A.1	Pre-processing Techniques	75
A.1.1	Speckle	75
A.1.2	Thermal Noise	76
A.1.3	Radiometric Calibration	76
A.2	Polarisation	77
A.2.1	Scattering	77
A.3	Hasselmann and Hasselmann (HH) Variable Definitions	78
Appendix B	SAR Spectrum Plots	79
B.1	Pipeline Verification	79
B.1.1	Power Spectral Expansions	79
B.1.2	Power Spectral Multiplication with Coefficient	80
Appendix C	Case Study Testing	83
C.1	October 15, 2023	84
C.1.1	Output Parameters	84
C.1.2	SAR Spectra Plots	86
C.1.3	Wave Spectra Plots	89
C.2	October 20, 2023	90
C.2.1	Output Parameters	91
C.2.2	SAR Spectra Plots	91
C.2.3	Wave Spectra Plots	95
Appendix D	Sentinel-1A Metadata Values	98
Appendix E	MATLAB Live Script	101

List of Figures

2.1	Comparison of sea ice extent in Antarctica between the winter of 2022 and summer of 2023. Taken from [1].	5
2.2	Interferometric Wide Swath (IW) mode coverage of one Sentinel-1 satellite after one 12-day orbit cycle. Taken from [2].	12
2.3	Antarctic coast line coverage and reconstructed continuous SAR image using Sentinel-1. Taken from [3].	13
3.1	Visual definition of surface elevation ($\eta(t)$), wave height (H), wave period (T), and zero-down crossing. Adapted from [4].	18
3.2	Graphical representation of Equations 3.28, 3.30, 3.29 which represent the Pierson-Moskowitz spectrum, peak enhancement function and the resultant JONSWAP spectrum respectively. Adapted from [4].	24
3.3	Graphical representation of a SLAR system with important features labelled. Adapted from [5, 6].	25
3.4	Graphical representation of commonly used SAR satellite capture and specific S1A capture modes.	28
4.1	Pipeline system design in the context of the entire parameter extraction process. The scope of this project is shown in black text, with parts of the pipeline outside of the scope, shown in grey.	34
4.2	Pipeline system design for the scope of this project broken down further than the entire pipeline shown in Figure 4.1. All the required metadata and external models are shown, along with the part of the process at which they are required.	34
4.3	Block diagram depicting an expanded pre-processing sub-block of Figure 4.2.	35
4.4	Implemented geometry for taking transects of full SAR scene.	37
4.5	S1A Intensity VV data, pre-processed according to Section 4.2, from Oct. 2, 2023 at 34.78S, 16.77E with transects at 30° for $x_1, y_1 = 1, 1$ displayed and labelled. Respective individual transects are shown.	38
4.6	Block diagram depicting an expanded metadata extraction sub-block of Figure 4.2.	38
4.7	Block diagram depicting an expanded wave spectrum generation sub-block of Figure 4.2.	40
4.8	One-dimensional JONSWAP wave spectra, $E(\omega)$, generated using NCEP wave data.	44
4.9	Directional distribution function, $D(\theta)$, defined using the $\cos^2(\theta)$ model detailed in Section 3.1.6.	44
4.10	Two-dimensional frequency-direction wave spectra, $E(\omega, \theta)$, generated using NCEP wave data, and the directional distribution function shown in Figure 4.9.	45
4.11	Two-dimensional wave-number spectra, $E(k_x, k_y)$, generated using NCEP wave data.	45
4.12	Observed SAR intensity spectrum plotted on a power spectrum scale from Oct. 20, 2023. Two transects are shown and the difference in spectra can be visually seen.	46

4.13	Block diagram depicting an expanded SAR spectrum generation sub-block of Figure 4.2	47
4.14	Block diagram depicting an expanded orbitalVelocityCovariance sub-block of Figure 4.13.	49
4.15	Block diagram depicting an expanded rarImageIntensityAutocovariance sub-block of Figure 4.13.	50
4.16	Block diagram depicting an expanded rarImageIntensityCovariance sub-block of Figure 4.13.	50
4.17	Generated SAR spectrum plotted using the SARSpectrumPlot function. The shown SAR spectrum is generated using the wave spectrum depicted in Figure 4.11.	52
4.18	Block diagram depicting the expanded inversion sub-block of Figure 4.2.	52
5.1	Geographic points of generated JONSWAP spectra using NCEP wave data. The points indicating each wave spectrum location correspond to the colours of the plots generated in Figure 5.2a. The contours shown in colour represent the respective depths below sea level: 25 m, 50 m, 100 m, 200 m, 250 m, 500 m, and 750 m. Bathymetric contour data sourced and adapted from OpenDem.	55
5.2	One-dimensional JONSWAP wave spectra, $E(\omega)$, which changed over different geographical locations surrounding Cape Point. Generated using NCEP wave data.	56
5.3	Plot generated using the nValidation function for varying depths using Equation 3.27.	57
5.4	Plot of the area underneath the directional distribution function, $D(\theta)$, over all θ . An area of 1 is marked on the plot to verify that the expected condition is satisfied.	58
5.5	Comparison of $E(\underline{k})$ and $E(-\underline{k})$, from Oct. 2, 2023, at 34,25° S, 18.25° E. Generated using NCEP wave data and the waveNumberSpectrum function.	58
5.6	Spectral expansions of both $P_{n,2n}^S$ and $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from Oct. 2, 2023, at 34.25° S, 18.25° E. Generated using NCEP wave data, the spectralExpansion2n , spectralExpansion2n_1 , and SARSpectrumPlot functions.	59
5.7	Plot of the coefficient, $(k_x \beta)$, used in Equation 3.55 which is multiplied by each power spectral expansion.	60
5.8	Plot of the coefficient, $(k_x \beta)^2$, used in Equation 3.55 which is multiplied by each power spectral expansion.	60
5.9	Spectral expansions of both $P_{n,2n}^S$ and $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a after being multiplied by the coefficient described in Equation 3.55.	61
5.10	Comparison of best-fit and first-guess one-dimensional wave spectra obtained using the HH minimisation technique for different numbers of iterations.	62
5.11	Comparison of best-fit wave parameters over each iteration of the HH minimisation technique compared to first-guess values obtained from NCEP wave data.	63
5.12	Cost function minimisation over 8 iterations.	63
6.1	Geographic points of the four case study testing locations. The contours shown in colour represent the respective depths below sea level: 25 m, 50 m, 100 m, 200 m, 250 m, 500 m, and 750 m. Bathymetric contour data sourced and adapted from OpenDem.	66
6.2	S1A SAR image over Cape Town acquired on October 20, 2023, at 18:56 UTC. Transects used in case study testing are highlighted and labelled.	67
6.3	Best-fit and first-guess wave spectra after 5 iterations for all transects.	68

6.4	S1A SAR image over Cape Town. Acquired on October 15, 2023 at 17:26 UTC, and October 20, 2023, at 18:56 UTC. Transects used in case study testing are highlighted and labelled.	70
6.5	Best-fit and first-guess wave spectra after 5 iterations for all transects.	70
A.1	Unprocessed GRD S1A data.	75
A.2	GRD S1A data with thermal noise removed using SNAP.	76
A.3	Radiometrically calibrated GRD S1A data done using SNAP.	77
A.4	Comparison of surface roughness and surface backscatter. Adapted from [6].	78
B.1	Spectral expansion of $P_{n,2n}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the <code>spectralExpansion2n</code> , and <code>SARspectrumPlot</code> functions.	79
B.2	Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the <code>spectralExpansion2n_1</code> , and <code>SARspectrumPlot</code> functions.	80
B.3	Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the <code>spectralExpansion2n_2</code> , and <code>SARspectrumPlot</code> functions.	80
B.4	Spectral expansion of $P_{n,2n}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x\beta)$	81
B.5	Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x\beta)^2$	81
B.6	Spectral expansion of $P_{n,2n-2}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x\beta)^0 = 1$	82
C.1	Observed S1A SAR spectra, first-guess JONSWAP modelled wave spectrum, calculated first-guess SAR spectrum, best-fit wave spectrum, and best-fit SAR spectrum for all three transect locations, with labelled rows, as depicted in Figure 6.4a, for Oct. 15, 2023.	83
C.2	Observed S1A SAR spectra, first-guess JONSWAP modelled wave spectrum, calculated first-guess SAR spectrum, best-fit wave spectrum, and best-fit SAR spectrum for all four transect locations, with labelled rows, as depicted in Figure 6.2, for Oct. 20, 2023.	84
C.3	Best-fit significant wave height per iteration for Oct. 20, 2023.	85
C.4	Best-fit significant wave period per iteration for Oct. 20, 2023.	85
C.5	Full resolution SAR data transects, from Figure 6.4a. These transects were used to calculate the SAR spectra in Figure C.7.	86
C.6	Full resolution SAR data transects, from Figure 6.4a. These transects were used to calculate the SAR spectra in Figure C.7.	86
C.7	Observed SAR intensity spectra of data transects, from Figure 6.4a.	87
C.8	Observed SAR intensity spectra of data transects, from Figure 6.4a, without a spectral threshold.	87
C.9	Full view of observed SAR intensity spectra of data transects, from Figure 6.4a, without a spectral threshold.	87
C.10	Generated HH SAR spectra of wave spectra, from Figure 6.4a.	88
C.11	Generated HH SAR spectra of wave spectra, from Figure 6.5, without a spectral threshold.	88
C.12	Best-fit SAR spectra.	88
C.13	Best-fit HH SAR spectra, without a spectral threshold.	89
C.14	Full view of best-fit SAR spectra, without a spectral threshold.	89

C.15 Best-fit and first-guess wave spectra after 5 iterations for all transects. Zoomed out version plots of Figure 6.5	89
C.16 Two-dimensional first-guess frequency-direction wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.	90
C.17 Two-dimensional first-guess wave-number wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.	90
C.18 Best-fit significant wave height per iteration for Oct. 20, 2023.	91
C.19 Best-fit significant wave period per iteration for Oct. 20, 2023.	91
C.20 Full resolution SAR data transects, from Figure 6.2. These transects were used to calculate the SAR spectra in Figure C.22.	92
C.21 Full resolution SAR data transects, from Figure 6.2. These transects were used to calculate the SAR spectra in Figure C.22.	92
C.22 Observed SAR intensity spectra of data transects, from Figure 6.2.	92
C.23 Observed SAR intensity spectra of data transects, from Figure 6.2, without a spectral threshold.	93
C.24 Full view of observed SAR intensity spectra of data transects, from Figure 6.2, without a spectral threshold.	93
C.25 Generated HH SAR spectra of wave spectra, from Figure 6.3.	93
C.26 Generated HH SAR spectra of wave spectra, from Figure 6.3, without a spectral threshold.	94
C.27 Full view of generated HH SAR spectra of wave spectra, from Figure 6.3, without a spectral threshold.	94
C.28 Best-fit SAR spectra.	94
C.29 Best-fit HH SAR spectra, without a spectral threshold.	95
C.30 Full view of best-fit SAR spectra, without a spectral threshold.	95
C.31 Best-fit and first-guess wave spectra after 5 iterations for all transects. Zoomed out version plots of Figure 6.3.	96
C.32 Two-dimensional first-guess frequency-direction wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.	96
C.33 Two-dimensional first-guess wave-number wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.	97

List of Tables

2.1	Commonly used SAR sensors for sea ice research and a comparison of their important features. Adapted from [7].	11
3.1	Comparison of ground swath width and resolution of S1A imaging modes. Adapted from [8].	29
4.1	Metadata values required for each individual MTF used in the SAR spectrum generation of ocean waves. Additional functions created are named.	48
5.1	<code>transectMSEStruct</code> values after running Listing 5.1.	54
5.2	Data points generated by the <code>createLatLonGrid</code> function not used to verify wave evolution over space.	56
5.3	m value for a non-linearity order of 1, and the associated implemented power spectrum used based on Equation 3.55.	60
5.4	RE after each iteration of the cost function.	64
6.1	Case study testing locations and associated depths.	65
6.2	First-guess wave parameters for locations of interest for October 20, 2023. Sourced from NCEP wave data.	67
6.3	Best-fit wave parameters for four defined transect locations for October 20, 2023. Generated using the pipeline mlx after 5 iterations.	68
6.4	First-guess wave parameters for locations of interest for October 15, 2023 and October 20, 2023. Sourced from NCEP wave data.	70
6.5	Best-fit wave parameters for three defined transect locations for October 15, 2023 and October 20, 2023. Generated using the pipeline mlx after 5 iterations.	71
C.1	Mean values of SWH and SWP for both sea states.	85
C.2	RE for all transects for October 15, 2023 and October 20, 2023 after 5 iterations.	85
C.3	RE for all transects for October 20, 2023 after 5 iterations.	91
D.1	S1A NetCDF exported metadata values. Attribute Names given can be used with filterAtt function.	100

Abbreviations

AABW Antarctic Bottom Water

BSWH Best-fit Significant Wave Height

BSWP Best-fit Significant Wave Period

CMIP Climate Model Intercomparison Project

CMIP5 Climate Model Intercomparison Phase 5

CSIR Council for Scientific and Industrial Research

EM Electromagnetic

ENVI ENvironment for Visualizing Images

ERS European Remote-Sensing Satellite

ESA European Space Agency

EW Extra Wide Swath

FSPW First-guess Significant Wave Period

FSWH First-guess Significant Wave Height

FFT Fast Fourier Transform

GMES Global Monitoring for Environment and Security

GRIB GRIdded Binary

GRD Ground Range Detected

HDF Hierarchical Data Format

HH Hasselmann and Hasselmann

IW Interferometric Wide Swath

IEEE Institute of Electrical and Electronics Engineers

JONSWAP JOint North Sea WAve Project

MARiS Marine and Antarctic Research Centre for Innovation and Sustainability

MSE Mean Squared Error

MIZ Marginal Ice Zone

MTF Modulation Transfer Function

NCEP National Centers for Environmental Prediction

- NetCDF** Network Common Data Form
- NOAA** National Oceanic and Atmospheric Administration
- RAR** Real Aperture Radar
- RCS** Radar Cross Section
- RE** Reliative Error
- S1A** Sentinel-1A
- SCALE** Southern oCean seAsonal Experiment
- SAR** Synthetic Aperture Radar
- SIC** Sea Ice Concentration
- SLAR** Side-looking Airborne Radar
- SO** Southern Ocean
- SNAP** SeNtinel Applications Platform
- SM** StripMap
- SWP** Significant Wave Period
- SWH** Significant Wave Height
- TOPSAR** Terrain Observation with Progressive Scan SAR
- UCT** University of Cape Town
- URL** Uniform Resource Locator

Chapter 1

Introduction

This project report outlines the development of a MATLAB pipeline designed for processing [SAR](#) data for parameter extraction. The intended application of this pipeline is within broader Antarctic research initiatives conducted by the Marine and Antarctic Research Centre for Innovation and Sustainability ([MARiS](#)) group at the University of Cape Town ([UCT](#)). Which will enable the extraction of sea ice characteristics from [SAR](#) data. This report encompasses an exploration into wave parameter extraction using [SAR](#). It introduces the pipeline's design and implementation for extracting the desired characteristics from these data. Furthermore, it covers the testing and validation procedures undertaken to ensure the pipeline's reliability and accuracy.

The primary goal of this introductory chapter is to provide contextual background for the project, framing its objectives, scope, limitations, and developmental stages.

1.1 Background

The contribution of Antarctica to the global climate is attributed to the distribution and composition of sea ice within the [MIZ](#), as well as the sea ice-ocean-atmospheric interactions with the [SO](#). Furthermore, the [SO](#) acts as a reservoir for heat and carbon dioxide, and is a crucial component in the global ocean overturning circulation [9, 10]. Thus, understanding how the sea state of the [SO](#), and by extension, the formation and development of sea ice in the [MIZ](#) impacts the Antarctic climate is vital in understanding global climatic conditions.

Significant research has been dedicated to examining sea ice extents in both the Arctic and Antarctic regions. However, the vastly different geographies of these regions means that models generated for Arctic regions are unsuitable for the Antarctic region [11]. This disparity can be attributed to two distinct features unique to Antarctic sea ice: sea ice-ocean interactions, and the presence of pancake ice.

Presently, determining pancake ice parameters relies solely on in-situ measurements. However, the harsh conditions of the [SO](#) and the Antarctic climate hinder in-situ data collection for a large majority of the year [12]. To mitigate this, research is currently being done into the application of remote sensing techniques to extract sea ice parameters, with a focus on pancake ice. The ability to extract year-round pancake ice parameters will enable the development of more accurate models for pancake ice, thereby enhancing wave attenuation and dispersion models for the [MIZ](#).

Interestingly, the composition of sea ice in the Arctic region is changing, and due to its decrease in sea ice extent, is subject to increased sea ice-ocean interactions. This has led to an increased presence of pancake ice in the Arctic [13, 14]. As a result, existing models for the region are

gradually decreasing in accuracy. Therefore, it is of utmost importance for both the Antarctic and Arctic regions to develop sea ice and ocean wave attenuation and dispersion models that can effectively characterise pancake ice and the associated sea ice-ocean interactions.

The need for improved pancake ice characterisation necessitates year-round parameter extraction. Fortunately the use of [SAR](#) for wave parameter extraction is a well-established field. However, the attenuation of these wave parameters is not fully modelled or understood. Consequently, there is currently no fully-fledged processing pipeline that extracts wave parameters and applies the desired wave attenuation and dispersion model. The development of this processing pipeline is vital, as it aids rigorous testing and validation of newly developed wave attenuation and dispersion models.

1.2 Problem Statement

There is currently a lack of accurate measurement data for the [MIZ](#), particularly during the winter season. This is primarily due to the remote and harsh conditions of the Antarctic continent and the surrounding [SO](#). Understanding sea ice in the [MIZ](#) is an essential aspect of investigating global climate as it plays a pivotal role in storing and transporting heat and greenhouse gases [10]. Due to the geographical and environmental differences between the Arctic and the Antarctic, the application of remote sensing techniques in the Arctic does not directly translate to the Antarctic. As a result, there has been poor characterisation of sea ice-ocean interactions, ice floe size, and shape – all of which are important factors when considering the impact of Antarctica on the global climate. To enhance the characterisation of these features remotely, parameter extraction algorithms need to be developed, analysed, and compared to a small set of in-situ measurements. This report aims to develop a pipeline for parameter extraction using [SAR](#) images, which will aid in understanding Antarctic sea ice characteristics, particularly the size, shape, and thickness of pancake ice in the [MIZ](#).

1.3 Project Objectives

The primary objective of this project was to develop a [SAR](#) processing pipeline for parameter extraction using wave modelling techniques. The pipeline's purpose was to be tested, analysed and well documented to aid future integration into a pipeline dedicated to sea ice parameter extraction. The pipeline was to be initially tested using [SAR](#) data with known first-guess values to establish its efficacy and reliability. This approach ensured the validation of the system's performance before being tested on [SAR](#) data captures from the [Southern oCean seAsonAL Experiment \(SCALE\) 2022](#) cruise.

1.4 Scope and Limitations

This project was severely limited by time constraints as only 13 weeks were allocated to the entire design of this pipeline. Consequently, the scope of the project was narrowed with respect to the broader research goals of this project.

As a result, this project formed as a preliminary investigation into developing a [SAR](#) processing pipeline for sea ice parameter extraction. Specifically, the project focused on a foundational

aspect of sea ice parameter extraction: wave parameter extraction. Consequently, the scope of the project centred around the development of a MATLAB pipeline which was able to extract two key wave parameters from [SAR](#) data - namely, significant wave height and significant wave period. The entire pipeline was required to be designed with a modular approach in mind, which necessitated rigorous testing and validation at various processing stages. These implemented validation methods, experiments, as well as the pipeline's design, must be well-documented and repeatable for future development and expansion of the parameter extraction pipeline.

As this project's focus was limited to wave parameter extraction from [SAR](#) data, investigation into the application of Antarctic sea ice dispersion and attenuation models for parameter extraction from [SAR](#) data fell outside of this project's scope. Whilst this project focused on wave parameter extraction, the broader context of sea ice parameter extraction needed to be understood. This context was essential in determining the pipeline's outputs.

Numerous limitations were associated with this project. Firstly, ground-truth data was unable to be obtained from the [CSIR](#) for analysis of the pipeline's accuracy and efficacy. This hindered an in-depth analysis of the pipeline's accuracy and efficacy. Secondly, the project relied solely on open-source data. This prevented the pipeline from achieving a spatial resolution better than 0.25 degrees. Lastly, both [S1A](#) and wave data, sourced from the [NOAA](#) National Centers for Environmental Prediction ([NCEP](#)), presented challenges with time and spatial coverage. [S1A](#) data was only available over the testing region every five days, whilst wave data was limited to a 10-day window, retroactively from the present date.

1.5 Report Outline

This project report begins with an overview of the relevant literature related to this project in Chapter [2](#), followed by the introduction and explanation of the relevant theory in Chapter [3](#). These two chapters introduce the pertinent subject matter of this report and provide context for the work undertaken in this project. Chapter [2](#) frames the context of this work by discussing previous research efforts and highlighting the necessity for this project. In contrast, Chapter [3](#) aims to provide a theoretical introduction to the themes of [SAR](#), along with the associated wave modelling and parameter extraction techniques. After framing the theoretical background of this project and its context, Chapter [4](#) outlines the approach taken to design the wave parameter extraction pipeline, and includes outputs from the sample data. These implemented methods in Chapter [4](#), were then verified in Chapter [5](#) using a variety of tests. After verification of the pipeline's implementation, two case studies were investigated in Chapter [6](#) of the pipeline's full implementation. Finally, Chapter [7](#) concluded the project, before introduction recommendations for future work. Appendices are provided thereafter, and contain additional plots and information used throughout the report.

Chapter 2

Literature Review

This chapter aims to introduce and discuss all the relevant literature associated with the subject matter of this project as well as focus on the areas discussed in the problem statement in Section 1.2. This literature review will begin by introducing the climate of the Antarctic continent and the SO, and discussing their importance to the global climate, as well as the ways in which these interact. As discussed in Section 1.2, the differences between the Antarctic and Arctic will be compared, and it will be evident from the differences between the two geographies, environment, and interactions with external features, that the two polar regions are vastly different. After exploring the distinctions and various types of ice in both regions, the way in which remote sensing can be used to characterise sea ice will be introduced.

The chapter examines various remote sensing techniques, however, the use of SAR for wave and sea ice parameter extraction is the focus of this report, and consequently, this literature review. This investigation details multiple algorithms and models as well as their advantages, and limitations when implemented in certain cases. Finally, this chapter concludes by critically evaluating the discussed literature and contextualising the necessity of this report.

2.1 Antarctic Climate

The Antarctic climate holds a unique position within the global context. During the winter, Antarctica's climate poses significant limitations, primarily due to the extensive reach of sea ice beyond the continent, rendering it inaccessible to ice-breaking ships [12]. Overcoming this limitation is crucial, as comprehending the Antarctic climate holds significance in understanding global climate variability and changes [10].

This section aims to provide a concise overview of Antarctica's significance in understanding global climate dynamics. It will subsequently introduce the influence of sea ice and the MIZ on the Antarctic climate. This discussion will encompass the MIZ, interactions between sea ice and the ocean, as well as SO dynamics that hold significance in the context of global climate changes and variations.

The contribution of Antarctica to the global climate is attributed to the Antarctic ice sheets, the SO, and the sea ice that blankets it. The Antarctic ice sheets have a significant impact on global sea levels, and this influence has escalated in recent years [10]. Predictions indicate that by the year 2100, the melting of Antarctic sea ice could lead to a global sea-level rise of up to 1 m [15]. The primary driver of this ice melt is basal melting¹ leading to the thinning of the ice

¹Basal melting results from the presence of warmer water beneath an ice sheet, intensifying melting from beneath the ice [16]. Lazeroms et al. have documented this phenomenon in Western Antarctica, where warmer seawater has intruded into shallower continental shelves beneath ice sheets and sea ice [17]

shelves [10].

The vast and unbounded SO plays a significant role in regulating the global climate. This is due to the fact that the SO acts as a reservoir for a substantial amount of anthropogenic heat², in addition to greenhouse gases - primarily carbon dioxide [10]. Additionally, the SO is a crucial component of the global ocean overturning circulation due to its unique underwater geography [19].

Sea ice within the SO plays a pivotal role in this overturning circulation as well as modulating heat exchanges [9]. It acts as a barrier between the strong westerly winds and the ocean surface, while also managing the density of the SO. The mechanisms underlying this heat storage and circulation are elaborated upon in the theoretical development chapter, whereas the interactions between sea ice and the SO are explored further in the subsequent sub-sections.

2.1.1 Antarctic Sea Ice

The extent of the sea ice in Antarctica varies significantly, between a seasonal maximum of 19 million km² during winter and a minimum of 3-4 million km² - a range of more than 1.5 times larger than the seasonal changes in the Arctic [19]. The seasonal sea ice extent is shown in Figure 2.1.

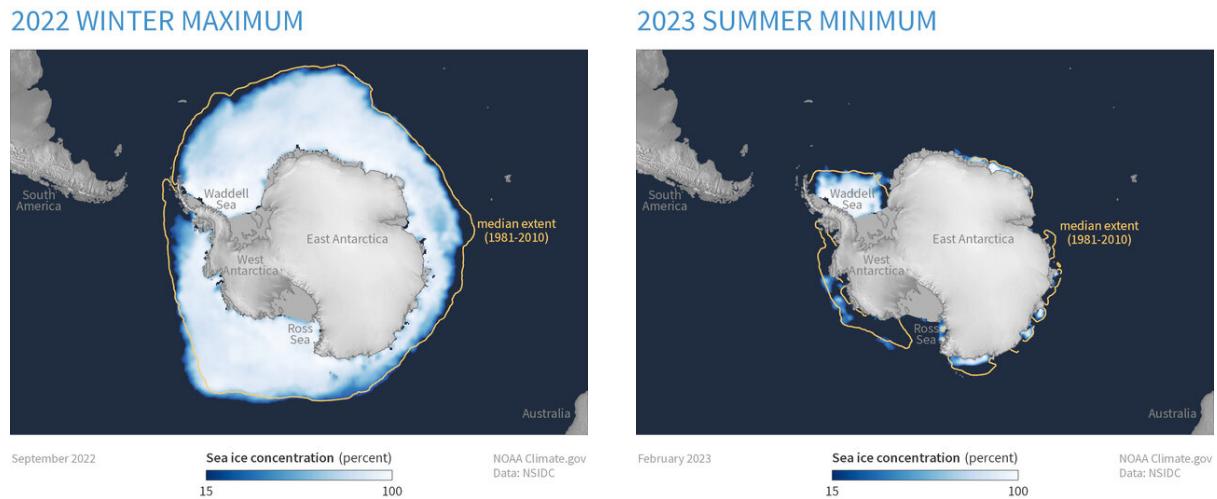


Figure 2.1: Comparison of sea ice extent in Antarctica between the winter of 2022 and summer of 2023. Taken from [1].

The maximum and minimum extents of sea ice can vary from year to year. Over the past 35 years, a slight increase in Antarctic sea ice extent each decade has been observed [11, 12, 20]. However, these observations contrast with predictive models used to estimate the Antarctic sea ice extent. Simmons [11] examined the application of Climate Model Intercomparison Phase 5 (CMIP5)³ models in Antarctica. Simmons found that these models perform well in analysing

²Anthropogenic heat can be defined as the excess heat generated through the direct consumption of energy by human activities. A comprehensive global database of anthropogenic heat has been constructed by Dong et al. [18].

³CMIP5, which forms part of the Climate Model Intercomparison Project (CMIP), serves as a standard experimental framework designed to examine the output of coupled atmosphere-ocean general circulation models. This approach allows for the evaluation of a climate model's strengths and weaknesses when analysing climate variability and change. An overview of CMIP5 can be found in the following paper by Taylor [21].

Arctic sea ice extent over a 35-year period. However, when applied to Antarctica, these models predict a decrease in sea ice extent, contrary to the observations made in [12, 20].

Simmonds suggests that the discrepancies between the model outputs of the Arctic and the Antarctic can be attributed to natural variability [11]. He also argues that, given the 35-year study period, natural variability has less impact on the incorrect simulation outputs. Instead, Simmonds proposes that the differences can be attributed to the physics of the models. This is supported by the fact that the CMIP5 models perform well in simulating Arctic ice but poorly in simulating Antarctic sea ice [11]. This idea confirms the work of Maksym et al. [12], as pancake ice, a significant component of the MIZ and a key factor in sea ice-ocean interactions, is not yet considered in these models. The importance of pancake ice is discussed in Section 2.1.3.

Maksym et al. [12] build upon the idea that the physics of sea ice models used in the Arctic may not be directly applicable to the Antarctic. They introduce the distinctions between these two polar regions, highlighting key differences. The main disparities can be attributed to geographical factors, sea ice growth and decay patterns, climate interactions, and ice-ocean dynamics. In terms of geography, Arctic sea ice is enclosed by the Eurasian and North American continents [12, 22], affording it greater protection compared to the more exposed Antarctic sea ice, which interacts with the SO. Consequently, Arctic sea ice is considered perennial⁴, whereas Antarctic sea ice is seasonal. As mentioned earlier, the extent of sea ice in Antarctica is 1.5 times greater than that of the Arctic [19]. Regarding climate interactions, the SO experiences strong westerly winds, resulting in substantial waves and frequent storms [12]. These prevailing westerly winds drive the sea ice northward⁵, expanding the sea ice coverage and consequently exposing it to warmer temperatures [12, 22]. The roughness and vastness of the SO account for the distinctions in ice-ocean interactions between Arctic and Antarctic sea ice [22], which constitutes a dynamic system characterised by constant motion [12].

From the literature above, it is evident that significant scientific endeavour is necessary to create accurate predictive models of Antarctic sea ice, rather than attempting to apply predictive models built for an Arctic environment. The severity of the Antarctic climate has resulted in a lack of field measurements, particularly during winter, regarding the spatial distribution of sea ice in Antarctica [12]. Both Gherardi et al. [23] and Kennicut et al. [10] agree that implementing improved remote sensing techniques is essential to monitor the extent of sea ice in Antarctica. Taking this further, Maksym et al. [12] hypothesise that the thickness of sea ice may be more responsive to climate variability than sea ice extent. This not only explains the increase in melt but also the discussed increase in extent. However, measuring ice thickness in Antarctica requires in-situ measurements instead of the remote sensing techniques employed in the Arctic [19, 23].

Maksym et al. [12] argue that the inability to evaluate models of Antarctic sea ice by researchers raises questions about the accuracy of projections. This is because models fail to capture the current state of sea ice in Antarctica. This highlights the necessity for developing a pipeline to extract sea ice parameters in Antarctica. Nonetheless, the aforementioned issues cannot be easily mitigated. Notably, the differences between Arctic and Antarctic models include the interactions between the MIZ and the SO, and pancake ice, which play a crucial role in sea ice-ocean interactions in Antarctica.

⁴Perennial ice refers to ice packs that persist for more than one year and do not melt during the summer months.

⁵This is due to the ice being pushed to the left of the westerly winds through a mechanism known as Ekman transport [12].

2.1.2 Marginal Ice Zone

The **MIZ** in Antarctica is defined as the area that is close enough to the **SO** to be affected by its presence [24]. This region experiences dynamic conditions due to the prevalence of strong westerly winds and storms in the ice-free zones of the **SO** [9, 12, 25], allowing for greater freedom of movement. The **MIZ** is a region where sea ice-ocean-atmosphere interactions are closely linked [24, 26]. As discussed, the climate of Antarctica and the **SO** is incredibly harsh. Sea ice within the Antarctic **MIZ** plays a crucial role in the global climate system [22], and comprehending the dynamics of the MIZ contributes to understanding the responses of Antarctic sea ice to climate change and variability, along with its associated impacts [25]. The **MIZ** is influenced by a variety of processes, which are introduced in the following subsection.

The **MIZ** plays a critical role in the global climate by forming a seasonal, insulating "skin" atop the **SO**. Due to the characteristics of sea ice, its high albedo reflects solar radiation and serves as an effective insulator [9, 22]. Consequently, sea ice within the **MIZ** helps regulate **SO** temperatures by limiting heat exchanges between the **SO** and the atmosphere [22]. Moreover, the presence of sea ice over the **SO** increases the salinity and density of the **SO** due to the seasonal formation of sea ice [9, 22]. This aspect is crucial for global ocean circulation [19]. The increased density beneath the sea ice in the **MIZ** leads to overturning and the formation of bottom water⁶ [22].

In addition to the discussions on sea ice-atmosphere and ocean interactions, the most crucial aspect of the **MIZ** in the context of this project is its interaction with wave motion from the **SO**, as well as the characteristics of the **MIZ** that regulate this process. Brouwer et al. [25] summarise this phenomenon by stating that wave-ice interactions are mutual. According to Brouwer et al., the waves propagating through the **MIZ** modify the properties of the sea ice. However, during this process, the wave amplitude is attenuated due to the energy required to alter the sea ice properties. The manner in which sea ice within the **MIZ** attenuates waves is comparable to that of a low-pass filter. This implies that waves with higher frequencies are attenuated, with the rate of attenuation depending on the physical characteristics of the sea ice, such as floe size, thickness, and concentration [29, 30].

Stammerjohn et al. [31] state that understanding sea ice-ocean interactions is crucial, as they influence sea ice features and distribution. These, in turn, affect both regional and polar climates, as well as the global climate. This impact has been observed in the Arctic **MIZ** by Asplin et al. [32] and Stopa et al. [33]. These researchers noted the penetration of long-period surface gravity waves⁷, reaching hundreds of kilometres into the Arctic **MIZ** before being completely attenuated [32, 33]. This observation holds significance not only for the Arctic but also for the Antarctic **MIZ**, where long-period and high-amplitude waves occur frequently due to the harshness of the **SO** [34]. It is imperative to understand how these waves penetrate and are attenuated within the Antarctic **MIZ** [35].

This understanding is required because the current definition of the **MIZ** relies on remote sensing techniques employing passive satellite technology to define the **MIZ** through the use of Sea Ice Concentration (**SIC**). The ice edge is conventionally defined as having 15 % **SIC**, while compact ice is characterized by 80 % **SIC** [36]. However, employing this metric to define the Antarctic **MIZ** is inefficient, as it overlooks wave interactions with the sea ice [25, 37]. According to

⁶The resulting Antarctic Bottom Water (**AABW**) is the coldest in the world and has far-reaching climatic impacts. More information on bottom water formation, as well as its importance to the global climate, can be found in [19, 27, 28].

⁷For an in-depth, quantitative explanation of ocean wave dynamics, including various types in shallow and deep waters, refer to Chapter 3 in this report. Additional information can be found in [4].

Vichi [37], the use of **SIC** to comprehend **MIZ** characteristics in the Arctic is effective; whilst, its application to the Antarctic **MIZ** lacks reliability. Vichi [26, 37] asserts that this discrepancy is attributed to heightened wave penetration and sea ice drift. In place of **SIC**, Vichi proposes an alternate approach to defining the **MIZ**. Vichi's recommendation involves utilising the statistical properties of **SIC** within the **MIZ**, alongside the spatial and temporal variability of **SIC** in the same zone [37]. This approach will enable the consideration of distinct ice types, along with wave activity from the **SO** [37].

2.1.3 Pancake Ice

As noted by Maksym et al. [12], one of the key processes in the Antarctic **MIZ** is pancake ice formation. Given the vastness of the **SO** and the prevalence of the **MIZ** in Antarctica, the existence of waves within the **MIZ** significantly influences the types of ice found within the **MIZ** [25]. This section provides a brief introduction to the formation and life cycle of pancake ice, followed by an exploration of its significance in shaping the Antarctic climate. Furthermore, this section introduces its crucial role in Antarctic sea ice modelling.

The formation of pancake ice in the **MIZ** is poorly understood [38]. In summer, large ice floes make up the **MIZ**, whereas, in winter the **MIZ** is made up of millions of smaller pancake ice floes [13, 39]. The presence of the rough sea surface means that the formation of ice is not allowed to settle into a fully formed ice sheet. Rather, when sea ice begins to form in the Antarctic in winter, it begins to create a suspension of crystals known as frazil ice [38, 40]. Over time, these frazil crystals converge, due to roughened seas, and collect to form small cakes [38, 41]. Initially, these small cakes are referred to as "dollar pancakes" [40], with a diameter of approximately 2-3 cm and a thickness of only a few millimetres [38, 40]. At greater distances from the ice edge in the **MIZ**, these dollar pancakes can fuse together due to wave attenuation, forming larger pancakes with a diameter of up to 5 m and a thickness of approximately 50 cm [38, 40, 42]. In 2019, Alberello et al. [13] found that approximately 50 % of the observed sea ice area was made up of pancake ice floes ranging from 2.3 - 4m in diameter. Only further into the **MIZ**, at approximately 270 km [43], where wave action has been attenuated even more, can these pancakes freeze together to form an ice sheet known as consolidated pancake ice. Once consolidated pancake ice is formed, thickening of these consolidated pancakes occurs to form pack ice [38].

Pancake ice formation is key to pack ice formation in Antarctica [44, 45]. As discussed in Section 2.1.2, the importance of sea ice is critical to the global climate. Understanding the processes behind the formation, evolution, and distribution of sea ice, particularly pancake ice, is vital for explaining the geophysical interactions in the polar regions, especially the **MIZ** [46].

Firstly, pancake ice plays a significant role in the production of a fertile winter ecosystem due to its ability to promote increased algal growth and provide a shelter for krill as its rafted bottom creates a large surface area [47]. Secondly, the gravity-induced drainage of brine after pancake formation helps regulate the salinity of the **SO**. Additionally, as previously discussed, the high albedo of sea ice serves as an insulating layer atop the **SO** and aids in regulating its temperatures [38].

The modelling of pancake ice is not particularly well understood and is currently being incorporated into existing sea ice models [13]. The understanding of pancake ice, particularly its impact on wave attenuation, is important, as indicated by both Doble et al. [14] and Alberello et al. [13]. The increased open water area in the Arctic is leading to a greater presence of pancake ice in the region [13, 14]. This is significant because the existing models for this region are

gradually decreasing in accuracy due to the changing composition of the sea ice in the region. In order to properly model the region, sea ice models including properly characterised pancake ice floes are required [14].

2.2 Characterisation of Sea Ice

This section will introduce the existing technologies and research that have been employed to characterise sea ice in both the Arctic and Antarctic regions. These methods will be briefly examined, focusing on their advantages and disadvantages in implementation. Subsequently, the technique of sea ice characterisation using **SAR** will be introduced in the subsequent section. The importance of ocean waves in modelling will also be discussed.

Whilst in-situ measurements are the most practical method for determining sea ice characteristics, as discussed in Section 2.1, a vast expanse of the Antarctic **MIZ** is inaccessible in winter due to the formation of thick sea ice that cannot be navigated, even by ice-breaking ships [12]. Thus, remote sensing techniques⁸ need to be implemented to monitor the Antarctic **MIZ** year-round due to the harshness of the region [10, 12]. Since the scope of this project revolves around **SAR** - a remote sensing technique in itself - this section will focus on remote sensing techniques for sea ice characterisation.

Remote sensing techniques rely on the Electromagnetic (**EM**) spectrum and satellite remote sensing of sea ice is conducted in multiple parts of the **EM** spectrum [7]. Different frequency ranges of the spectrum are used, and the reflected intensity measured by the sensors depends on the ability of sea ice, water, and snow to emit **EM** radiation. The ability of sea ice to emit this **EM** radiation is determined by its geophysical properties, such as salinity, roughness, and air content [7]. These geophysical properties can be inferred from the received intensity at the sensor. The **EM** radiation emitted back by sea ice can originate from either a passive source, such as the sun, or an active source, such as a **SAR** transmitter. The following sub-sections introduce three different remote sensing techniques which operate at different frequencies in the **EM** spectrum.

2.2.1 Microwave Radiometry and Scatterometry

For optimal sea ice microwave remote sensing, frequencies of less than 10 GHz are used. While higher frequencies result in better spatial resolution, the influence of atmospheric conditions increases. Thus, microwave remote sensing of sea ice always involves a trade-off between atmospheric influence and spatial resolution [7].

Microwave radiometry has been employed to measure sea ice extent and types. For sea ice extent determination, both microwave radiometry and scatterometry have been utilised. To gauge sea ice extent through microwave radiometry, the brightness temperature, denoted as T_B , plays a significant role. T_B is predominantly influenced by the emissivity⁹ of the sea ice or sea surface. The distinction in T_B between the sea ice and sea is utilised to define the sea ice extent. However, this approach encounters limitations when the sea surface becomes roughened by wind, leading to an increase in the emissivity of the sea's surface. Consequently, the T_B of the sea surface reaches a similar level to that of the sea ice [7]. In the context of Antarctica, this

⁸Spreen and Kern [7] define remote sensing as any form of data acquisition without the need for physical contact or measurement.

⁹Emissivity, a material-specific property, is a function of the dielectric properties of the material [7].

issue arises due to the rough nature of the SO, potentially interfering with the differentiation between sea ice and the ocean surface.

Scatterometry can also be used to determine sea ice extent. The difference in radar backscatter¹⁰ between uncovered open water and sea ice enables the detection of an ice-covered region [7]. Rivas et al. [48] discovered that scatterometry is more accurate than microwave radiometry in determining sea ice extent, especially when the edge of the extent is composed of frazil ice.

The backscatter of different ice types differs depending on the ice's salinity, density, roughness, temperature, and snow coverage [7]. This differentiation allows the identification of various types of ice within a sea ice region based on the received backscatter intensity. Dierking [49] found that pancake ice exhibits higher radar backscatter compared to other ice types due to its increased surface roughness and the presence of numerous edges. This observation has been confirmed by Lange and Eicken [45] through their study of higher radar backscatter in the Antarctic MIZ.

2.2.2 Altimetry

Altimeters measure the time taken by an EM signal to be transmitted, reflected off the Earth's surface, and received. From this, the distance can be calculated. Laser and radar altimeters can be used to measure sea ice thickness. However, instead of directly measuring the thickness, these altimeters measure the sea ice freeboard¹¹, and infer the thickness from the sea surface height. To extract the surface elevation, the satellite's position in space needs to be known with centimetre accuracy [7].

To calculate sea ice thickness, both the freeboard and sea surface height need to be determined by the altimeter. One issue with sea surface height determination is the need to access uncovered ocean to act as a reference point [7]. In order to convert freeboard height to sea ice thickness, the density and snow depths need to be known. Ricker et al. [50] show that the origin of the return signal of the altimeter cannot always be known with certainty and state that snow cover is not negligible in determining distances due to the fact that the signals do not penetrate the snow completely, thus creating bias in measurements. Given that Antarctica experiences some of the highest snowfall rates in the world [12], this method will be unable to be implemented without the aid of in-situ measurements of snow depth [50]. Spreen and Kern [7] explain that improved retrieval methods using altimeters need to be implemented to overcome this.

2.2.3 Optical and Thermal Infrared Imaging

Optical and thermal infrared imaging techniques rely on the fact that sea ice reflects solar radiation more than open, uncovered water [7]. Optical imagery has been used to measure SIC [51] as well as thickness. Thermal imaging is useful, as sensors can measure the surface temperature, T_{surface} , of sea ice [7].

The average sea ice albedo is greater than 0.6, compared to the low albedo of water, which is 0.07. Brandt et al. [52] found that the albedo of ice increased with thickness [7], allowing these techniques to be used for measuring ice thickness. To measure the surface albedo using a satellite, a sensor capable of measuring surface reflectance needs to be employed [7].

¹⁰Backscatter is the reflection of a signal back towards its source.

¹¹Freeboard is defined by Ricker et al. [50] as the height of sea ice above the local sea level.

The main disadvantage of this technology is its limitations. Both optical and thermal imaging technologies require daylight as well as clear sky conditions to capture the desired data. Clouds pose a significant challenge for both of these technologies. In the optical spectrum, clouds exhibit similar reflectivity to sea ice, and the infrared temperatures of the surface of sea ice and clouds are also similar [7]. These issues imply that while this technology has been utilised for parameter extraction of sea ice, it can only be applied within a narrow time frame determined by environmental conditions, preventing year-round coverage of Antarctica.

2.3 Sea Ice Characterisation using SAR

The main disadvantage associated with the previously mentioned remote sensing techniques is the low spatial resolution. **SAR** overcomes this by using the motion of a satellite to synthetically increase the size of the antenna [7]. This effect is detailed in Section 3.2. Additionally, **SAR** data acquisition is independent of light and weather conditions, including cloud cover [41].

SAR can be used to determine sea ice type through measuring the differences in radar backscatter. Multi-year ice exhibits higher levels of volume scattering than first-year ice [7]. Furthermore, pancake ice can be distinguished from other ice types due to its increased number of rough surfaces, as discussed in Section 2.1.3. This differentiates its appearance in **SAR** data from pack ice [7].

However, Dierking emphasises that, to develop robust, and reliable sea ice parameter extraction algorithms, in-situ measurements are required to validate the accuracy of these algorithms [41]. As discussed in Section 2.1, this proves challenging due to Antarctica's climate.

2.3.1 SAR Satellites

Due to the nature of orbits around the planet¹², only specific **SAR** satellites can effectively monitor the polar regions [7]. These commonly used satellites are outlined in Table 2.1, along with key features of interest. The meaning and significance of these features will be expanded on and explained in Chapter 3.

Name	Frequency (GHz)	Frequency Band	Resolution (m)	Swath (km)	Incidence Angle (°)	Inclination (°)	Operational	Polarisation
ERS-1 and 2	5.3	C	30	100	20-26	98.5	1991-2011	VV
Envisat ASAR	5.3	C	30-1000	100-400	15-45	98.5	2002-2012	Dual-polarimetric ¹³
Radarsat-1 and 2	5.3/5.4	C	3-100	18-500	10-60	98.6	1995-	HH (RS-1) Full ¹⁴ (RS-2)
Sentinel-1	5.4	C	5-100	80-400	19-47	98.2	2014-	Full
TerraSAR-X	9.6	X	1-40	5-200	15-60	97.4	2007-	Full
ALOS-1 and 2 PALSAR	1.3	L	3-100	20-350	8-70	98.7 (AL-1), 97.9 (AL-2)	2006-2011 (AL-1) 2014- (AL-2)	Full
COSMO-SkyMed	9.6	X	1-100	10-100	16-51	97.9	2007-	Dual-polarimetric

Table 2.1: Commonly used **SAR** sensors for sea ice research and a comparison of their important features. Adapted from [7].

As per the scope of this project, data obtained from **S1A** will be used for pipeline development and testing. As such, this satellite will be investigated further in this subsection.

¹²For an orbit to capture polar regions, it needs to have a 90° inclination orbit [53].

Sentinel-1

Sentinel-1 is the first in a series of operational satellites launched by the European Space Agency ([ESA](#)) as part of the Global Monitoring for Environment and Security ([GMES](#)) program - an initiative aimed at meeting the Earth Observation needs of the European Union [54]. The Sentinel-1 constellation comprises two [SAR](#) satellites: [S1A](#) and Sentinel-1B. These satellites were launched on April 3, 2014, and April 25, 2016, respectively [55, 56]. Equipped with an active phased array antenna operating at C-Band frequency, Sentinel-1 is capable of operating in various modes¹⁵. While Sentinel-1 serves multiple applications, its primary focus is monitoring sea ice zones in the Arctic and Antarctic regions [54, 55], in addition to operating on an open-source data policy [56].

This application is achieved through the orbit of Sentinel-1. Both [S1A](#) and Sentinel-1B fly in the same orbital plane; however, they are phased by 180° ¹⁶. The mission is on a sun-synchronous orbit¹⁷ and has a 12-day orbit repeat cycle [58]. To ensure a polar orbit, the satellites have an inclination angle of 98.2° , as shown in Table 2.1. The open-source policy of this mission is useful for this project, as Level 0, Level 1, and Level 2 data products¹⁸ are available for download on the [Sentinel Copernicus Open Hub](#) [56].

The global coverage of Sentinel-1 Interferometric Wide Swath ([IW](#)) mode is depicted in Figure 2.2. These captures encompassing the Antarctic continent in Figure 2.2 are of particular interest to this project. Garkusha and Hnatushenko [3] utilised data from this region to compile a mosaic of the entire Antarctic coastline using Sentinel-1 [SAR](#) data. The coverage of the coastline is illustrated in Figure 2.3a. To achieve complete coverage for the construction of this mosaic, Garkusha and Hnatushenko employed GRD data from 98 captures from Sentinel-1. This mosaic construction is illustrated in Figure 2.3b.

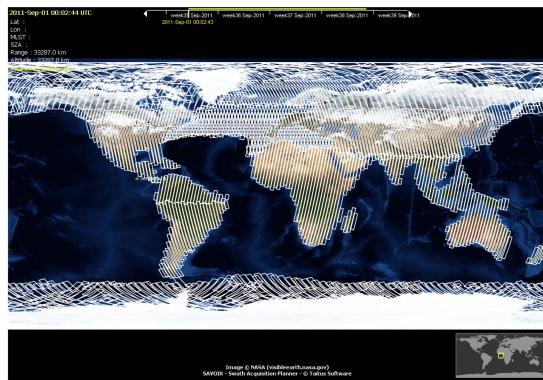


Figure 2.2: [IW](#) mode coverage of one Sentinel-1 satellite after one 12-day orbit cycle. Taken from [2].

Overall, the Sentinel-1 mission has been successful in capturing [SAR](#) data over Antarctica and the [SO](#), as depicted in Figures 2.2 and 2.3a. Due to its open-source policy, the mission will be able to provide the desired data for this project.

¹⁵These modes include Interferometric Wide Swath ([IW](#)), Extra Wide Swath ([EW](#)), StripMap ([SM](#)), and Wave. All these modes, except Wave, operate in dual polarisation. Further details on the Sentinel-1 imaging modes can be found in Figure 1 and Table 1 of [55].

¹⁶This phased orbit of the constellation means that the satellites are on opposite sides of the Earth at any given time. More information can be found in [55].

¹⁷Sun-synchronous orbits allow a satellite to cross the same latitude at the same solar time of day on each orbit [57].

¹⁸The differentiation between these data is discussed in the theoretical development chapter.

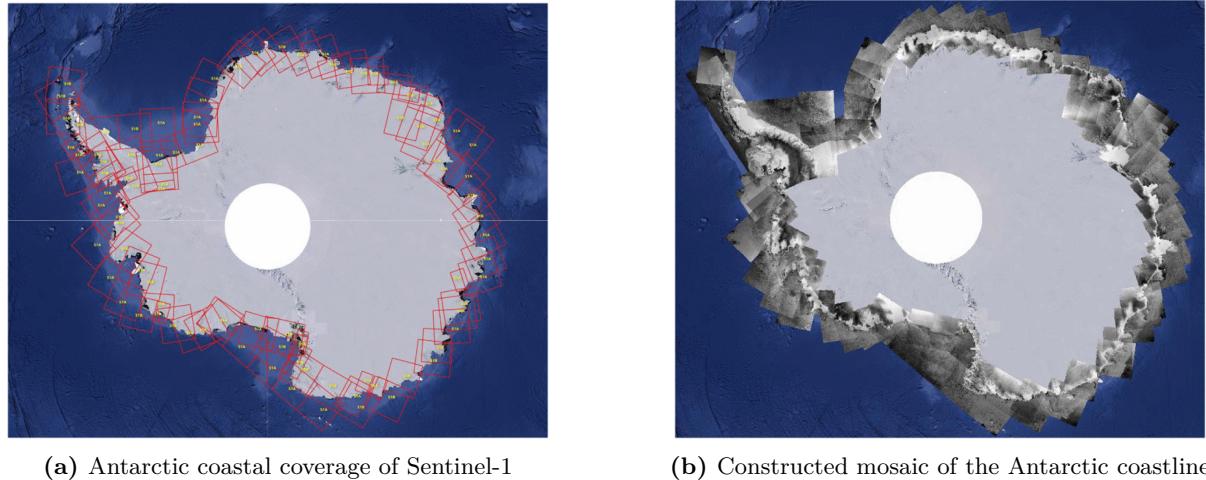


Figure 2.3: Antarctic coast line coverage and reconstructed continuous **SAR** image using Sentinel-1. Taken from [3].

2.3.2 SAR Ocean Wave Inversion

In order to convert a **SAR** spectrum into an directional wave spectrum, two main conversion methods are employed. The first technique is the Hasselmann inversion technique, originally proposed by HH [59] in 1991, and further improved upon in 1996 by Hasselmann et al. [60]. The theory encompassing this inversion technique, along with the related wave spectra theory, is discussed in Chapter 3. This section will introduce the foundational aspects of this techniques, as well as their limitations and further research conducted to improve the technique. Subsequently, the following section will introduce the applications of these models in sea ice and wave modelling.

Hasselmann Inversion Technique

HH developed a closed, non-linear integral mapping transformation [47], which relates a **SAR** spectrum to a two-dimensional ocean-directional wave spectrum denoted as $E(\mathbf{k})^{19}$. This wave spectrum enables the derivation of all statistical properties of an ocean wave field at any location and time [59].

HH's inversion technique builds upon prior ocean inversion methods; however, these techniques were computationally inefficient and required a brute-force approach through the utilisation of Monte Carlo simulations²⁰ for the statistical modelling of ocean waves using a pixel-by-pixel approach [59].

HH state that not all of the wave spectral information is mapped into the **SAR** image plane, and as such, they can only determine wave propagation direction to a specific sign [59]. This loss of information is due to two main issues. Firstly, the 180° propagation ambiguity²¹, and secondly, due to the non-linear orbital motion of waves - particularly short waves and waves in

¹⁹ $E(\mathbf{k})$ describes the distribution of wave energy with respect to the wave propagation number, \mathbf{k} . Further information about the two-dimensional wave spectrum is provided in Section 3.1.2 and can be found in [4, 59].

²⁰Monte Carlo simulations predict outcomes by employing estimated variable ranges and probability distributions. They iteratively generate results with various random numbers to produce a spectrum of likely scenarios.

²¹This ambiguity is due to the polarisation of the **SAR** waveform, as it does not allow the direction of the waves to be properly determined, i.e., whether they are travelling towards or away from a coastline or sea ice [61].

windy oceans. This non-linearity is known as velocity bunching and occurs due to fluctuations in the cross-range displacements of image backscattering elements, caused by changing orbital velocities within the wave field [59]. The non-linearity in this relationship stems from the interference between wave orbital velocities and the way in which **SAR** constructs its cross-range resolution [62]. **HH** state that velocity bunching dominates over other non-linearities and is the cause of smearing in the output **SAR** image spectrum [59]. This results in a loss of information for high wave numbers of the wave spectrum in the cross-range direction [59, 60].

In order to mitigate these non-linearities, **HH** adopts the approach of utilising a first-guess wave spectrum, derived from an existing model of the sea state. Using this initial guess spectrum, **HH**'s technique aims to minimise a cost function for extracting the wave spectrum. **HH** has discovered that iterative minimisation of this cost function leads to convergence within 3-4 iterations [59]. While this approach to mitigating non-linearities proved successful, it requires prior knowledge of the sea state. Consequently, several researchers have attempted to address this limitation using various methods. Some of these methods are discussed in the subsequent subsections.

Image Cross Spectra Inversion Technique

Building upon the **HH** inversion technique, Engen and Johnsen were able to mitigate the 180° propagation ambiguity without using an existing first-guess wave model [63]. This was achieved using the image cross spectra which was derived from pairs of single look **SAR** images [63]. These pairs of single look **SAR** images were separated by a time less than the dominant wave period. Engen and Jonhsen explain that this method had an additional advantage over the **HH** technique. The use of image cross spectra significantly reduced the impact of speckle noise in the **SAR** spectrum [63]. However, this method requires satellites which are based on the image cross spectra rather than the regular image spectra.

Semiparametric Algorithm

Mastenbroek and De Valk [62] further developed the **HH** technique by eliminating the need for a first-guess wave spectrum. This was achieved through the use of an algorithm, known as the semiparametric retrieval algorithm, which made use of wind vector data sourced from the European Remote-Sensing Satellite (**ERS**). **ERS**-1 is equipped with an on-board scatterometer, as well as a **SAR** sensor whose swath overlap [62]. Whilst this limited this algorithm to a certain subset of satellites, it is still an interesting approach to mitigate the use of a first-guess wave model. An interesting result found by [62], was the ability to estimate the stage of development of the wind sea state.

2.3.3 Wave Propagation Model in Sea Ice

The process of modelling sea ice and waves is achieved through the use of three wave attenuation and dispersion models [64, 65]. These models are crucial to Antarctic science, as they enable the assessment of wave attenuation rate, spreading, and dispersion within sea ice, which are closely linked to the physical characteristics of the sea ice itself. This, in turn, allows for the inference of sea ice's physical characteristics [46].

These two models are the Keller model, developed in 1998 by Keller [66], and the two-layer viscous model, developed in 2002 by De Carolis and Desiderio [46]. More information on these

models can be found in the respective references. These models can then be used to infer properties of the sea ice region, such as pancake ice thickness and wave attenuation rate [67]. This subsection will introduce each of these models, discuss their results when applied following the definition of an open ocean wave spectrum using either of the inversion techniques described in the above subsection and state the difficulties and inaccuracies found within these models.

Keller Model

The Keller model [66] views the ice layer as a suspension with a higher viscosity than the water it sits on top of, as well as having a density slightly less than that of the water. The sea ice is treated as a highly viscous incompressible liquid, whereas the water underneath it is treated as an inviscid²² incompressible liquid. This allows breaking the problem down into a two-layer problem, for which linear theory can be applied to solve the dispersion equation for waves entering a sea ice field [66]. Keller's model relies heavily on the effective viscosity coefficient, $\mu(c)$, which differs for different ice types based on their shape and concentration. The model has only two free parameters: the viscosity and thickness of the sea ice. The best values for these are found by minimising the difference between the observed and simulated SAR spectrum [68].

Keller's model improves on the previously used mass loading model since, at high wave frequencies, it fits laboratory experiments well by predicting an increase in wave wavelength upon entering the sea ice. Furthermore, Wadhams et al. [68] also found that the inferred thickness using Keller's model showed excellent agreement between calculated ice thickness and the average ice thickness from in-situ measurements in the MIZ.

Two-layer Viscous Model

The two-layer viscous model [46] builds upon the Keller model due to the constraints from specific approximations. This extension incorporates wave attenuation rate and dispersion as functions of wave frequency. Additionally, it introduces an eddy viscosity for the water beneath the sea ice, contributing to improved accuracy. The emergence of this eddy viscosity stems from the consideration of turbulence at the interface between the sea ice and the ocean, particularly at the base of the ice layer in order to parameterise water flow underneath sea ice [46, 65].

De Carolis and Desiderio [46] represent pancake ice as a layer of viscous fluid [14, 46]. This viscosity of the ice layer governs various interactions that occur when waves cross a region covered by pancake ice, including bending and collisions among individual pancakes [14]. As with the Keller model, the two-layer viscous model maintains the idea that the ice layer exhibits a higher viscosity than the underlying ocean, and it considers the influence of ice density, which is dependent on ice type and concentration [14, 46]. The two-layer viscous model demonstrates good agreement with laboratory results concerning observed wave attenuation and dispersion in grease ice [46]. However, De Carolis and Desiderio [46] state that while the inclusion of an eddy viscosity holds scientific value, it needs to be modelled using more robust numerical methods for further validation.

²²An inviscid liquid has negligible or zero viscosity.

2.4 Literature Critique

This chapter has clearly demonstrated that comprehensive research has been conducted into both Antarctic climatic science, as well as remote sensing techniques to extract sea ice parameters. The significance of the Antarctic region, and particularly pancake ice, contained within the [MIZ](#), on the global climate were discussed.

To achieve year-round sea ice parameter extraction, it is evident that remote sensing techniques are required. This chapter has investigated various remote sensing methods, and highlighted their advantages and limitations. In the context of [SAR](#), three wave parameter extraction algorithms were briefly assessed. The literature compared the results of these algorithms to ground-truth data. Additional investigation was done into wave attenuation and dispersion models in sea ice. The literature revealed that these models were successful, however, and certain limitations were found when tested with respect to ground-truth data, and laboratory experiments.

However, the existing literature fails to address the complete processing pipeline for sea ice and wave parameter extraction from SAR data. While previous research concentrates on algorithm development, it omits the critical step of implementation. Therefore, the objective of this report is to implement the discussed [HH](#) technique, providing insights into the pipeline's design whilst carefully considering the limitations and nuances associated with the [HH](#) technique. Provided that this pipeline is correctly implemented, it will enable year-round validation of newly developed wave attenuation and dispersion models for the [MIZ](#) and pancake ice.

Chapter 3

Theoretical Background

To understand the [HH](#) inversion technique, the theoretical foundations surrounding the modelling of ocean waves¹, in both deep and shallow water, need to be understood. Furthermore, a foundational understanding of [SAR](#), encompassing its data acquisition and types, along with the pre-processing techniques applied in implementing the [HH](#) technique, is necessary.

This chapter's objective is to introduce the theoretical aspects surrounding these fields, thereby providing context for the subsequent chapter's introduction of the pipeline design. Additionally, this chapter builds upon the theories previously introduced in Chapter 2.

3.1 Ocean Waves

The majority of this section draws upon content collated by Holthuijsen [4], with equations and figures cited appropriately. This section is to provides an introductory overview of the fundamental theory related to ocean waves and their mathematical modelling as well as highlighting the differences in modelling deep and shallow water.

3.1.1 Description of Ocean Waves

The wave height, H , can be defined as the vertical distance between the lowest and highest surface elevation in a wave. Where surface elevation, denoted using $\eta(t)$, refers to the instantaneous elevation of the sea surface relative to some reference level. The zero-crossing wave period, T_0 , can be defined as the time between one zero-down crossing and the following one. Where a zero-down crossing is the crossing of the mean surface elevation when the gradient of the wave is negative. All of these terms are depicted in Figure 3.1.

Holthuijsen [4] defines the following equations for the mean wave height and zero-crossing wave period for N wave where i represents the sequence number of the waves in the wave record².

$$\bar{H} = \frac{1}{N} \sum_i^N H_i \quad (3.1)$$

$$\bar{T}_0 = \frac{1}{N} \sum_i^N T_{0,i} \quad (3.2)$$

¹Ocean waves are defined as wind-generated surface gravity waves [4].

²A wave record is defined as the surface elevation, $\eta(t)$, as a function of time with duration, D [4].

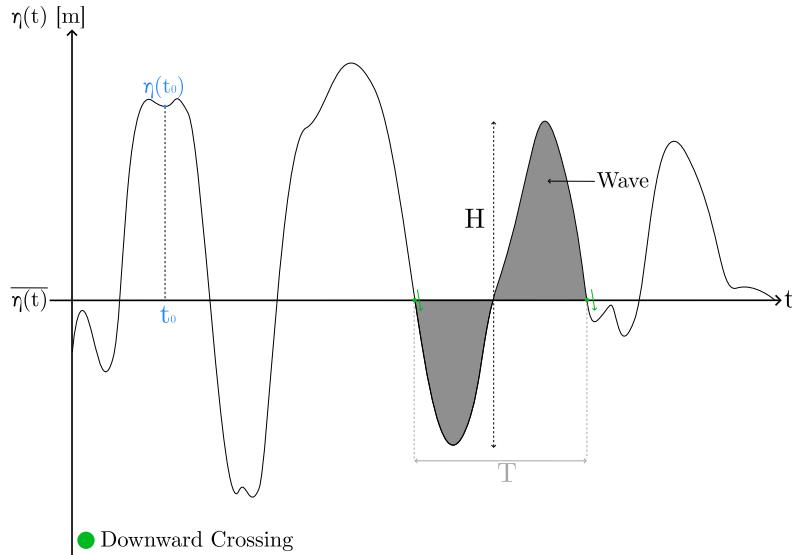


Figure 3.1: Visual definition of surface elevation ($\eta(t)$), wave height (H), wave period (T), and zero-down crossing. Adapted from [4].

The definition of the Significant Wave Height (**SWH**) and Significant Wave Period (**SWP**) follows, where j is the rank number of the sequence of waves, based on wave height, in descending order. The **SWH** is a useful metric, as it can be estimated from the wave spectrum [4].

$$H_{1/3} = \frac{1}{N/3} \sum_j^{N/3} H_j \quad (3.3)$$

$$T_{1/3} = \frac{1}{N/3} \sum_j^{N/3} T_{0,j} \quad (3.4)$$

3.1.2 Wave Spectra

The aim of representing ocean waves as a spectrum is to define the sea state as a stochastic process encompassing all feasible time record observations, based on current sea state conditions. The most important form in which ocean waves are described is the random phase/amplitude model, which forms the basis of the wave spectrum [4].

Random Phase/Amplitude Model

The wave spectrum aims to describe the ocean surface as a stationary stochastic process³ [4]. It is based on the random-phase/amplitude model, where the surface elevation is treated as the sum of a large number of harmonic waves. Each of these waves possesses a constant amplitude and a phase that is randomly selected for each time record realisation. This approach enables the modelling of the wave spectrum as a Fourier series, as illustrated in Equation 3.5 [4]. For a more detailed explanation, please refer to [4].

³A stochastic process represents a collection of random variables indexed by a different variable, such as time, and is classified as stationary if its statistical characteristics remain constant over this index variable, ie., time.

$$\underline{\eta}(t) = \sum_{i=1}^N \underline{a}_i \cos(2\pi f_i t + \underline{\phi}_i) \quad (3.5)$$

The random phase/amplitude model has two key considerations related to its applicability. Firstly, the model aims to generate a stationary stochastic sea state, even though real sea conditions are never truly stationary. This is achieved by dividing up the larger record into intervals, typically lasting 15-30 minutes [4]. Secondly, the model represents a discrete sum of frequencies, whereas, in reality, continuous frequencies exist at sea [4]. To mitigate this second consideration, the continuous variance density spectrum is defined.

Variance Density Spectrum

The variance spectrum can be constructed by utilising the variance⁴, $E\left\{\frac{1}{2}\underline{a}_i^2\right\}$, as opposed to the expected value, $E\{\underline{a}_i\}$. Variance is preferred as it allows the change across different samples to be observed, as opposed to each sample's pure magnitude.

To mitigate the second issue with the application of the random phase/amplitude model, it is modified by distributing the variance over the frequency interval Δf_i and the frequency f_i [4]. This gives rise to the variance **density** spectrum, $E^*\left\{\frac{1}{2}\underline{a}_i^2\right\}$.

$$E^*\{f_i\} = \frac{1}{\Delta f_i} E\left\{\frac{1}{2}\underline{a}_i^2\right\} \quad \text{for all } f_i \quad (3.6)$$

Whilst Equation 3.6 is defined for all frequencies, it is still discontinuous between frequency bands [4]. A continuous version of $E^*\{f_i\}$ can be determined by taking the limit of $\Delta f_i \rightarrow 0$.

$$E(f) = \lim_{\Delta f \rightarrow 0} \frac{1}{\Delta f} E\left\{\frac{1}{2}\underline{a}^2\right\} \quad (3.7)$$

The variance density spectrum provides a complete description of the surface elevation of ocean waves as all statistical characteristics of the wave field can be expressed in terms of this spectrum [4], provided it is modelled as a stationary, Gaussian process.

Spectral Domain Definition

Equations 3.5 and 3.7 are defined in terms of time and frequency. To represent the variance density spectrum in the radian frequency, ω , the relationship, $\omega = 2\pi/T$ needs to be used. Equation 3.5 is then updated to the following form

$$\underline{\eta}(t) = \sum_{i=1}^N \underline{a}_i \cos(\omega t + \underline{\phi}_i) \quad (3.8)$$

⁴The variance of $\eta(t)$, is the average of the time-averaged squared surface elevation, $\overline{\eta^2} = \frac{1}{2}\underline{a}^2$ [4].

This allows the variance density spectrum, $E(\omega)$, to be defined as follows, where J represents the Jacobian. In this case, $J = 1/2\pi$ [4].

$$E(\omega) = E(f) \frac{df}{d\omega} = E(f)J = \frac{1}{2\pi} E(f) \quad (3.9)$$

Frequency-direction Spectrum

The variance density spectrum equation as described in Equation 3.7 represents a one-dimensional function of time. To consider a two-dimensional spectrum, the random phase/amplitude model needs to be expanded to consider a wave propagating in the x, y -space in a direction θ , relative to the $+x$ -axis. This allows the model to be expanded as follows, where the wave number, $k = 2\pi/L$ [4] with L equal to the length of the harmonic wave.

$$\underline{\eta}(x, y, t) = \sum_{i=1}^N \sum_{j=1}^M \underline{a}_{i,j} \cos(\omega_i t - k_i x \cos(\theta_j) - k_i y \sin(\theta_j) + \underline{\phi}_{i,j}) \quad (3.10)$$

where each wave component is indicated by two indices. Where i represents the frequency index or wave number and j represents the wave direction.

Following the same process of transforming a one-dimensional amplitude spectrum into a continuous variance density spectrum, as detailed in Equations 3.6, 3.7, the two-dimensional variance density spectrum is found as

$$E(\omega, \theta) = \lim_{\Delta\omega \rightarrow 0} \lim_{\Delta\theta \rightarrow 0} \frac{1}{\Delta\omega \Delta\theta} E \left\{ \frac{1}{2} \underline{a}^2 \right\} \quad (3.11)$$

Using the Jacobian determined in Equation 3.9, the same relationship can be developed for the two-dimensional variance density spectrum

$$E(\omega, \theta) = \frac{1}{2\pi} E(f, \theta) \quad (3.12)$$

3.1.3 Wave-number Spectra

Where the wave spectrum considers the sea as a function of space and time, $\eta(x, y, t)$, the ocean can also be considered as a function of space at a single moment in time. This idea is useful for remote sensing applications.

To define the two-dimensional wave-number spectrum, the method described in Equations 3.5, 3.7 can be used. Where Δk_x and Δk_y represent the spectral bandwidths [4].

$$\eta(x, y) = a_{i,j} \cos(k_{x,i} x + k_{y,j} y + \phi_{i,j}) \quad (3.13)$$

$$E(k_x, k_y) = \lim_{\Delta k_x \rightarrow 0} \lim_{\Delta k_y \rightarrow 0} \frac{1}{\Delta k_x \Delta k_y} E \left\{ \frac{1}{2} \underline{a}^2 \right\} \quad (3.14)$$

With $k_x = k \cos(\theta)$, and $k_y = k \sin(\theta)$, k and θ can be defined as: $k = \sqrt{k_x^2 + k_y^2}$ and $\theta = \arctan(k_y/k_x)$. This allows an equivalent spectrum to be defined in terms of k and θ .

$$E(k, \theta) = \lim_{\Delta k \rightarrow 0} \lim_{\Delta \theta \rightarrow 0} \frac{1}{\Delta k \Delta \theta} E \left\{ \frac{1}{2} \underline{a}^2 \right\} \quad (3.15)$$

These two spectra are related by the following relationship [4].

$$E(k, \theta) = E(k_x, k_y) J = k E(k_x, k_y) \quad (3.16)$$

The two-dimensional frequency-direction spectrum in Equation 3.11 is related to the spectrum in Equation 3.14 by the following relationship, where $c_w = \omega/k$ and the Jacobian, $J = 1/c_g$ [4]. c_w and c_g are defined as the wave speed and group wave speed respectively and are discussed in the following subsection.

$$E(k_x, k_y) = k E(\omega, \theta) J = \frac{c_w c_g}{\omega} E(\omega, \theta) \quad (3.17)$$

These two-dimensional spectra are useful in two different applications, however, in varying forms. The wave-number spectrum, $E(k_x, k_y)$ is useful for remote sensing applications, and the frequency-direction spectrum, $E(f, \theta)$, is used for wave models [4].

3.1.4 Linear Wave Theory

The linear theory for ocean waves provides a detailed description of harmonic waves. It relies on two equations and associated boundary conditions, which are discussed in Chapter 5.3 of [4]. The solution to the kinematic boundary conditions results in a wave propagating in the x -direction. This propagating harmonic can be described in two different ways.

$$\eta(x, t) = a \sin(\omega t - kx) \quad (3.18a)$$

$$\eta(x, t) = \frac{H}{2} \sin \left(\frac{2\pi}{T} t - \frac{2\pi}{L} x \right) \quad (3.18b)$$

3.1.5 Ocean Wave Dynamics

Considering the harmonic surface profile of ocean waves, as well as the velocity potential function, which are described by Holthuijsen in Chapter 5.4.2 [4], a relationship, known as the dispersion relationship can be developed.

Dispersion Relationship

The dispersion relationship relates the radian frequency, ω , to the wave number, k [4]. The dispersion relationship for an arbitrary depth is given as

$$\omega^2 = g k \tanh(kd). \quad (3.19)$$

The propagation speed of a wave can be determined from Equation 3.19 with the fact that $c_w = \omega/k$. This gives the wave phase speed for an arbitrary depth as

$$c_w = \frac{g}{\omega} \tanh(kd) = \sqrt{\frac{g}{k} \tanh(kd)} \quad (3.20)$$

Considering Equation 3.19 and 3.20 for both deep water and shallow water yields the following approximations to be determined.

Deep Water

In deep water, the term, $kd \rightarrow \infty$, and due to this, $\tanh(kd) \rightarrow 1$ [4]. This means that the dispersion relationship for deep water approaches

$$\omega_0 = \sqrt{gk_0} \quad (3.21)$$

where k_0 is the deep water wave number. The same logic with respect to $kd \rightarrow \infty$ applies to the phase velocity and means that the deep water phase velocity reduces to

$$c_{w0} = \sqrt{\frac{g}{k_0}} = \frac{g}{\omega_0} = \frac{g}{2\pi} T \quad (3.22)$$

Shallow Water

In shallow water, the term, $kd \rightarrow 0$, and due to this, $\tanh(kd) \rightarrow kd$ [4]. This means that the dispersion relationship for shallow water approaches

$$\omega = k\sqrt{gd} \quad (3.23)$$

The same logic with respect to $kd \rightarrow 0$ applies to the phase velocity and means that the shallow water phase velocity reduces to

$$c_{w_{shallow}} = \sqrt{gd} \quad (3.24)$$

Equation 3.24 shows that the phase speed does not depend on wavelength or frequency, and as such, this means that the waves are non-dispersive⁵.

Wave Groups

Using the definition of a propagating harmonic wave in Equation 3.18, two of these waves can be combined in the following form.

⁵Ocean waves are defined as non-dispersive if the phase speed of the wave does not depend on wave number, wavelength or frequency [4].

$$\eta = \eta_1 + \eta_2 = a\sin(\omega_1 t - k_1 x) + a\sin(\omega_2 t - k_2 x) \quad (3.25)$$

These two waves combine to create a sequence of wave groups. This group reaches its peak surface elevation when both η_1 and η_2 are in phase with each other. The phase velocity of these waves can be defined as the phase speed of the surface elevation envelope. This envelope can be determined as the cosine term, whilst the sine term acts as the carrier wave [4]. The phase velocity of the envelope represents the velocity of the wave group and can be derived as

$$c_g = \frac{\partial \omega}{\partial k} = nc_w \quad (3.26)$$

where n is derived from Equation 3.19 [4] as

$$n = \frac{1}{2} \left(1 + \frac{2kd}{\sinh(2kd)} \right) \quad (3.27)$$

The value of n varies between $\frac{1}{2}$ for deep water, and 1 for shallow water [4] and implies that $c_w \geq c_g$.

3.1.6 Wave Modelling for Idealised Cases

One-dimensional Wave Spectrum

The one-dimensional wave spectrum has been significantly advanced by the JOint North Sea WAve Project (**JONSWAP**), which is the most widely employed spectrum for wave modelling [4]. This spectrum builds upon the Pierson-Moskowitz spectrum [69], which models a fully developed spectrum in deep water. It expands upon the Pierson-Moskowitz spectrum by incorporating its spectral shape and introducing a peak enhancement factor denoted as $G(f)$ [70, 71].

The Pierson-Moskowitz spectrum is defined as:

$$E_{PM}(f) = \alpha g^2 (2\pi)^{-4} f^{-5} \exp \left[-\frac{5}{4} \left(\frac{f}{f_{peak}} \right)^4 \right] \quad (3.28)$$

which results in the derivation of the **JONSWAP** spectrum as

$$E_{JONSWAP}(f) = E_{PM}(f) G(f) \quad (3.29)$$

where,

$$G(f) = \gamma \exp \left[-\frac{1}{2} \left(\frac{f/f_{peak}^{-1}}{\sigma} \right)^2 \right] \quad (3.30)$$

γ is the peak enhancement factor and is determined experimentally for the specific region over which waves need to be modelled. σ is the peak width parameter which varies between σ_a and σ_b for the following conditions [4]:

$$\sigma = \begin{cases} \sigma_a = 0.07 & \text{for } f \leq f_{peak} \\ \sigma_b = 0.09 & \text{for } f > f_{peak} \end{cases}$$

α is called the energy scale parameter and is calculated using the SWH and peak frequency. Equations 3.28, 3.30, 3.29 are represented visually in Figure 3.2.

$$\alpha = 0.2 \frac{H_{1/3}^2 f_{peak}^4}{g^2} \quad (3.31)$$

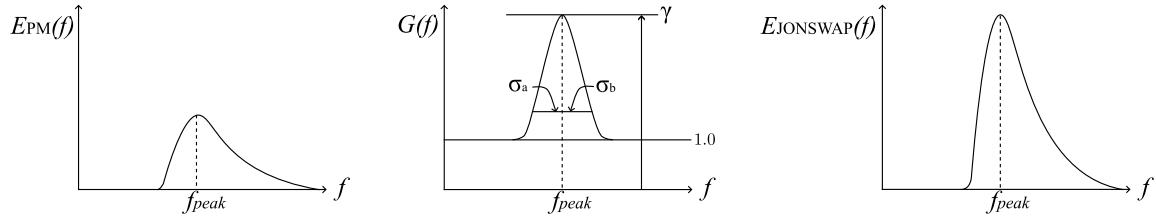


Figure 3.2: Graphical representation of Equations 3.28, 3.30, 3.29 which represent the Pierson-Moskowitz spectrum, peak enhancement function and the resultant JONSWAP spectrum respectively. Adapted from [4].

Two-dimensional Wave Spectrum

To extend one-dimensional wave models generated using the JONSWAP spectrum to two dimensions, the introduction of a directional distribution function is required. This directional distribution function represents the cross-section of the two-dimensional spectrum at a specific frequency and is normalised so that the integral of the function over its distribution equals one [4] and relates the one-dimensional wave spectrum to the two-dimensional spectrum using the relationship in Equation 3.32.

$$D(\omega, \theta) = \frac{E(\omega, \theta)}{E(\omega)} \quad (3.32)$$

It is a function of both θ and frequency but is often denoted as $D(\theta)$. The directional distribution employs a $\cos^2 \theta$ model [4] and takes the form

$$D(\theta) = A_2 \cos^{2s} \left(\frac{1}{2} (\theta_{wave} - \theta_{wind}) \right) \quad (3.33)$$

where A_2 is determined using the gamma function⁶, $\Gamma(\cdot)$, and s controls the distribution's width.

$$A_2 = \frac{\Gamma(s+1)}{\Gamma\left(s + \frac{1}{2}\right) 2\sqrt{\pi}} \quad (3.34)$$

and s is related to the directional width, σ_θ by,

⁶The gamma function, $\Gamma(\cdot)$, is an extension of the general factorial function, but allows both complex and real numbers. $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$.

$$s = \frac{2}{\sigma_\theta^2} - 1 \quad (3.35)$$

A reasonable approximation for σ_θ is given by

$$\sigma_\theta = \begin{cases} 26.9 \left(\frac{f}{f_{peak}} \right)^{-1.05} & \text{for } f < f_{peak} \\ 26.9 \left(\frac{f}{f_{peak}} \right)^{0.68} & \text{for } f \geq f_{peak} \end{cases}$$

3.2 Synthetic Aperture Radar (SAR)

SAR is an active air- or space-borne sensor which transmits and receives EM radiation at a certain carrier frequency, f_c . SAR is a form of imaging radar and is built upon general radar principles.

3.2.1 RAR vs. SAR

Both SAR and Real Aperture Radar (RAR) systems consist of a radar sensor mounted on an airborne or space-borne platform, which is oriented away from the nadir⁷ track. When a radar sensor is used in this manner in an airborne context, it is referred to as Side-looking Airborne Radar (SLAR). This orientation is determined by a look angle, denoted as θ_l . Figure 3.3 provides a graphical representation of the geometry of a SLAR system. This configuration results in the illumination of a specific area on the ground known as the swath. The size of this swath changes as the aircraft progresses along its flight path.

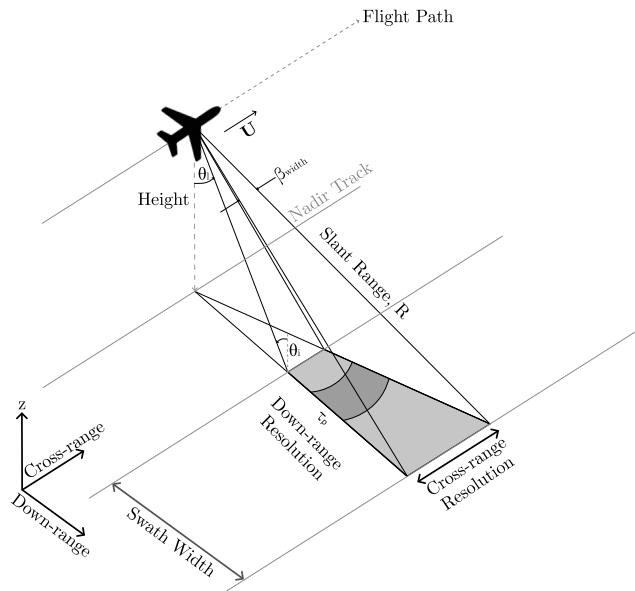


Figure 3.3: Graphical representation of a SLAR system with important features labelled. Adapted from [5, 6].

⁷Nadir is defined as the location directly below the radar sensor on the Earth's surface. From this, the nadir track can be defined as the path directly below the airborne sensor's flight path.

The radar system operates by transmitting a series of chirps, each with a pulse length defined as τ_p , which illuminate the region of the ground referred to as the antenna footprint [6]. To calculate the size of this footprint, consider the radar's slant range, R , which represents the distance between the antenna and its footprint, as well as the beamwidth, denoted as β_{width} . The beamwidth can be defined as the ratio between the radar's wavelength, λ , and the antenna length, L_{ant} . This definition enables the determination of the antenna footprint size, S [6]

$$S \approx \frac{\lambda}{L_{ant}} R = \beta_{width} R \text{ [m]} \quad (3.36)$$

This footprint is shown in dark grey in Figure 3.3 as well as depicting the incidence angle, θ_i , which can be derived as $90^\circ - \theta_l$. The resolution of this beamwidth in both the down-range and cross-range directions can be empirically determined. The slant range resolution of a **SLAR** system is defined in terms of the speed of light, c and the pulse length, τ_p as

$$\delta_{sr} = \frac{c\tau_p}{2} \text{ [m]} \quad (3.37)$$

The resolution of a radar system allows it to differentiate between different objects at different slant ranges from the sensor [5]. It is also useful to define the down-range resolution in terms of the incidence angle and slant range resolution as

$$\delta_{dr} = \frac{\delta_{sr}}{\sin(\theta_i)} \text{ [m]} \quad (3.38)$$

and the cross-range resolution as

$$\delta_{cr} = \frac{\lambda}{L_{ant}} R \text{ [m]}. \quad (3.39)$$

To see the drawback of **RAR** in terms of cross-range resolution, it is useful to apply values to Equation 3.39. Consider an X-band⁸ **RAR** radar system with a slant range to the target of 7 km, as in a space-borne application, and with an antenna length of 5 m.

$$\delta_{cr} = \frac{0.03 \text{ m}}{5 \text{ m}} \cdot 7000 \text{ m} = 42 \text{ m}$$

The above example shows that a space-borne application of **RAR** results in a loss of cross-range resolution. In the context of this project, this is impractical as ocean waves need to be imaged and differentiated between.

The limited resolution of **RAR** led to the development of **SAR**. This technology addresses the impact of antenna length by synthetically imitating a longer antenna length, which enhances the cross-range resolution. This is accomplished through exploiting the Doppler shift phenomenon and capturing multiple images of the same scene. For a more comprehensive understanding of the mathematical and signal processing aspects involved in generating a **SAR**, refer to [73].

⁸X-band is a standard band designation of radar frequencies defined by Institute of Electrical and Electronics Engineers (IEEE). X-band is defined to have a frequency range of 8 GHz - 12 GHz, with a wavelength from 2.5 cm to 3.8 cm [72].

The synthetic length of the antenna [6] can be determined as

$$L_{SA} \approx \beta_{width} \cdot R \quad (3.40)$$

where the artificial beamwidth can be calculated as

$$\beta_{width_{SA}} = \frac{\lambda}{2L_{SA}} \quad (3.41)$$

This, in turn, allows the cross-range resolution [5] for a **SAR** sensor to be calculated as

$$\delta_{cr_{SA}} = R \cdot \beta_{SA} = R \cdot \frac{\lambda}{2L_{SA}} = \frac{L_{ant}}{2} \quad (3.42)$$

Using the previous example of a space-borne sensor with an antenna of length, 5 m, it can be seen that **SAR** improves the cross-range resolution when compared to **RAR**.

$$\delta_{cr_{SA}} = \frac{5 \text{ m}}{2} = 2.5 \text{ m}$$

This example shows that **SAR** allows fine cross-range resolution, a vital parameter in the application of imaging ocean waves. Along with this, **SAR** offers year-round imaging in any weather conditions as discussed in Section 2.3.

3.2.2 SAR Imaging Modes

Each **SAR** sensor can capture data in multiple modes with the most common modes being Spotlight, **SM**, and ScanSAR. Sentinel-1A offers data products using three main capture modes: **SM**, **IW**, and **EW** modes. **IW** and **EW** are both a new type of Scan**SAR** based on Terrain Observation with Progressive Scan **SAR** (**TOPSAR**). These capture modes are shown visually in Figure 3.4 where U represents the satellite's velocity.

StripMap (**SM**) Mode

SM mode is the conventional **SAR** imaging mode [74]. In **SM** mode, the antenna has a fixed direction, which observes a fixed swath, whilst the satellite moves along its flight path [5] as depicted in Figure 3.4a. This allows a continuous ground swath to be illuminated with a continuous sequence of chirps. The echoes received back by the sensor are processed and combined to form a continuous image of the observed scene. **SM** mode provides increased cross-range resolution as multiple echoes are received back from the same target within the scene. **SM** mode is a versatile imaging mode as it allows detailed imagery over a large area to be obtained over a single, continuous strip at a constant incidence angle [5, 8, 74].

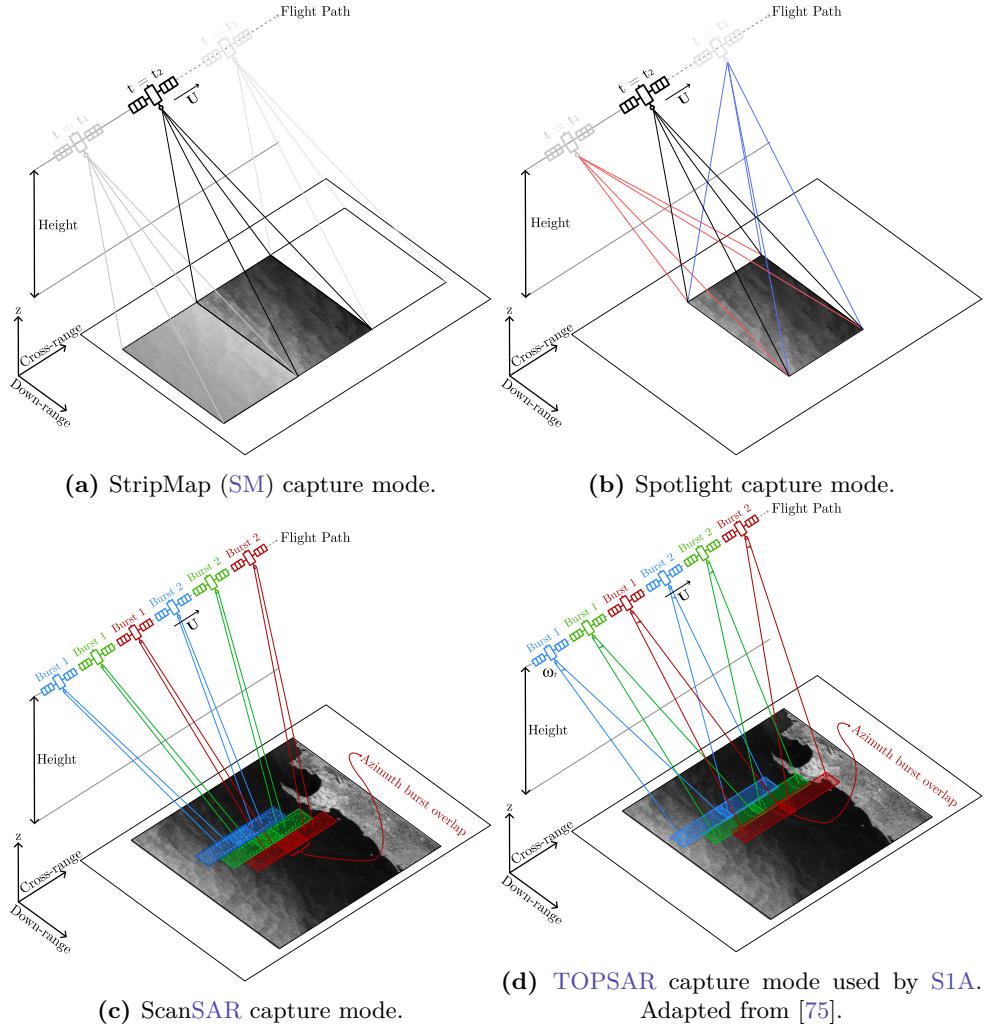


Figure 3.4: Graphical representation of commonly used SAR satellite capture and specific S1A capture modes.

Spotlight Mode

Spotlight mode is used for high-resolution SAR imaging [6], and involves targeting a specific area while continuously illuminating and capturing echoes during satellite movement along its flight path, as depicted in Figure 3.4b. Antenna beam control can be achieved mechanically or electronically through beam steering [74]. Spotlight mode improves both down- and cross-range resolutions. Whilst its spatial coverage is smaller compared to SM mode. By extending the synthetic antenna's length, Spotlight mode enhances cross-range resolution, making it ideal for applications requiring maximum resolution whilst accepting reduced spatial coverage [5].

ScanSAR Mode

ScanSAR mode provides improved spatial coverage compared to both SM and Spotlight modes, although at the cost of reduced cross-range resolution [74]. In ScanSAR, the antenna periodically sweeps over multiple sub-swaths associated with various antenna orientations [5, 74], as shown in Figure 3.4c. Each sub-swath is illuminated by multiple chirps but for a shorter duration compared to SM mode [5]. The loss in resolution is due to the fact that the synthetic antenna

length is divided up across sub-swaths [74].

TOPSAR Mode

TOPSAR is a specialised form of ScanSAR imaging used by **S1A** in both **IW** and **EW** modes. It acquires data by transmitting bursts and periodically switching the antenna beam between adjacent sub-swaths, as illustrated in Figure 3.4d. While ScanSAR uses mechanical or electronic beam steering, **TOPSAR** exclusively uses electronic beam steering in both down- and cross-range directions. **TOPSAR** offers the same spatial coverage as ScanSAR but with marginally improved cross-range resolution due to shorter burst times [76]. Consequently, **TOPSAR** is ideal for applications demanding high resolution and extensive swath coverage.

S1A's **IW** mode uses **TOPSAR** imaging, retaining wide swath coverage and improved cross-range resolution [8]. **IW** mode consists of three wide sub-swaths captured by steering the antenna in the cross-range direction by an angle ω_r [8]. On the other hand, **EW** mode, also based on **TOPSAR** imaging, differs in the number of sub-swaths used for complete scene capture. While **IW** mode utilises three sub-swaths, **EW** mode employs five wide sub-swaths to construct the entire image, effectively enhancing spatial coverage while maintaining cross-range resolution.

Table 3.1 provides the ground resolution of all **S1A** imaging modes for **GRD** data.

Mode	Minimum Ground Swath Width [km]	Resolution (dr x cr) [m]
SM	80	10x10
IW	250	10x10
EW	410	25x25

Table 3.1: Comparison of ground swath width and resolution of **S1A** imaging modes. Adapted from [8].

Due to the fact that this project requires imaging ocean waves, fine-range resolution is required. To this end, data products captured using **SM** or **IW** mode are utilised due to their improved resolution over **EW** mode depicted in Table 3.1.

3.2.3 Pre-processing Techniques

Pre-processing techniques applied to **S1A GRD** data, as well as the **SAR** phenomena which these techniques intent to mitigate, are discussed in Appendix A. In terms of pre-processing techniques used in the implementation of the **HH** technique in this report, these were thermal noise removal, and radiometric calibration, which are respectively discussed in Appendix A.1.

3.3 Hasselmann and Hasselmann Ocean Wave Inversion Technique

The **HH** process maps an ocean wave spectrum, as described in Section 3.1.3, into a **SAR** spectrum. The **HH** inversion technique consists of two main parts - the **SAR** imaging of ocean waves, followed by an inversion to extract a wave spectrum.

3.3.1 SAR Imaging of Ocean Waves

HH [59] state that the waves captured by a SAR spectrum are modulated through three different modulation processes. Firstly, the change in incidence angle is due to the change in the slope of the wave face. Secondly, the interactions between short and long waves, which modulate the energy and wave number of the short, Bragg scattering, ripple waves. And thirdly, the orbital velocity of the long waves, which produces a Doppler shift in the received, return signal. This causes an azimuthal displacement of the scatterers in the SAR image. This third modulation process is known as velocity bunching and is the main cause of non-linearity in the imaging of ocean waves. All three of these modulation processes can be represented by their respective modulation transfer functions, and are denoted by the general form, T_k^x .

To derive a SAR spectrum from an ocean wave spectrum, HH treat the captured SAR image as two separate procedures, namely the frozen surface, defined by the RAR imaging mechanism, and the motion effects, related to SAR imaging. The reason these motion effects arise is due to the way in which a SAR image is captured, as discussed in Section 3.2.2.

Frozen Surface Contribution

The RAR MTF [59] is defined as

$$T_k^R = T_k^t + T_k^h \quad (3.43)$$

where T_k^{t9} and T_k^h are the tilt and hydrodynamic interaction MTFs respectively [59].

$$\text{For VV Polarisation: } T_k^t = 4ik_l \cot(\theta_i) (1 + \sin^2 \theta_i)^{-1} \quad (3.44a)$$

$$\text{For HH Polarisation: } T_k^t = 8ik_l (\sin 2\theta_i)^{-1} \quad (3.44b)$$

Where θ_i represents the radar incidence angle and k_l represents the component of the wave-number vector in the radar look direction. These coordinates are chosen such that the x -axis represents the SAR satellite flight direction, and the y -axis points in the positive or negative look direction, l , for a left or right looking SAR satellite respectively [59].

$$T_k^h = \frac{\omega - i\mu}{\omega^2 + \mu^2} (4.5) k \omega \left(\frac{k_y^2}{k^2} \right) \quad (3.45)$$

Where k is the wave number vector and $\omega = \sqrt{gk}$.

Motion Effects

The range velocity MTF [59] is defined as

⁹Polarisation refers to the orientation of the plane in which an EM wave oscillates as it propagates. More theoretical background on polarisation can be found in Appendix A.2.

$$T_k^v = -\omega \left(\sin(\theta_i) \frac{k_l}{|k|} + i \cos(\theta_i) \right) \quad (3.46)$$

and the velocity bunching MTF [59] is defined as

$$T_k^{vb} = -i\beta k_x T_k^v \quad (3.47)$$

where β is the ratio of slant range, R , and satellite velocity, U ; $\beta = R/U$ [59]. Defining the velocity bunching MTF allows the net SAR imaging MTF [59] to be defined as

$$T_k^S = T_k^R + T_k^{vb} \quad (3.48)$$

Co/Autocovariance Functions

The MTFs described above allow three covariance functions to be defined. These are the orbital velocity covariance function, $f^v(\underline{r})$, the autocovariance function of RAR image intensity, $f^R(\underline{r})$, and the covariance function of RAR image intensity and non-linear velocity, $f^{Rv}(\underline{r})$ [59].

$$f^v(\underline{r}) = \int E(\underline{k}) |T_k^v|^2 e^{i\underline{k} \cdot \underline{r}} d\underline{k} \quad (3.49)$$

$$f^R(\underline{r}) = \frac{1}{2} \int \left[E(\underline{k}) |T_k^R|^2 + E(-\underline{k}) |T_{-\underline{k}}^R|^2 \right] e^{i\underline{k} \cdot \underline{r}} d\underline{k} \quad (3.50)$$

$$f^{Rv}(\underline{r}) = \frac{1}{2} \int \left[E(\underline{k}) T_k^R (T_k^v)^* + E(-\underline{k}) T_{-\underline{k}}^R (T_{-\underline{k}}^v)^* \right] e^{i\underline{k} \cdot \underline{r}} d\underline{k} \quad (3.51)$$

Spectral Expansion

The functions described in Equations 3.49, 3.50 and 3.51 can, using the Fourier Transform, be expanded into spectral expansion terms using the following equations. Ω_n is the Fourier Transform operator given by $\frac{1}{4\pi} \int d\underline{r} e^{-i\underline{k} \cdot \underline{r}}$ [59], which represents a two-dimensional Fast Fourier Transform (FFT).

$$P_{n,2n}^S = \Omega_n \left\{ \frac{f^v(\underline{r})^n}{n!} \right\} \quad (3.52)$$

$$P_{n,2n-1}^S = \Omega_n \left\{ \frac{i [f^{Rv}(\underline{r}) - f^{Rv}(-\underline{r})] f^v(\underline{r})^{n-1}}{(n-1)!} \right\} \quad (3.53)$$

$$P_{n,2n-2}^S = \Omega_n \left\{ \frac{1}{(n-1)!} f^R(\underline{r}) f^v(\underline{r})^{n-1} + \frac{1}{(n-2)!} [f^{Rv}(\underline{r}) - f^{Rv}(0)] \cdot [f^{Rv}(-\underline{r}) - f^{Rv}(0)] f'(\underline{r})^{n-2} \right\} \quad (3.54)$$

To calculate the full spectral expansion, these three equations can be combined as

$$P^S(\underline{k}) = \exp(-k_x^2 \xi'^2) \sum_{n=1}^{\infty} \sum_{m=2n-2}^{2n} (k_x \beta)^m P_{n,m}^S(\underline{k}) \quad (3.55)$$

The term $\exp(-k_x^2 \xi'^2)$ is known as the quasilinear coefficient [59], where ξ' is defined as the azimuthal displacement. $\xi' = \beta \langle v \rangle$. $\langle v \rangle$ is defined as the time average over the period of viewing the scene. ξ'^2 can also be determined using the following equation [47, 59]

$$\xi'^2 = \beta^2 \int |T_k^v|^2 E(\underline{k}) d\underline{k} \quad (3.56)$$

n in Equation 3.55 represents the non-linearity order of the ocean waves. More information on the non-linearity of ocean waves can be found in Chapter 6.4.4 and Chapter 7 of [4].

3.3.2 inversion technique

The HH inversion [59] requires the minimisation of a cost function [59] with respect to the generated SAR spectrum and first-guess wave spectrum, $\hat{E}(\underline{k})$. The cost function is given as

$$J_{cost} = \int \left[P(\underline{k}) - \hat{P}(\underline{k}) \right]^2 d\underline{k} + \mu \int \left[\frac{E(\underline{k}) - \hat{E}(\underline{k})}{B + \hat{E}(\underline{k})} \right]^2 d\underline{k} \quad (3.57)$$

where μ represents the confidence between the observed SAR spectrum and $\hat{E}(\underline{k})$. μ is set to $0.1\hat{P}_{max}^2$ and B is set to $0.01\hat{E}_{max}$ [59].

The first estimate, $E_1(\underline{k}) = \hat{E}(\underline{k})$, where $E_n(\underline{k})$ and $P_n(\underline{k})$ represent the approximate solution after n iterations. This allows E_{n+1} and P_{n+1} to be defined as

$$E_{n+1} = E_n + \Delta E_n \quad (3.58)$$

$$P_{n+1} = P_n + \Delta P_n \quad (3.59)$$

where $\Delta P_n(\underline{k})$ is defined as

$$\Delta P_n(\underline{k}) = \frac{1}{2} \exp[-k_x^2 \xi_n'^2] \left(|T_k^S|^2 \Delta E_n(\underline{k}) + |T_{-k}^S|^2 \Delta E_n(-\underline{k}) \right) \quad (3.60)$$

Equation 3.60 can be substituted into Equation 3.57 to obtain

$$J_{cost} = \int \left[\Delta P_n - (\hat{P} - P_n) \right]^2 d\underline{k} + \mu \int \left[\Delta E_n - (\hat{E} - E_n) \right]^2 d\underline{k} \quad (3.61)$$

where the solution of Equation 3.61 allows ΔE_n to be defined as [59]

$$\Delta E_n = \frac{A_{-k}(W_k \delta P + \mu \delta E_k) - B_k(W_{-k} \delta P + \mu \delta E_{-k})}{[A_k A_{-k} - B_k^2]} \quad (3.62)$$

where the variables defined in Equation 3.62 are given in Appendix A.3.

Chapter 4

Pipeline Design

This chapter details the core design process of this project, a wave parameter extraction pipeline. The fundamental design which needed to be achieved was the extraction of wave parameters from [SAR](#) data. To achieve this, the [HH](#) technique was implemented. The overarching idea of the [HH](#) technique requires an input first-guess wave spectrum, as well as the observed [SAR](#) data. These data are used to minimise a cost function which outputs the observed [SWH](#) and [SWP](#).

This chapter begins by providing an overview of the entire pipeline - part of which falls outside of the scope of this project, however, this context is relevant in terms of providing context to the desired results for this project in the context of the broader goals of this pipeline. Each subsequent section unpacks the five main sub-modules within the pipeline in a respective section. Each of these sections further breaks down the sub-module into smaller blocks. The use of block diagrams allows the process to be understood in terms of the flow of the pipeline, and to break down higher-level processes. The variable and function names utilised in this chapter, match those used in the MATLAB pipeline.

4.1 Overview

As detailed in Section 1.4, this project required wave parameter extraction for use in a sea ice parameter extraction pipeline. This entire system design is shown as a block diagram in Figure 4.1. It is evident from Figure 4.1, which parts of the [SAR](#) sea ice parameter are relevant to this project. A more detailed block diagram of the wave parameter extraction process is shown in Figure 4.2. Within this pipeline, five main functional blocks were identified and are highlighted by five unique colours. MATLAB structure arrays¹ were utilised throughout the entirety of this project to store multi-dimensional data associated with a certain value. As such, all design decisions were made to generate MATLAB structures of data.

¹In MATLAB, a structure array is a type of data that organises related data into groups. These groups are indexed using an associated field value which can hold any particular data type. More information on structure arrays can be found in the [MATLAB documentation](#). The rest of this report will refer to structure arrays simply as structures.

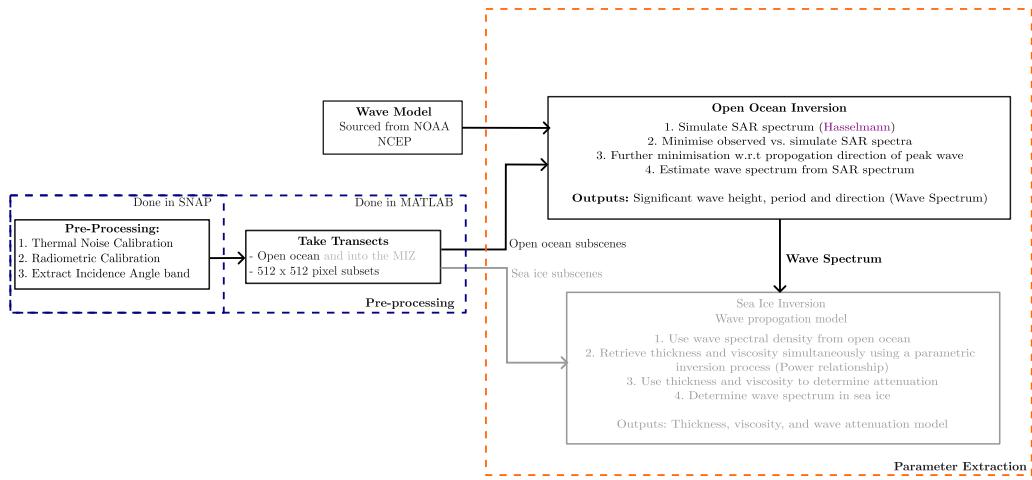


Figure 4.1: Pipeline system design in the context of the entire parameter extraction process. The scope of this project is shown in black text, with parts of the pipeline outside of the scope, shown in grey.

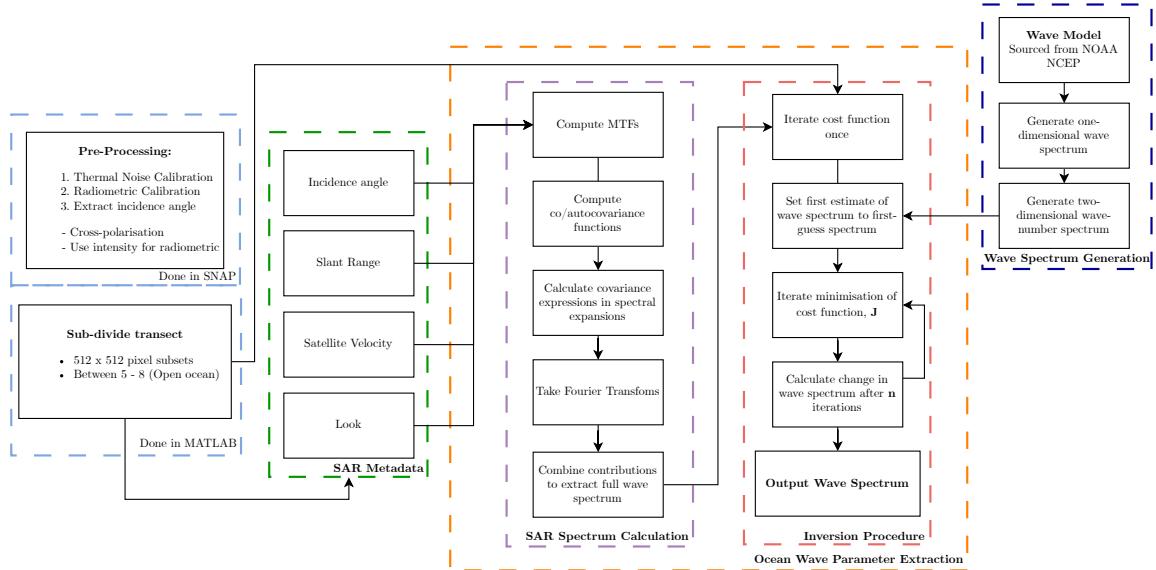


Figure 4.2: Pipeline system design for the scope of this project broken down further than the entire pipeline shown in Figure 4.1. All the required metadata and external models are shown, along with the part of the process at which they are required.

The five functional blocks were identified as: pre-processing, metadata extraction, **SAR** spectrum calculation, wave spectrum generation, and inversion. The pre-processing block involved the use of the SeNtinel Applications Platform (**SNAP**) Toolbox developed by **ESA**, as well as MATLAB, developed by MathWorks Inc. The pre-processing block read in a downloaded **SAR** Level-1 Ground Range Detected (**GRD**) data file and outputs a MATLAB array of n equal-sized transects. These equal-sized transects were input to the inversion block. The output Network Common Data Form (**NetCDF**) file from **SNAP** was used to generate views of **SAR** data in MATLAB, as well as storing the metadata for the metadata extraction block. The **NetCDF** file was read by the metadata extraction block and outputs the desired metadata values required for the **SAR** spectrum calculation block. The **SAR** spectrum calculation block read in a wave spectrum generated by the wave spectrum generation block, with data sourced from **NOAA (NCEP)**, and generated a **SAR** spectrum of the generated wave spectrum. This generated **SAR**

spectrum was input to the inversion block along with the output of the metadata extraction block. The inversion block outputs the wave parameters from the input pre-processed SAR data.

4.2 Pre-processing

The pre-processing block of the pipeline was designed in a hybrid manner, with part of the pre-processing done in the [SNAP](#) toolbox developed by [ESA](#) prior to the use of MATLAB. The pre-processing block was designed to have an overall function of reading in [S1A](#) data, apply the desired pre-processing techniques to the [SAR](#) data, and take 512x512 pixel sized transects of this larger [SAR](#) data. The applied pre-processing techniques to calibrate [SAR](#) data for wave parameter extraction were on the recommendation of Giacomo De Carolis and Francesca De Santi of the [irea!](#), however, future implementations of this pipeline should investigate the use of different processing techniques, as discussed in Section 7.1. This block was designed to keep pre-processing outside of the MATLAB environment to a minimum.

The pre-processing block had six notable stages: Thermal noise calibration of [S1A GRD](#) data, radiometric calibration of these data, extraction of individual pixel incidence angle, exporting calibrated [GRD](#) data to a usable format for MATLAB, reading in the exported data, and taking 512x512 pixel transects of the larger [SAR](#) data. Thermal noise calibration, radiometric calibration, and incidence angle extraction were all implemented using [SNAP](#) Desktop and are not discussed further. Details of these implementations can be found in [pipeline.mlx](#). The flow of this block of the pipeline is shown graphically in Figure 4.3.

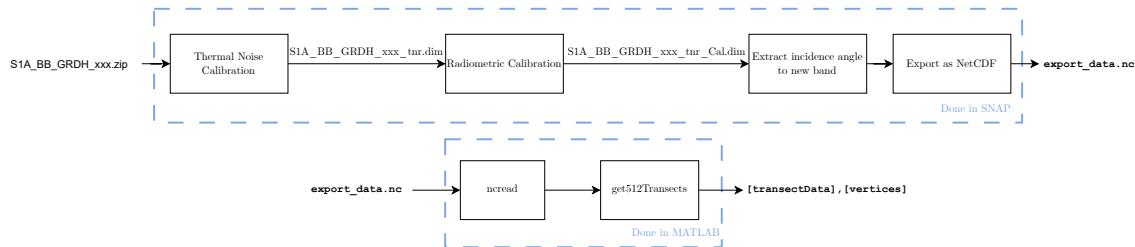


Figure 4.3: Block diagram depicting an expanded pre-processing sub-block of Figure 4.2.

4.2.1 NetCDF Export

[SNAP](#) Desktop allows a multitude of export options for pre-processed data. The choice of export data type was narrowed down to data formats which contained metadata, as well as the [SAR](#) data. A list of these formats is shown below.

- GeoTIFF
- BEAM-DIMAP
- ENvironment for Visualizing Images ([ENVI](#))
- Hierarchical Data Format ([HDF](#))5
- [NetCDF](#)4

To decide which file format to export data as from **SNAP** Desktop, an analysis of the compatibility of each file format with MATLAB was done and the following was found.

A GeoTIFF file can be read into MATLAB using the `readgeoraster` function built into the Mapping Toolbox provided by MATLAB. BEAM-DIMAP is a file format used by **SNAP** products, and cannot be imported to MATLAB. An **ENVI** file can be read into MATLAB to extract metadata using the `enviinfo` function built into the Image Processing Toolbox by MATLAB. A **HDF5** file can be displayed and read in MATLAB using a variety of functions built into MATLAB such as the `h5disp` and `h5read` functions. Finally, a **NetCDF4** file can, as with **HDF5** files, be displayed and read in MATLAB using a variety of functions built into MATLAB.

Of the 5 export file types, only BEAM-DIMAP cannot be read into MATLAB. Due to the desire to only use built-in MATLAB functions and not make use of external toolboxes, this left **HDF5** and **NetCDF4** as the two file types to be chosen from. Both file formats were able to be loaded as a structure in MATLAB, and as such were both desirable choices. However, exporting **SAR** data from **SNAP** was faster when exporting as **NetCDF** as opposed to **HDF**. Furthermore, **NCEP** data was downloaded as a Grib2 file, which could be converted to a **NetCDF** file, and in order to keep consistency, **NetCDF** was chosen as the file type to export from **SNAP** Desktop.

4.2.2 MATLAB Data Import

The first sub-block in the MATLAB implementation of the pre-processing block was `ncread`. As discussed in Section 4.2.1, the **NetCDF** file format was used for all imported data into MATLAB. The `ncread` MATLAB function was used to import the exported data from **SNAP** Desktop.

To access different bands within the **NetCDF** file, different variable names were used as a second input parameter to the `ncread` functions. Listing 4.1 details the required variable names to extract the two bands of interest required for each subsequent block, and sub-block in Figure 4.2.

```

1 filepath = "D:\UCT\EEE4022S\Data\CPT\export_data.nc";
2 sarData = ncread(filepath, 'Sigma0_VV'); % Intensity VV pre-processed
    ↪ data
3 incidenceAngle = ncread(filepath, 'Incidence_Angle'); % Incidence
    ↪ Angle band

```

Listing 4.1: MATLAB code used to import all desired bands from exported **NetCDF** data from **SNAP** Desktop.

By utilising `ncread` to extract data, this assigned both `sarData` and `incidenceAngle` variables to a $n \times m$ sized matrix, where n and m represent the size of the **SAR** data in terms of cross-range and down-range directions respectively. The extraction of metadata is detailed in Section 4.3.1.

4.2.3 Data Transects

The most important part of the pre-processing for wave parameter extraction for use in sea ice parameter extraction is the ability to 'follow' the flow of ocean waves into the **MIZ**. To achieve this, transects of a fixed size are taken of open ocean scenes as well as sea ice scenes. This allows the way in which ocean waves propagate through sea ice to be studied, and consequently, sea ice parameters are able to be extracted using the models described in Section 2.3.3. As per [65, 68]

the size of these transects should be 512x512 pixels in size. Taking transects of a larger image in MATLAB is relatively straightforward, however, these transects needed to be able to be taken at an angle to follow the direction of the visible ocean waves. If the ocean wave direction is not followed, this could cause issues in determining the way in which ocean waves are attenuated in the MIZ. Along with this, the transect needed to start at a certain location of the whole scene. Due to the fact that latitude and longitude are not encoded in the [S1A GRD](#) data, it was decided that a pixel starting position would be used to determine the starting position.

The way angled transects were designed to be taken is visually represented in Figure 4.4.

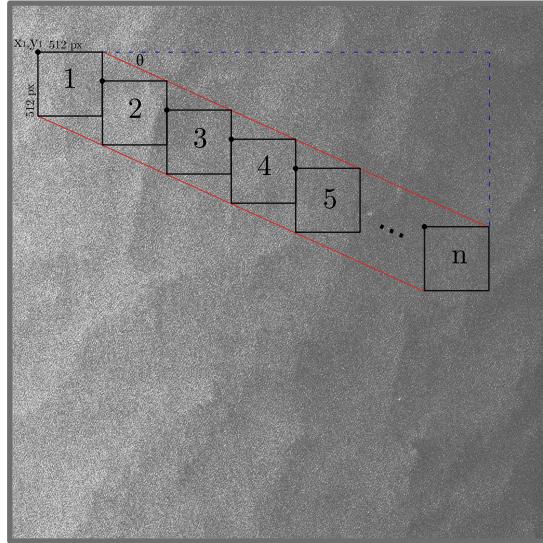


Figure 4.4: Implemented geometry for taking transects of full [SAR](#) scene.

θ , in Figure 4.4, represents the user-supplied angle at which to take the n number of 512x512 transects measured clockwise from the $+x$ -axis. x_1, y_1 represent the user-supplied pixel starting location at which transects should be taken from. The `get512Transects` function was designed to incorporate all of these design decisions, and as a result, takes in the full scene of [SAR](#) data, the top left x and y coordinates, as well as the user-defined angle, θ and number of transects, n . `get512Transects` returned a 512x512xn array of the transects of the full scene, along with a nx4 matrix of the corner pixel values of each transect. As well as using the `get512Transects` function to take transects of the [SAR](#) image, the function was also used to take transects of the `incidenceAngle` variable determined in Listing 4.1. The returned variable for `incidenceAngle` followed the same form as discussed above for [SAR](#) data.

To annotate the transects, the `annotate512Transects` function was designed which allowed the user to control the colour of the transect outline, as well as text colour and need for a background on the text colour. Using both the `get512Transects` and `annotate512Transects` functions allowed the plots in Figure 4.5 to be generated. The ability to view individual transects was desired to be able to follow the direction of waves within the [SAR](#) scene.

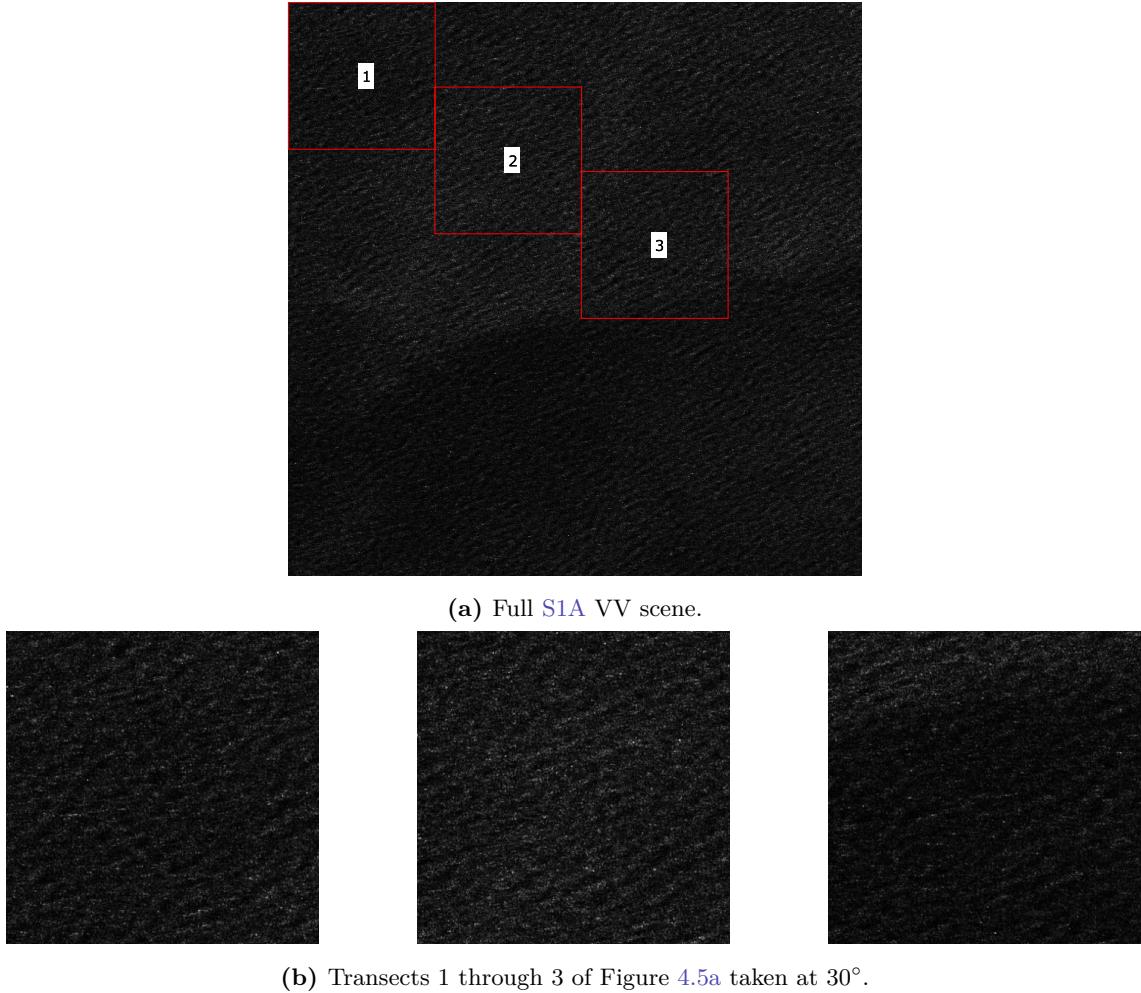


Figure 4.5: S1A Intensity VV data, pre-processed according to Section 4.2, from Oct. 2, 2023 at 34.78S, 16.77E with transects at 30° for $x_1, y_1 = 1, 1$ displayed and labelled. Respective individual transects are shown.

4.3 Metadata Extraction

The metadata extraction block of the pipeline was designed to allow desired metadata values to be extracted from SAR data. The metadata extraction block was designed to take in an exported NetCDF file from SNAP Desktop and extract the desired required attributes. The metadata extraction block had two notable stages: Data import of NetCDF data into a MATLAB structure, and filtering this structure to an array containing only the values of interest. The flow of this block of the pipeline is shown graphically in Figure 4.6.

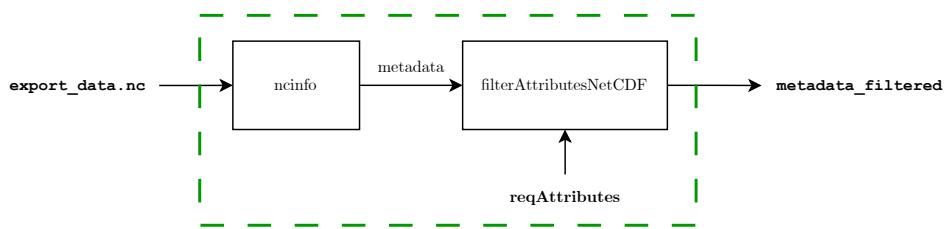


Figure 4.6: Block diagram depicting an expanded metadata extraction sub-block of Figure 4.2.

4.3.1 Data Import

The first sub-block of the metadata extraction block was called `ncinfo`. As discussed in Section 4.2.1, the `NetCDF` file format was used for importing data into MATLAB. The `ncinfo` MATLAB function was used to import the exported data from `SNAP` Desktop.

To access the exported metadata within the `NetCDF` file, the `ncinfo` MATLAB function was used. Listing 4.2 details the required variable name to extract the metadata for the subsequent sub-block in Figure 4.6.

```
1 filepath = "D:\UCT\EEE4022S\Data\CPT\export_data.nc";
2 metadata = ncinfo(filepath, 'metadata'); % Extract all metadata
```

Listing 4.2: MATLAB code used to import metadata from exported `NetCDF` data from `SNAP` Desktop.

Using `ncinfo` to extract metadata, assigns the `metadata` variable to a 1x1 structure with all desired metadata in the `Attributes` field. This field was parsed to the `filterAttributesNetCDF` sub-block.

4.3.2 Filter Attributes

The `filterAttributesNetCDF` block took in a structure containing metadata, as well as a list of strings of the desired attributes to keep from all metadata values. Exact attribute names can be found in Table D.1 in Appendix D. Due to the way that the `NetCDF` file imported metadata into MATLAB, all data types were prefixed with `Abstracted_Metadata`. Due to this, the `filterAttributesNetCDF` function added this prefix to all input required attribute strings to avoid the desired metadata values not being found. The function outputs a 1xlength(`reqAttributes`) structure with two columns: Name and Value. Additional metadata formatting functions are discussed in Section 4.5 and 4.7 when required by respective `MTFs`.

4.4 Wave Spectrum Generation

The implementation of a first-guess wave spectrum is vital in the implementation of the `HH` technique. Therefore, the generation of a first-guess wave spectrum of the sea state is vital to the implementation of this project. The wave spectrum generation block of the pipeline was designed with a modular approach. The block read in the `SAR` metadata, and downloaded and generated three different types of wave spectrum: A one-dimensional `JONSWAP` spectrum, a two-dimensional frequency-direction wave spectrum, and a two-dimensional wave-number spectrum. The two-dimensional wave-number spectrum was the output of the block and was an input to the inversion block detailed in Section 4.7.

The wave spectrum generation block had ten stages, which could be divided into six notable stages: Downloading the associated wave data based on the `SAR` metadata, converting the downloaded data into `NetCDF` format, extracting a subset of the wave data surrounding the region of interest, generating a one-dimensional wave spectrum, generating a two-dimensional frequency-direction wave spectrum, and generating a two-dimensional wave-number spectrum. The flow of this block of the pipeline is shown graphically in Figure 4.7.

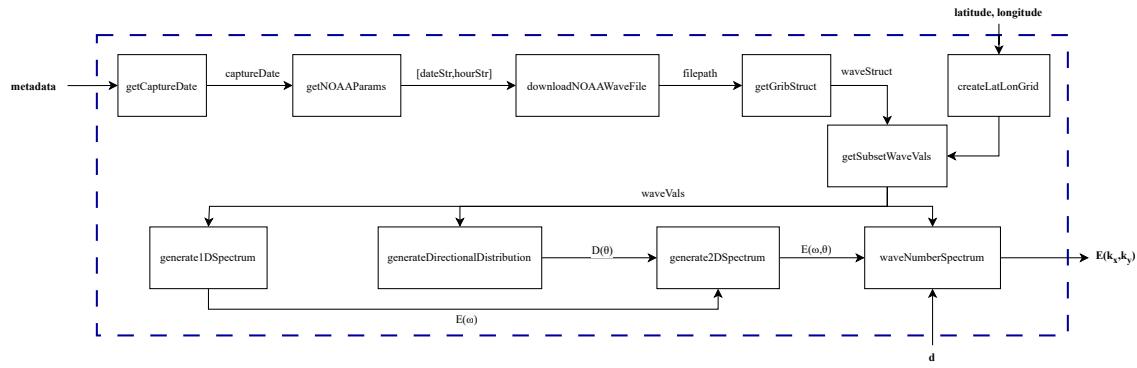


Figure 4.7: Block diagram depicting an expanded wave spectrum generation sub-block of Figure 4.2.

4.4.1 Wave Data Download

NCEP NOAA wave data is available in 6-hour intervals. These intervals are 00, 06, 12, and 18 hours in UTC time. As S1A captures are not set to exact time intervals, a function `getNOAAParams` was created which took in the S1A capture datetime object, obtained using the `getCaptureDate` function, and outputted two strings. Both of these strings were formatted as per the URL used to download NCEP wave data.

The way in which the closest hour value was determined is shown in Listing 4.3.

```

1 noaaHour = datetime(SARCaptureDate,"Format","HH:mm");
2 noaaHour = double(hour(noaaHour));
3 allowedHours = [0,6,12,18];
4 hourDiff = abs(allowedHours - noaaHour);
5 [~,hourIndex] = min(hourDiff);
6 nearestHour = allowedHours(hourIndex);

```

Listing 4.3: Initial implementation of MATLAB code to extract wave parameters at certain geographical location.

The date of the S1A capture was extracted using the `filterAttributesNetCDF` function and MATLAB's built-in `datetime` function. The form of the date and hour value required by NCEP was of the form YYYYMMDD and HH respectively. To achieve this, the `getNOAAParams` function used basic string handling after extracting all individual values. Interestingly, isolating a month or day from a datetime object with a numeric value < 10 in MATLAB, resulted in a single character. As NCEP required two characters for both of these parameters, a simple `if` statement checked the length of the value, and updated them accordingly if they had length 1 by prepending a 0 to the front of the single character value.

These functions allowed associated wave data to be downloaded, using the `downloadNOAAWaveFile` function, for the corresponding S1A data.

4.4.2 GRIB2 to NetCDF Conversion

Downloaded wave data from NCEP was in the form of a .grib2 file. The GRIBded Binary (GRIB)2 file format was not able to be imported natively into MATLAB, and due to this, an

investigation was done into converting a **GRIB2** file into a **NetCDF** file format. Fortunately, **NOAA NCEP** have released a utility tool known as **wgrib2**, which can convert **GRIB** files into **NetCDF** files whilst maintaining the integrity of the gridded data contained within the **GRIB** file.

To use **wgrib2** in MATLAB, the **getGribStruct** function was used, which took in the path of the location of the installed **wgrib2** executable, as well as the path to the downloaded **GRIB2** wave data file. The key elements of code used to implement this conversion are shown in Listing 4.4 which parses a command to the **wgrib2** executable in the required form to convert the **GRIB2** file into a **NetCDF** file after checking that the file to write does not exist.

```

1 if ~exist('output_file','file')
2     command = sprintf('%s "%s" -netcdf "%s" >nul 2>&1',
3         ↳ wgrib2_executable, file_path, output_file);
4     system(command); % Convert grib2 to netcdf
5 end

```

Listing 4.4: MATLAB code used to parse commands to the **wgrib2** executable for conversion of **GRIB** data into **NetCDF** format.

After the conversion shown in Listing 4.4, the rest of the **getGribStruct** function extracts the following parameters using the **ncread** function: latitude, longitude, time, significant swell wave height, significant swell wave period, wave direction, wind speed, and wind direction. These data were stored in a MATLAB structure. This structure was then fed into the **getSubsetWaveVals** function after defining the location of the region of interest.

4.4.3 Defining Location

The **createLatLonGrid** sub-block in Figure 4.7 is vital to the generation of wave spectra due to the limitation of the resolution of **NCEP** wave data. **NCEP** wave data had 0.25 degree resolution in both latitude and longitude directions, and due to the fact that the region of interest, may not be located at the points available in **NCEP** wave data, a grid of latitude and longitude coordinates needed to be generated surrounding the point of interest.

This was achieved in MATLAB using the **createLatLonGrid** function which created two arrays of length 3, with the three closest latitude and longitude values based on the provided latitude, longitude, and resolution of wave data. This allowed surrounding wave data to be extracted as detailed in Section 4.4.4. A sample output of running Listing 4.5 is shown in Listing 4.6.

```

1 latitude = 34.20;      % Input latitude
2 longitude = -18.28;    % Input longitude
3 resolution = 0.25;    % Resolution in degrees
4 [grid_lat, grid_lon] = createLatLonGrid(latitude, longitude,
   ↳ resolution);

```

Listing 4.5: MATLAB code used to define grid values of latitude and longitude.

Listing 4.6 displays a 1x3 array where the centre value represented the closest value to the given **latitude** and **longitude** in Listing 4.5. These arrays were used in the **extractWaveParams** sub-block to define the location over which to extract wave parameters.

```

1 grid_lat =
2
3     34.0000    34.2500    34.5000
4
5 grid_lon =
6
7     -18.5000   -18.2500   -18.0000

```

Listing 4.6: MATLAB output of Listing 4.5.

4.4.4 Wave Parameter Extraction

When examining the latitude and longitude values obtained from [NCEP](#), it was noted that these data points contained decimal points of order 10^{-6} . This proved an issue when trying to index the original downloaded wave data to filter a subset of these data and resulted in none of the desired data being found.

To mitigate this, an investigation into the resolution of decimal degrees was conducted to match the spatial resolution of wave data to the corresponding [SAR](#) resolution. It was found that 5 decimal places allowed a worst-case resolution of 1.11 m [77]. In the case of [S1A](#), such resolution was adequate, due to the satellite's highest spatial resolution of 9x9 m resolution for [GRD](#) data [8]. Using this information, the code in Listing 4.7 was used to accurately index spatial coordinates without the loss of spatial resolution.

```

1 lonVal = 45.25;
2 lonIndex = find(round(lon,5) == lonVal);

```

Listing 4.7: Updated implementation of MATLAB code to extract wave parameters at certain geographical location.

The `getSubsetWaveVals` function implemented the code seen in Listing 4.7, as well as then assigning the extract wave parameters to a new structure containing subsets of all data obtained using the `getGribStruct` function.

4.4.5 One Dimensional Wave Spectrum

Whilst the generation of a one-dimensional wave spectrum is only represented by one sub-block in Figure 4.7, the generation of this spectrum is vital to forming the foundation of the two-dimensional wave-number spectrum which was used as a first-guess wave spectrum when minimising the cost function described in Equation 3.57. As discussed in Section 3.1.6, the most widely used wave model is the [JONSWAP](#) model. Implementing this spectrum required a careful approach to ensure accurate modelling of waves was done using the downloaded wave data detailed in Section 4.4.1.

To construct a [JONSWAP](#) wave model, $E(\omega)$, the equations detailed in Section 3.1.6 were implemented in MATLAB, as outlined in the pseudocode, given in Algorithm 1 where $H_{1/3}$ and w_{peak} are wave parameters as detailed in Section 3.1.6 which are obtained using the `getSubsetWaveVals` function. ω is the range of frequencies over which to generate the wave spectrum. The peak enhancement factor for a region is determined experimentally. Gweba [78] determined the

peak enhancement factor, γ , at the location of all Council for Scientific and Industrial Research ([CSIR](#)) wave buoys in their Masters dissertation.

```

input      : Wave parameters,  $H_{1/3}$ ,  $\omega_{peak}$ 
              Frequency range,  $\omega$ 
output     : JONSWAP wave spectrum,  $E(\omega)$ 
parameter: Peak enhancement factor,  $\gamma$ 

begin
     $g \leftarrow 9.81$ 
     $\alpha \leftarrow 0.2 \cdot H_{1/3}^2 \cdot \frac{\omega_{peak}^4}{g^2}$ 
    if  $\gamma < 1$  or  $\gamma > 7$  then
         $k \leftarrow \frac{2\pi}{\omega_0 \sqrt{H_s}}$ 
        if  $k \leq 3.6$  then
             $\gamma \leftarrow 5$ 
        end
        if  $k \leq 5$  then
             $\gamma \leftarrow \exp(5.75 - 1.15 \cdot k)$ 
        else
             $\gamma \leftarrow 1$ 
        end
    end
    for  $k$  in 1 to  $length(\omega)$  do
        if  $\omega(k) < \omega_{peak}$  then
             $\sigma \leftarrow 0.07$ 
        end
        else
             $\sigma \leftarrow 0.09$ 
        end
         $E1 \leftarrow \alpha \cdot g^2 \cdot (\omega(k)^{-5}) \cdot \exp\left(-\frac{5}{4} \cdot \left(\frac{\omega_{peak}}{\omega(k)}\right)^4\right)$ 
         $\text{exponent} \leftarrow \exp\left(-\frac{(\omega(k)-\omega_0)^2}{2 \cdot (\sigma \cdot \omega_0)^2}\right)$ 
         $E2 \leftarrow \gamma^{\text{exponent}}$ 
         $E \leftarrow E1 \cdot E2$ 
    end
end

```

Algorithm 1: JONSWAP wave spectrum generation

Extending Algorithm 1 to allow multiple spectra to be calculated at multiple latitudes and longitudes allowed the validity of the spectrum generation to be validated, as plotting multiple wave spectra on the same set of axes allowed the way in which the wave spectra change as they approach the shore to be determined. This analysis is discussed in Section 5.2.1 and example plots obtained using the `generateSingleJONSWAP` and `generateMultipleJONSWAP` functions are shown in Figure 4.8.

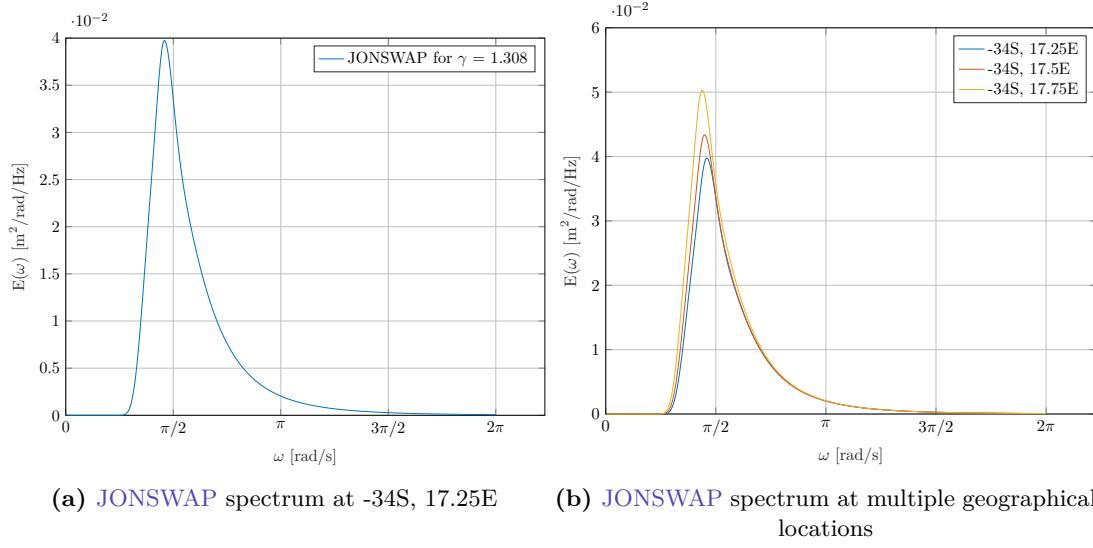


Figure 4.8: One-dimensional JONSWAP wave spectra, $E(\omega)$, generated using NCEP wave data.

4.4.6 Two-dimensional Wave Spectrum

To extend the one-dimensional wave spectrum into a two-dimensional wave spectrum, a directional distribution function needed to be applied to the one-dimensional wave spectrum. This is represented in Figure 4.7 with the `generateDirectionalDistribution` block, which took in the `SWH` obtained from `NCEP`, as well as the whole range of directions over which to define the directional distribution function. The calculation of the directional distribution function followed the definition of the $\cos^2(\theta)$ model, as detailed in Equations 3.33, 3.34, and 3.35 in Section 3.1.6. A plot of the directional distribution function generated using the `generateDirectionalDistribution` function is shown in Figure 4.9.

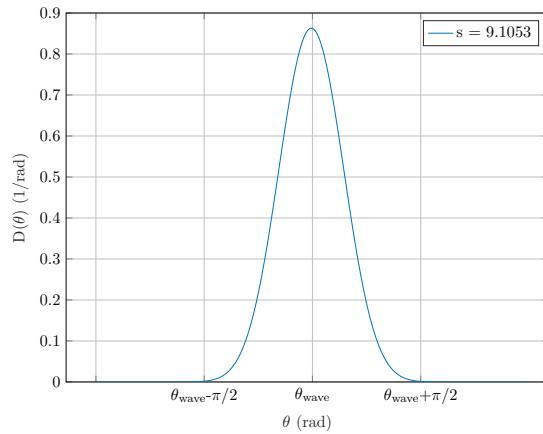


Figure 4.9: Directional distribution function, $D(\theta)$, defined using the $\cos^2(\theta)$ model detailed in Section 3.1.6.

In accordance with Equation 3.32, the two-dimensional frequency-direction wave spectrum can be obtained by multiplying the one-dimensional wave spectrum, $E(\omega)$, with the directional distribution function, $D(\theta)$. This was achieved using the `generate2DWaveSpectrum` block, which took in the one-dimensional wave spectrum, and the directional distribution function and simply

performed an element-wise multiplication using MATLAB's built-in 'dot' multiplication. This resulted in the two-dimensional wave spectrum shown in Figure 4.10

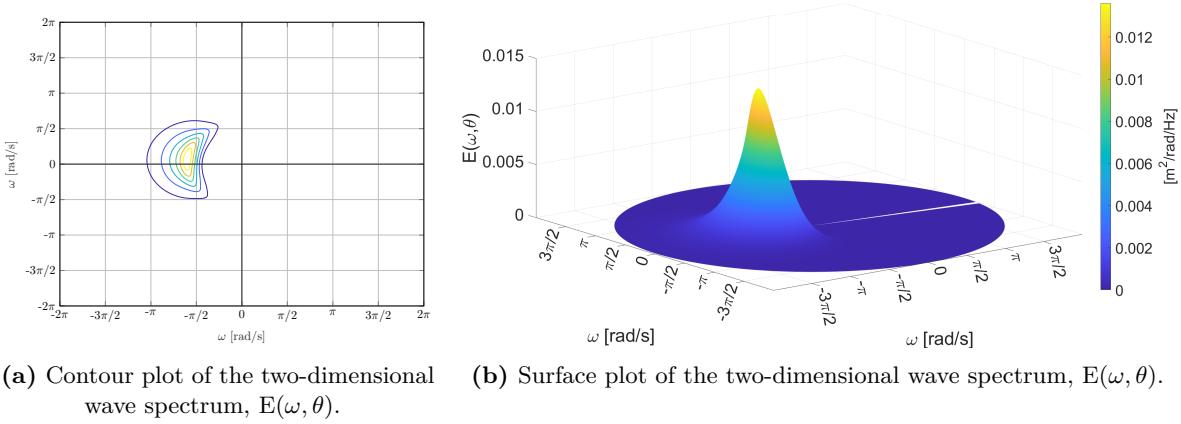


Figure 4.10: Two-dimensional frequency-direction wave spectra, $E(\omega, \theta)$, generated using NCEP wave data, and the directional distribution function shown in Figure 4.9.

4.4.7 Wave-number Spectrum

Converting the two-dimensional wave spectrum generated in Section 4.4.6 to a two-dimensional wave-number spectrum for use in the HH procedure was achieved using the `waveNumberSpectrum` ↗ function and required implementing Equation 3.17 in MATLAB code. To implement this equation, the wave speed as described in Equation 3.20 was used to determine the individual wave speed at the known depth of the region of interest. Following on from this, n needed to be determined as per Equation 3.27 to determine the group wave speed, c_g , as detailed in Equation 3.26. After determining all of these variables for the specific wave spectrum, the wave-number spectrum was determined using the relationship between the two-dimensional wave spectrum and two-dimensional wave-number spectrum detailed in Equation 3.17. To see the differences between the wave spectrum, and wave-number spectrum, a contour and surface plot are provided in Figure 4.11 of the equivalent wave spectrum shown in Figure 4.10.

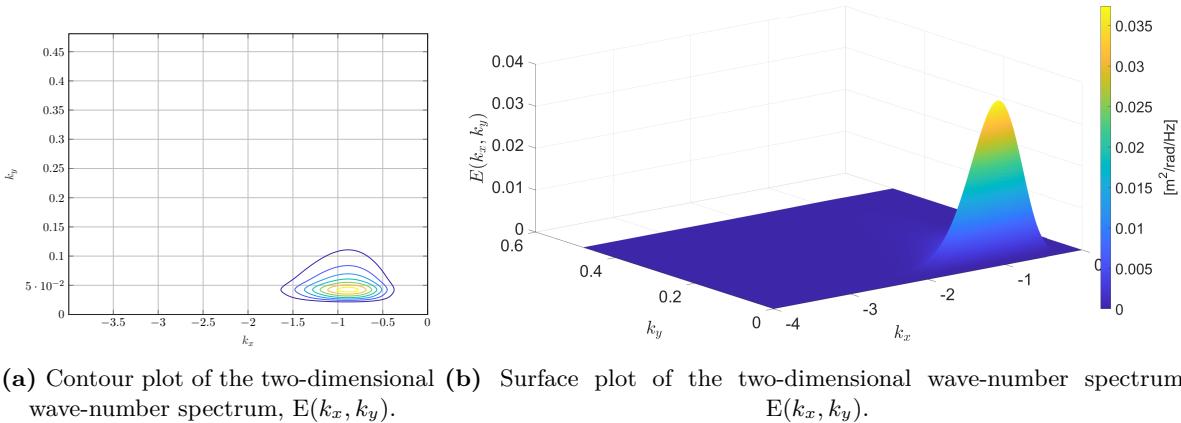


Figure 4.11: Two-dimensional wave-number spectra, $E(k_x, k_y)$, generated using NCEP wave data.

4.5 SAR Spectra

This report, as well as HH [59], make mention of a SAR spectrum. Whilst the generation of this spectrum was not shown in Figure 4.2, this spectrum is vital to the HH technique. This section briefly introduces how these plots were generated.

A SAR spectrum was generated by taking the two-dimensional FFT, `fft2`, of pre-processed intensity SAR data. Furthermore, the `fftshift` function was used, to move the zero-frequency components to the centre of the spectrum. The SAR spectrum was calculated and plotted in decibels and was generated using the `SARspectrumPlot` function.

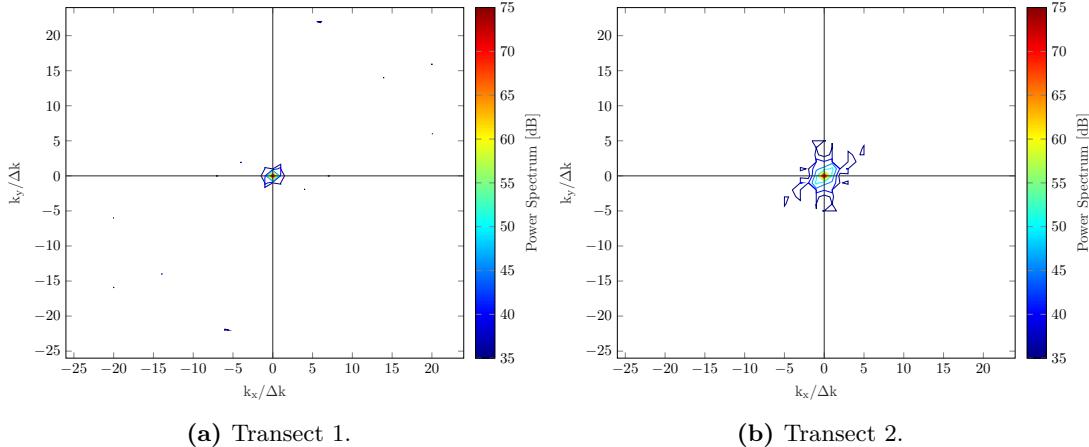


Figure 4.12: Observed SAR intensity spectrum plotted on a power spectrum scale from Oct. 20, 2023. Two transects are shown and the difference in spectra can be visually seen.

The axes of the plots in Figure 4.12 were determined from the wave spectrum. The spectral bandwidth, Δk , was determined using the wave spectrum's variance and was implemented in code as shown in Algorithm 2. This spectral bandwidth was used to scale the axes in Figure 4.12 to the conventional axes [47, 68] used when implementing the HH technique [59].

```

input      : Wave spectrum,  $E(\omega)$ 
            Frequency range,  $\omega$ 
output    : Spectral Bandwidth,  $\Delta k$ 
begin
    for  $i$  in 2 to  $length(\omega)$  do
        |  $d\omega(i) = \omega(i) - \omega(i - 1)$ 
    end
     $d\omega = mean(d\omega)$ 
     $E_{norm} = \frac{E(\omega)}{\int E(\omega) \cdot d\omega}$ 
     $E_{mean} = \int \omega \cdot E_{norm} \cdot d\omega$ 
     $E_{var} = \int (\omega - E_{mean})^2 \cdot E_{norm} \cdot d\omega$ 
     $\Delta k = \sqrt{E_{var}}$ 
end

```

Algorithm 2: Spectral bandwidth, Δk , calculation.

SAR spectra are used to compare the observed and generated spectra detailed in Section 4.5.

4.6 SAR Spectrum of Ocean Waves

The way in which an equivalent SAR spectrum of a wave-number spectrum was generated is described in Section 3.3.1. When designing this implementation in code, it was useful to break down the overall implementation into smaller, sub-blocks. Figure 4.13 shows all of the sub-blocks required to form a generated SAR spectrum and represents the `generateSARSpectrumOceanWaves` function. The SAR spectrum calculation block took in the wave-number spectrum generated using the wave spectrum generation block, the wave-number, k , along with its components, as well as three types of metadata: slant range, satellite velocity, and incidence angle. Each of the following subsections will introduce and discuss the individual sub-blocks of Figure 4.13, as well as break down these blocks further. Issues discovered when implementing are discussed, as well as any assumptions which were made.

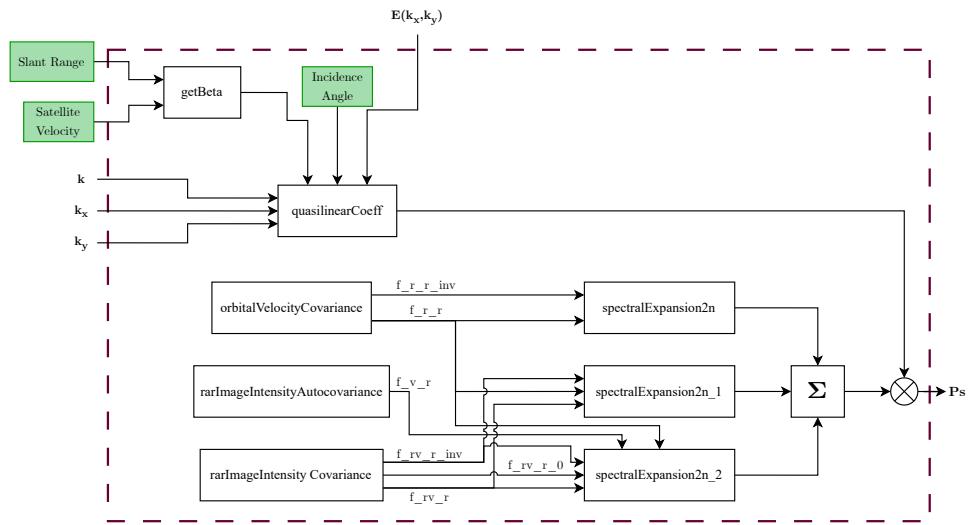


Figure 4.13: Block diagram depicting an expanded SAR spectrum generation sub-block of Figure 4.2.

4.6.1 Additional Metadata Extraction

The MTFs described in Section 3.3.1 each required some form of metadata from S1A SAR data to be calculated. These metadata values are tabulated in Table 4.1, along with additional functions that were developed to get the extracted metadata in the correct form for implementing the MTF equations.

MTF	Required Metadata Values	Additional Metadata Functions
Tilt MTF	Polarisation, Incidence Angle, Look	defineKLook, lookDiscretise
Hydrodynamic MTF	-	-
RAR MTF	Polarisation, Incidence Angle, Look	defineKLook, lookDiscretise
Range Velocity MTF	Incidence Angle, Look	defineKLook, lookDiscretise
Velocity Bunching MTF	Incidence Angle, Look Slant Range, Capture Time	defineKLook, lookDiscretise, getBeta, getSatVelocity
SAR Imaging MTF	Polarisation, Incidence Angle, Look, Slant Range, Capture Time	defineKLook, lookDiscretise, getBeta, getSatVelocity

Table 4.1: Metadata values required for each individual MTF used in the SAR spectrum generation of ocean waves. Additional functions created are named.

The metadata values shown in the **Required Metadata Values** column of Table 4.1 are simply extracted using the `filterAttributesNetCDF` function, with the appropriate required attributes chosen from Table 4.1. However, to implement the MTFs correctly, certain additional functions were required to get this metadata into the correct form or be used to calculate another value.

In the case of S1A's look, HH required that the component of the wave-number vector in the radar look direction was calculated. These coordinates were chosen such that the x -axis represented the SAR satellite flight direction, and the y -axis pointed in the positive or negative look direction, l , for a left or right looking SAR satellite respectively [59]. Therefore, the `lookDiscretise` function classified the extracted look value from the metadata as either 1 or 0. Where the look of S1A was left or right respectively. After this, the `defineKLook` function determined, based on the output of `lookDiscretise`, whether the y -axis pointed in the positive or negative look direction, and defined the look wave-number vector, k_l , accordingly.

The calculation of β proved troublesome. Remembering that β is the ratio of slant range, R , and satellite velocity, U . The main issue with this calculation was obtaining an accurate satellite velocity. S1A metadata provided orbit vector metadata in the form of the time, x-, y-, and z- position and velocity values. Calculating the satellite velocity, using the equation provided in Equation 4.1, for all orbit vectors seemed straightforward. However, it was found that calculating U for each individual orbit vector, resulted in an increase in U over time when the satellite's pass was ascending, and a decrease in U over time when the satellite's pass was descending.

$$U = \sqrt{U_x^2 + U_y^2 + U_z^2} \quad (4.1)$$

The difference between the maximum and minimum values of each individual orbit vector's velocity over Cape Point was found to be 4.6 m/s. Further investigation revealed that at different latitudes and longitudes, this difference was minimised and amplified. This led to the `getSatVelocity` function being designed to extract the closest orbit vector in time to the SAR data capture.

4.6.2 Modulation Transfer Functions (MTFs)

The implementation to calculate all MTFs was relatively straightforward and followed the respective equations described in Section 3.3.1. The only decision that needed to be made was to decide which algorithm were used to calculate the tilt MTF. As the extracted metadata value for polarisation using the `filterAttributesNetCDF` function, was a string of either *HH* or *VV*, a switch case was employed. Due to the fact that S1A is dual-polarised, the metadata values extracted were both co- and cross-polarised. To this effect, the polarisation value extracted was a 2x1 string array. Therefore, a loop was set up to loop through the array and determine if any of the polarisations matched the criteria set out by *HH*. The `switch` statement was then employed to apply the correct calculation for the tilt MTF as described in Equation 3.44. An error was thrown if neither *HH* nor *VV* polarisations were found. The implementation of the function used to calculate the tilt MTF in MATLAB, is detailed in Listing 4.8.

```

1 function Tt_k = tiltMTF( polarisation ,k_l, incidenceAngle )
2 status = 0;
3 for i = 1:length( polarisation )
4     switch polarisation(i)
5         case "HH"
6             Tt_k = 4.*1i.*k_l.*cotd( incidenceAngle ).*(1+sind(
7                 ↪ incidenceAngle ).^2).^( -1 );
8             status = 0;
9         case "VV"
10            Tt_k = 8.*1i.*k_l.* ( sind( 2.*incidenceAngle ) ).^( -1 );
11            status = 0;
12        otherwise
13            status = 1;
14        end
15    end
16    if( status )
17        error("SAR data are in neither VV or HH polarisations and Tilt
18            ↪ MTF cannot be computed");
19    end
20 end

```

Listing 4.8: MATLAB implementation of Equation 3.44 to determine tilt MTF.

4.6.3 Co and Autocovariance Functions

Orbital Velocity Covariance Function

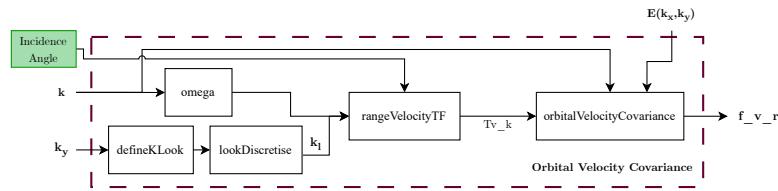


Figure 4.14: Block diagram depicting an expanded `orbitalVelocityCovariance` sub-block of Figure 4.13.

The `orbitalVelocityCovariance` function was implemented based on the equation described by HH [59] and detailed in Equation 3.49. The flow of this function is detailed in Figure 4.14. The way in which the wave number in the look direction was determined is described in Section 4.6.1, and this process is not repeated in the subsequent block diagrams as the procedure is identical. The `omega` function took in the wave number, k , and determined the equivalent radian frequency based on the relationship of $\omega = \sqrt{g \cdot k}$. After this, the range velocity MTF was calculated using Equation 3.46, which, in turn, allowed the orbital velocity covariance function to be calculated using Equation 3.49. An integral was performed using the built-in MATLAB function, `cumtrapz`, which is a trapezoidal approximation used to perform numerical integration, and the difference between each index of the wave number, k .

RAR Image Intensity Autocovariance Function

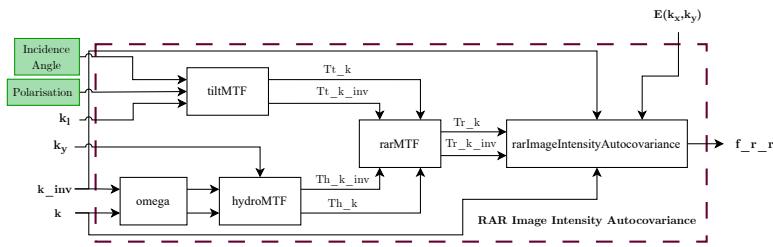


Figure 4.15: Block diagram depicting an expanded `rarImageIntensityAutocovariance` sub-block of Figure 4.13.

The `rarImageIntensityAutocovariance` function was designed based on Equation 3.50. The flow of this function is detailed in Figure 4.15. The wave number in the look direction was determined as detailed in Section 4.6.1, and is represented by k_l in Figure 4.15. The tilt, hydrodynamic and RAR MTFs were calculated using Equations 3.44, 3.45, and 3.43 respectively. The calculation of these MTFs allowed the RAR image intensity autocovariance function to be calculated using the `rarImageIntensityAutocovariance` function, which, like the orbital velocity covariance function, utilised the `cumtrapz` function to perform integration.

RAR Image Intensity Covariance Function

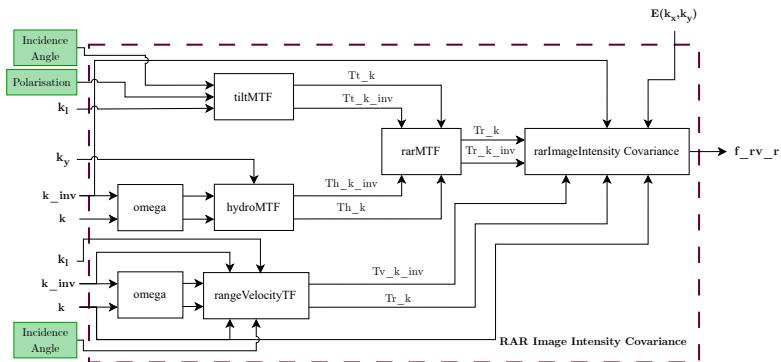


Figure 4.16: Block diagram depicting an expanded `rarImageIntensityCovariance` sub-block of Figure 4.13.

The `rarImageIntensityCovariance` function was designed based on Equation 3.51. The flow of this function is detailed in Figure 4.16. This function made use of functions described in Figure 4.14 and Figure 4.15, and as such the implementation was the same for these sub-blocks of Figure 4.16. The `rarImageIntensityCovariance` function, as with the previous two functions, performed integration using the `cumtrapz` MATLAB function.

4.6.4 Spectral Expansion

The three spectral expansion functions depicted in Figure 4.13 were simply implemented using Equations 3.52, 3.53, and 3.54 respectively. Whilst these equations appeared daunting and complicated, implementing these in code was relatively straight-forward, as all of the groundwork was done using the block described in Section 4.6.2 and Section 4.6.3. The only consideration for implementing these equations was in the determination of the non-linearity order of the ocean waves for which the `SAR` spectrum was generated.

As this project formed a preliminary investigation into the development of a pipeline for parameter extraction, the waves were assumed to have a non-linearity order of 1, which implied linear wave theory, as discussed in Section 3.1.4. Holthuijsen [4] states that ocean waves tend to become non-linear, and have a higher non-linearity order when entering shallow water, or when in their formation stage. Due to the fact that this project deals with deep water waves which are formed, this approximation was assumed to hold. This allowed Equation 3.55 to be implemented in MATLAB using a `for` loop as seen in Listing 4.9.

```

1 nonLinOrder = 1;
2 [ p_s_coeff , beta , xi_sqr , k_x ] = quasilinearCoeff(k,k_y,k_x,E_k,
   ↪ metadata , incidenceAngle );
3 for m = 2*nonLinOrder - 2 : 2 * nonLinOrder
4   coeff = ((k_x.*beta).^m);
5   if (m == 2*nonLinOrder)
6     P_s = P_s + coeff .* p_s_2n;
7   end
8   if (m == 2*nonLinOrder - 1)
9     P_s = P_s + coeff .* p_s_2n_1;
10  end
11  if (m == 2*nonLinOrder - 2)
12    P_s = P_s + coeff .* p_s_2n_2;
13 end
14
15 end
16 Ps = p_s_coeff .* P_s;
```

Listing 4.9: MATLAB implementation of Equation 3.55 to determine the full spectral expansion of the `SAR` spectrum of a wave spectrum.

The full spectral expansion was then plotted in MATLAB using the `SARSpectrumPlot` function and resulted in the power spectrum shown in Figure 4.17.

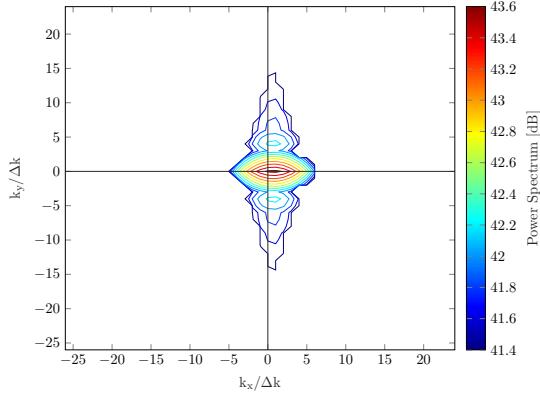


Figure 4.17: Generated SAR spectrum plotted using the `SARspectrumPlot` function. The shown SAR spectrum is generated using the wave spectrum depicted in Figure 4.11.

4.7 Inversion

The `inversion` sub-block of Figure 4.2 represented the core of the HH technique. The `inversion` block of the pipeline was designed with a modular approach and is shown in Figure 4.18. The main inputs of the `inversion` block were the first-guess wave-number spectrum, along with the observed SAR transect scene. On top of this, the block also took in the generated SAR spectrum of the wave spectrum, metadata, the number of iterations to minimise the cost function for, as well as the differences between indices in both k_x and k_y arrays. The `inversion` sub-block output all three types of wave spectra, from which wave parameters were extracted.

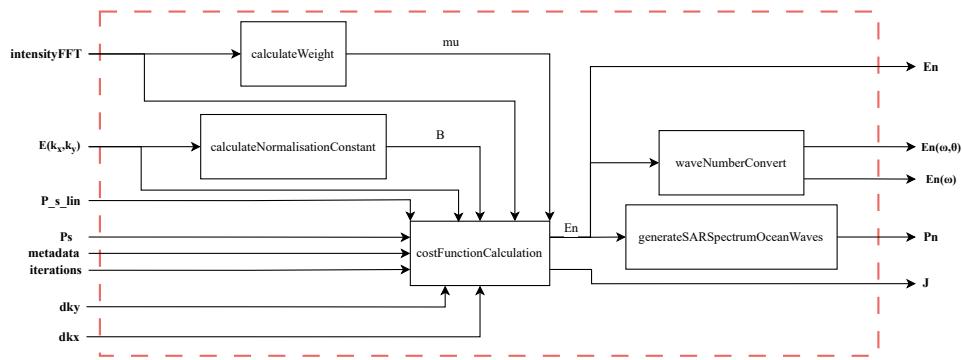


Figure 4.18: Block diagram depicting the expanded inversion sub-block of Figure 4.2.

4.7.1 Cost Function Calculation

The cost function used in the `costFunctionCalculation` block was designed using Equation 3.61. The function looped through the calculation an `iterations` number of times and each time, updated the respective variables. After each iteration, the block outputted the value of the cost function, J , as well as the best-fit wave spectrum, E_n . Using the best-fit wave spectrum determined by the cost function, the equivalent SAR spectrum was determined using the `generateSARSpectrumOceanWaves` function. These variables were used for analysis, and the best-fit wave spectrum was used to determine the best-fit wave parameters detailed in Section 4.7.2.

4.7.2 Best-fit Wave Parameters

Using the best-fit wave spectrum from the `costFunctionCalculation` function, which was in the form of a wave-number spectrum, different equations were applied to determine alternative forms of the wave-number spectrum. The `waveNumberConvert` function was used to achieve this. To calculate the equivalent frequency-direction best-fit wave spectrum, the inverse of the calculations done in Section 4.4.7 was applied. This resulted in an output two-dimensional wave spectrum, $E(\omega, \theta)$. This two-dimensional frequency-direction spectrum was integrated over all θ to give the one-dimensional wave spectrum, $E(\omega)$.

After determining the one-dimensional wave spectrum, this spectrum was used to determine the parameters of interest. The `SWP` was found by determining the index in $E(\omega)$ at which the maximum value occurred. This index value was referenced to the ω vector, to extract the frequency at which this peak value occurred, and the `SWP` was determined from this significant ω value using the relationship: $T_{1/3} = \frac{2\pi}{\omega_{\text{sig}}}$, where $\omega_{\text{sig}} = 0.95 \cdot \omega_{\text{peak}}$ [4].

The `SWH` was determined from $E(\omega)$ by taking the integral of $E(\omega)$ over all ω , known as the zeroth-moment, m_0 . The `SWH` is related to this zeroth-moment by the relationship $H_{1/3} = 4\sqrt{m_0}$ [4].

Chapter 5

Pipeline Verification Experiments

This chapter outlined the verification tests conducted during various stages of pipeline development. These tests played a crucial role in the pipeline's design process by identifying errors in both design and implementation. The primary objective of this chapter was to instil confidence in the accuracy of the best-fit wave parameters determined by the entire pipeline.

The structure of this chapter aligned with the sequential processing steps detailed in Chapter 4. This chapter utilised a single dataset of [S1A](#) and [NCEP](#) wave data. This date was chosen as October 2, 2023. Throughout this chapter, the [S1A](#) dataset, along with the corresponding wave data, served as the basis for verifying the implementation of each block in the pipeline.

5.1 [SAR](#) Data Transects

As mentioned in Section 4.2.3, the ability to take transects of full scene [SAR](#) data was vital. The difference between transects was measured by calculating Mean Squared Error ([MSE](#)) of all pixels in each transect using the [transectMSE](#) function.

```
1 transectMSEStruct = transectMSE(transectData);
```

Listing 5.1: MATLAB code used to calculate MSE values between all transects taken in Figure 4.4.

It was expected that identical transects would produce a [MSE](#) equal to 0, and the larger the [MSE](#), the more the transects differed. The transects were intensity data obtained by [S1A](#), therefore, it was expected that some pixels in different transects would be the same. The output structure is shown in Table 5.1.

Transects	MSE
Transect 1 vs. Transect 1	0
Transect 1 vs. Transect 2	136.3483
Transect 1 vs. Transect 3	384.8439
Transect 2 vs. Transect 1	136.3483
Transect 2 vs. Transect 2	0
Transect 2 vs. Transect 3	362.1531
Transect 3 vs. Transect 1	384.8439
Transect 3 vs. Transect 2	362.1531
Transect 3 vs. Transect 3	0

Table 5.1: [transectMSEStruct](#) values after running Listing 5.1.

The values in Table 5.1 confirmed that identical transects would return a MSE of 0 whilst different transects returned a high MSE value. This verified the uniqueness of the SAR data transects obtained using `get512Transects`.

5.2 Wave Spectra

Wave spectra were generated and used as a first-guess wave spectrum when minimising the cost function given in Equation 3.61. Thus, it was important to verify that wave spectra are correctly generated using NCEP wave data.

5.2.1 Wave Evolution

It was important to verify the generation of one-dimensional JONSWAP wave spectra. This verification included ensuring that these spectra exhibited the expected behaviour when entering shallower water. As depth decreases, the peak frequency of the wave was expected to decrease according to Equation 3.23. The height of the wave was expected to decrease due to the change in orbital motion of waves due to depth, as detailed in Chapter 5.4.2 of [4].

To see these changes, geographic coordinates surrounding the Cape Point were chosen. These coordinates were generated using the `createLatLonGrid` function and the resulting coordinates are shown in Listing 4.5. All of the individual points used to generate wave spectra are shown in Figure 5.1.

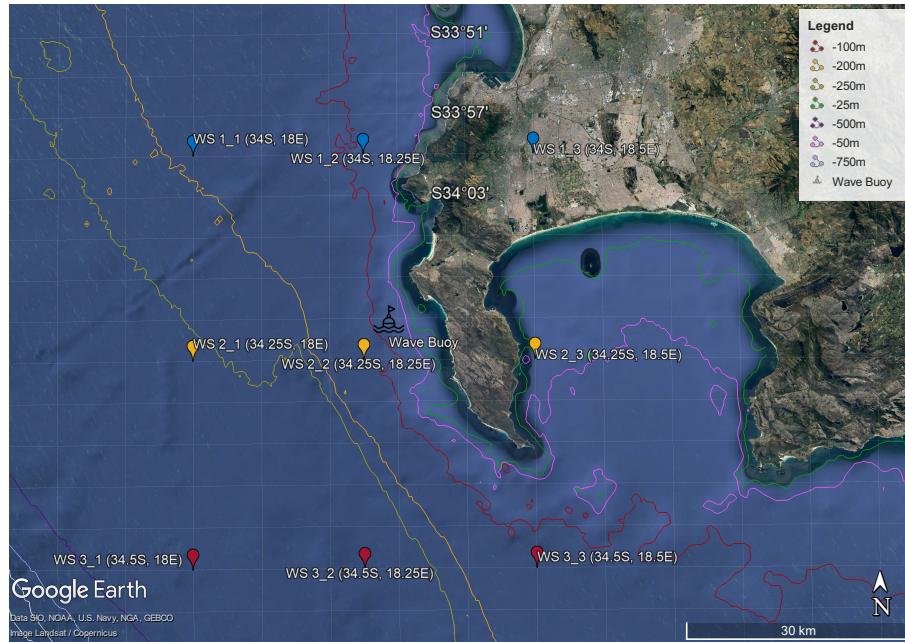


Figure 5.1: Geographic points of generated JONSWAP spectra using NCEP wave data. The points indicating each wave spectrum location correspond to the colours of the plots generated in Figure 5.2a. The contours shown in colour represent the respective depths below sea level: 25 m, 50 m, 100 m, 200 m, 250 m, 500 m, and 750 m. Bathymetric contour data sourced and adapted from OpenDem.

It is important to note that some points in Figure 5.1 did not aid the verification of generating

accurate JONSWAP spectra. The reasons why these were not suitable are discussed in Table 5.2.

Figure 5.1 Label	Location ($^{\circ}$)	Reason
WS 1_3 (34S, 18.5E)	34 $^{\circ}$ S, 18.5 $^{\circ}$ E	Data point located on land, as opposed to the ocean.
WS 2_3 (34.25S, 18.5E)	34.25 $^{\circ}$ S, 18.5 $^{\circ}$ E	Data point located in False Bay (East of Cape Point), as opposed to being West of Cape Point peninsula.
WS 3_3 (34.5S, 18.5E)	34.5 $^{\circ}$ S, 18.5 $^{\circ}$ E	No significant change in depth.

Table 5.2: Data points generated by the `createLatLonGrid` function not used to verify wave evolution over space.

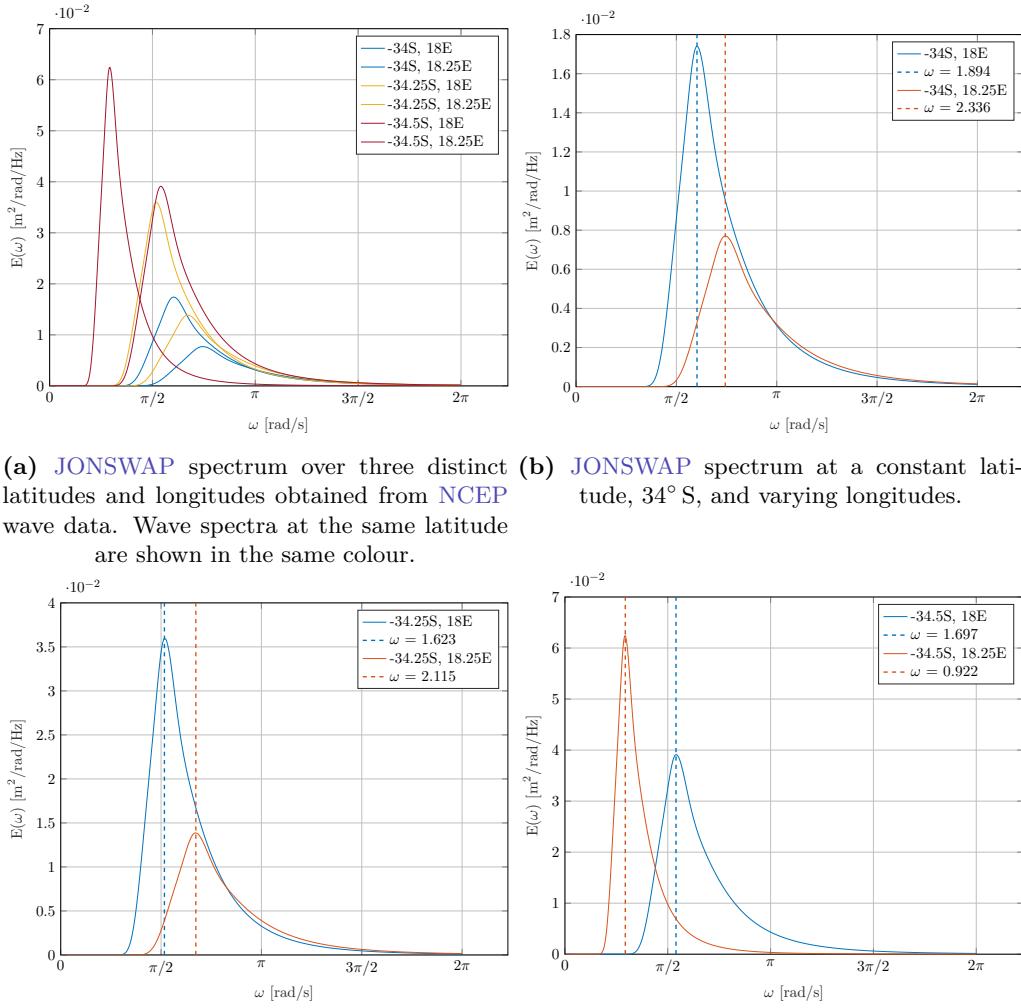


Figure 5.2: One-dimensional JONSWAP wave spectra, $E(\omega)$, which changed over different geographical locations surrounding Cape Point. Generated using NCEP wave data.

The wave spectra in Figure 5.2a varied for different geographical locations. Examining each individual plot in Figures 5.2b, 5.2c, and 5.2d allowed the effect of change in depth to be determined, and verified.

As the wave spectrum at 34° S moved from 18° E to 18.25° E in Figure 5.2b, the depth of the sea floor changed from approximately 200 m to 100 m as per the contour lines in Figure 5.1. As the depth changed, the location of the peak wave frequency, ω_{peak} , increased from 1.894 rad/s to 2.336 rad/s. This matched the relationship as defined in Equation 3.23 as a decrease in depth resulted in an increase in ω_{peak} . The spectrum height in 5.2b decreased from 1.74E-2 $\text{m}^2/\text{rad}/\text{Hz}$ to 0.77E-2 $\text{m}^2/\text{rad}/\text{Hz}$. This matched the change in the orbital motion of waves due to depth, as detailed in Chapter 5.4.2 of [4].

The same observations were made for the wave spectra at 34.25° S and 34.5° S seen in Figures 5.2c and 5.2d respectively as they both changed from 18° E to 18.25° E. The respective changes in depth from the contour lines in Figure 5.1, as previously explained, verified the output wave spectra generated in 5.2a as correct. Therefore, both the `generateSingleJONSWAP` and `generateMultipleJONSWAP` functions were implemented correctly.

5.2.2 Deep Water Approximation

Equation 3.26 shows the relationship between wave group and individual wave's phase velocities. Per [4], for deep water, this value should equal 0.5. To verify whether deep water assumptions held for the location of the CSIR directional wave buoy at a depth of 70 m, the value of n was expected to equal 0.5. This process was done using the `nValidation` plotting function, which took in the wave number, depth, and the step between depth values to plot.

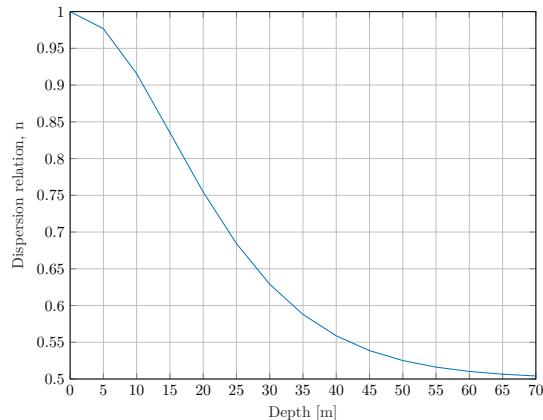


Figure 5.3: Plot generated using the `nValidation` function for varying depths using Equation 3.27.

Figure 5.3 showed that after the dispersion relation, n was within 10 % of 0.5 at a depth of 45 m and at 70 m depth, had a value of 0.5041. Therefore, the calculation of the n value was verified as correct and the deep water assumption held for calculations moving forward. Furthermore, this allowed the group wave velocities to be accurately calculated. Subsequently, the conversion between directional wave spectra and wave-number spectra using Equation 3.17 could be accurately performed.

5.2.3 Directional Distribution Function

The generation of the directional distribution function used in the calculation of the two-dimensional wave spectrum was verified through its definition. Holthuijsen [4] states that the

distribution is, "a normalised, circular transect through the two-dimensional spectrum". To meet the normalised criteria described by Holthuijsen, the integral over the directional distribution was taken. For the directional distribution to be correctly implemented, the area underneath the function was expected to equal 1.

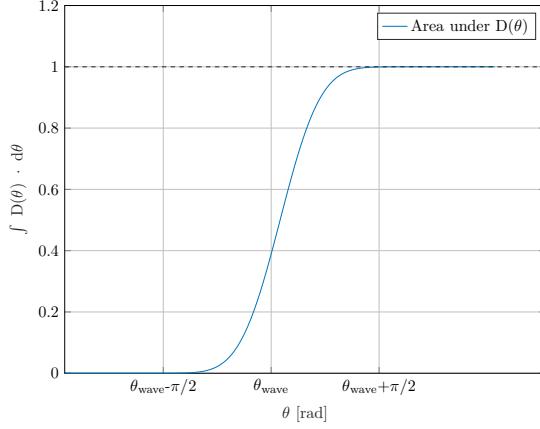
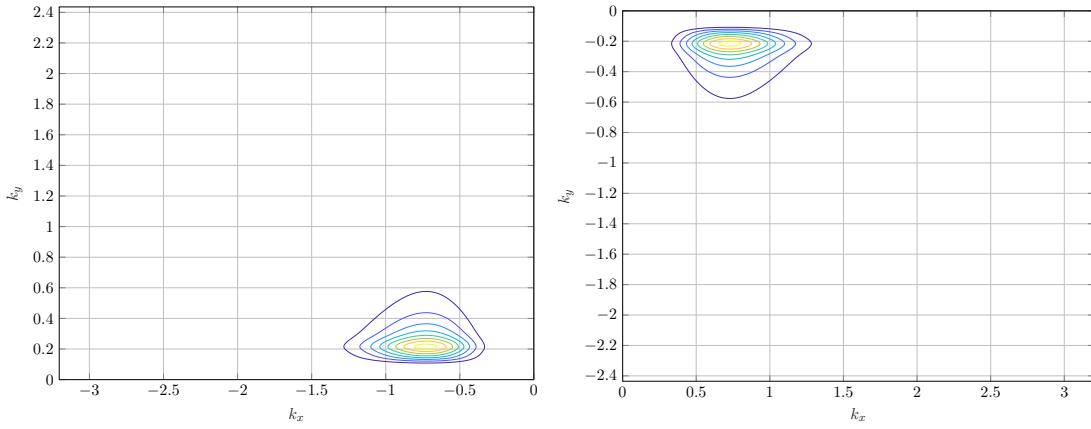


Figure 5.4: Plot of the area underneath the directional distribution function, $D(\theta)$, over all θ . An area of 1 is marked on the plot to verify that the expected condition is satisfied.

The value of $\int_0^{2\pi} D(\theta) \cdot d\theta$ was 1 and never went above this threshold in Figure 5.4 and therefore, the generation of the directional distribution function was verified as correct.

5.2.4 Negative k Wave-number Spectra

The negative wave-number spectrum, $E(-k)$, was required by Equations 3.50, 3.51 to compute the respective co- and autocovariance functions required for the SAR imaging of ocean waves. To verify that this spectrum was correctly generated, plots of both the original and negative wave-number spectra were compared. The $-k$ was calculated as $k_{\text{inv}} = -k$. The geometry used to determine the x and y components of k_{inv} is explained in Section ??.



(a) Two-dimensional wave-number JONSWAP spectrum, $E(k)$. (b) Two-dimensional negative wave-number JONSWAP spectrum, $E(-k)$.

Figure 5.5: Comparison of $E(k)$ and $E(-k)$, from Oct. 2, 2023, at 34,25° S, 18.25° E. Generated using NCEP wave data and the `waveNumberSpectrum` function.

Figure 5.5 shows a side-by-side comparison of both $E(\underline{k})$ and $E(-\underline{k})$. It was seen that determining the negative wave-number spectrum, $E(-\underline{k})$, resulted in a 'flipped' plot. This plot in Figure 5.5b was identical to the plot in Figure 5.5a, however, was defined over the negative k_x and k_y axes of those in Figure 5.5a. This represented a 180° rotation about the z -axis of Figure 5.5a and was the desired result. Therefore, the generation of a negative wave-number spectrum was verified as correct.

5.3 SAR Spectra of Ocean Waves

5.3.1 Power Spectral Expansions

The calculation of the power spectral expansions was vital to the HH procedure, as it formed the final step of generating an equivalent SAR spectrum from a wave-number spectrum. Equation 3.55 is repeated in this section as it is important to understand the context surrounding its formation.

$$P^S(\underline{k}) = \exp(-k_x^2 \xi'^2) \sum_{n=1}^{\infty} \sum_{m=2n-2}^{2n} (k_x \beta)^m P_{n,m}^S(\underline{k}) \quad (3.55)$$

The generation of each individual power spectrum was conducted using the respective equations detailed in Section 3.3.2.

When plotting these spectral expansions, a spectrum threshold of 120 was used throughout to clearly display the SAR spectrum. All of the plots produced in this section are also shown in Appendix B.1.1 without the applied threshold.

Plotting the three spectral expansions, $P_{n,2n}^S$, $P_{n,2n-1}^S$, $P_{n,2n-2}^S$, calculated using Equations 3.52, 3.53, and 3.54 respectively, resulted in the plots shown in Figure 5.6. These plots had normalised axes, as discussed in Section 5.3, and are zoomed in to clearly see the plotted spectra. The plot of $P_{n,2n-2}^S$ was not shown as it was zero for the entire zoomed-in range. Zoomed-out versions of Figure 5.6 are shown in Appendix B.1.1.

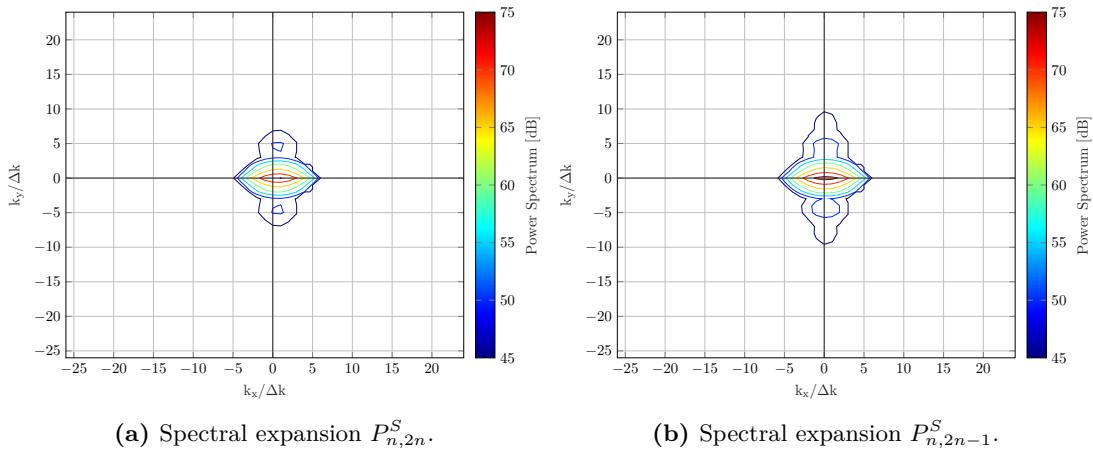


Figure 5.6: Spectral expansions of both $P_{n,2n}^S$ and $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from Oct. 2, 2023, at 34.25° S, 18.25° E. Generated using NCEP wave data, the `spectralExpansion2n`, `spectralExpansion2n_1`, and `SARspectrumPlot` functions.

Within each iteration of the summation described in Equation 3.55, each power spectrum was multiplied by a coefficient, $(k_x\beta)^m$. This coefficient is plotted in Figure 5.7 for $m = 1$ and Figure 5.8 for $m = 2$.

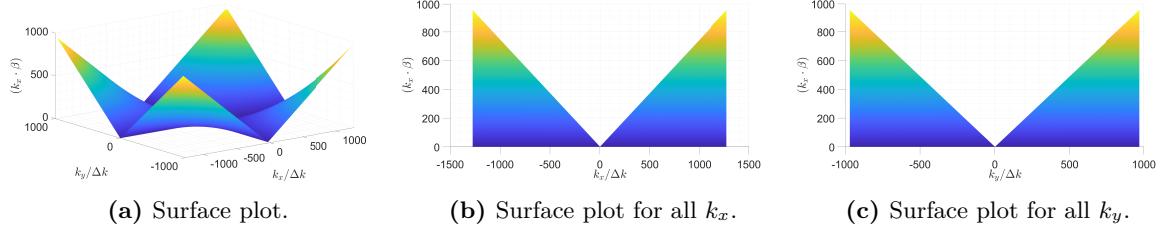


Figure 5.7: Plot of the coefficient, $(k_x\beta)$, used in Equation 3.55 which is multiplied by each power spectral expansion.

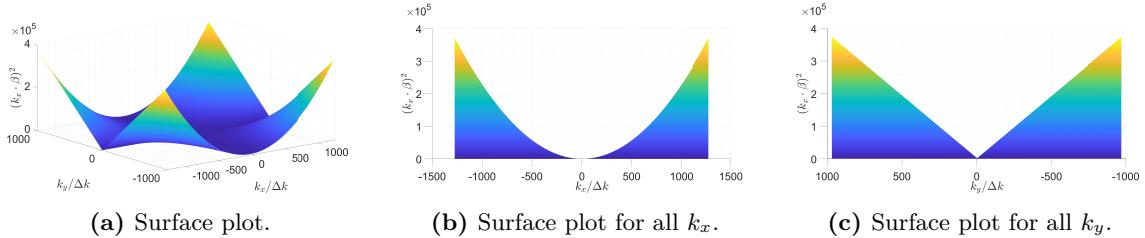


Figure 5.8: Plot of the coefficient, $(k_x\beta)^2$, used in Equation 3.55 which is multiplied by each power spectral expansion.

$k_x\beta$ represented a linear relationship for both k_x and k_y , and as such, a triangular shape was expected in the surface plots for all k_x and k_y in Figure 5.7b and Figure 5.7c respectively. The shape observed in Figure 5.7 verified the coefficient implementation, as the coefficient was of order 1 when $m = 1$.

Whereas, $(k_x\beta)^2$ represented a parabolic relationship for k_x and a linear relationship for k_y . This meant that a circular shape was expected for all k_x and a triangular shape was expected for all k_y . The plot of $(k_x\beta)^2$ for all k_x and k_y is shown in Figure 5.8 and the surface plots for k_x and k_y can be seen in Figure 5.7b and Figure 5.7c respectively. The expected shape of the coefficient surface plot was seen in Figure 5.8, as the coefficient was of order 2 when $m = 2$. This verified the implementation of the coefficient calculation.

Table 5.3 shows the relationship between n and m in Equation 3.55, and the implemented power spectral expansion for when a non-linearity order, n , of 1 was used.

m	Power Spectrum Used
0	$P_{n,2n-2}^S$
1	$P_{n,2n-1}^S$
2	$P_{n,2n}^S$

Table 5.3: m value for a non-linearity order of 1, and the associated implemented power spectrum used based on Equation 3.55.

Using Table 5.3, it was seen that when $m = 0$, $P_{n,2n-2}^S$ was multiplied by 1. Based on the fact that the spectrum was zero for all of the zoomed-in range, this multiplication was not plotted. The multiplication can be seen in Figure B.6a.

When $m = 1$, the coefficient seen in Figure 5.7 was multiplied with $P_{n,2n-1}^S$. Based on the plot in Figure 5.7 and the previous discussion about its form, it was expected that this would result in triangular, linear spreading about k_x and $k_y = 0$. This would take the form of linear ‘stretching’ about both axes.

When $m = 2$, the coefficient seen in Figure 5.8 was multiplied with $P_{n,2n}^S$. Based on the plot in Figure 5.7, and the prior discussion about its form, it was expected that this would result in circular spreading over k_x and triangular spreading over k_y .

The multiplication of both $P_{n,2n}^S$ and $P_{n,2n-1}^S$ in Figure 5.6 by their respective coefficients was shown in Figure 5.9. As previously stated, zoomed-out and zero threshold plots of the spectra in Figure 5.9 are shown in Appendix B.1.2.

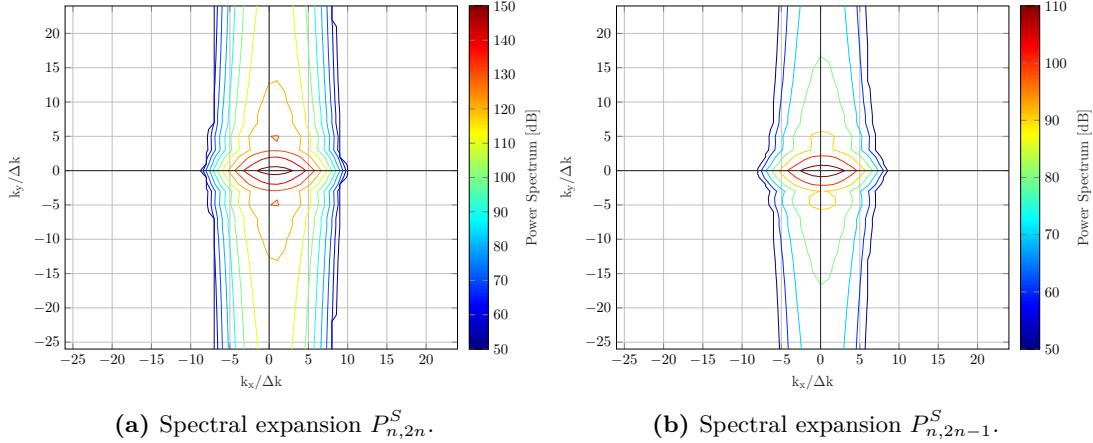


Figure 5.9: Spectral expansions of both $P_{n,2n}^S$ and $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a after being multiplied by the coefficient described in Equation 3.55.

Comparing Figure 5.9a and Figure 5.9b to their spectra prior to multiplication in Figure 5.6a and Figure 5.6b respectively, the difference in the coefficients discussed can be seen. The circular shaped components with a maximum at approximately $k_y = \pm 10$ in Figure 5.6b can be seen to straighten and follow a triangular shape in Figure 5.9b. This effect was amplified further away from k_y and $k_x = 0$. This verified the multiplication by $(k_x\beta)^2$.

In contrast, the spectral components at approximately $k_y = \pm 5$ in Figure 5.9a, were more rounded than those same components in Figure 5.6a. This verified the multiplication by $(k_x\beta)^2$. Furthermore, all components after multiplication had higher values, as indicated by the colour bar, in Figure 5.9a than in Figure 5.9b. This further verified the implementation of the coefficient multiplication based on the z values in Figure 5.8 and Figure 5.7 respectively.

5.4 Inversion

The cost function which was implemented as described in Equation 3.61. The change in the wave-number spectrum was calculated using Equation 3.62. To verify the correct implementation of the cost function, its behaviour was examined for 1 iteration, where the change in the output wave and SAR spectrum were expected to be equal to the first-guess wave and SAR spectra. After verifying the implementation of the cost function was correct, the behaviour was examined after multiple iterations to see whether the cost function updated the output

parameters correctly.

5.4.1 Single Iteration

The first-guess wave-number and **SAR** spectra were selected as the generated wave-number spectrum from **NCEP** wave data and the intensity **FFT** of **S1A GRD** data respectively.

The cost function described in Equation 3.61, was initially set using these first estimates. The output parameters, **deltaEk** and **deltaPk** from the **costFunctionCalculation** function were compared with the first-guess wave and **SAR** spectrum using the MATLAB **isequal** function. Both of these checks returned a value of 1, and therefore, the implementation of Equation 3.61 was verified as correct based on the behaviour for 1 iteration.

5.4.2 Multiple Iterations

The number of iterations was increased and verified against the sample data from October 2, 2023. For each iteration, the **SWH** and period were calculated from the output one-dimensional wave spectra using the relationships defined in Section 4.7.2. For each iteration, the best-fit one-dimensional wave spectrum was plotted on the same set of axes as the first-guess wave spectrum. Furthermore, the values of the **SWH** and **SWP** were recorded for each iteration and stored in an array, which was plotted over the number of iterations to see the Reliative Error (**RE**) of the best-fit values compared to the first-guess values after each minimisation.

Only three distinct iterations were plotted in Figure 5.10 and each highlighted a different, notable feature. Verification of the extracted parameters was conducted using Figure 5.11, which showed the output parameters after each iteration.

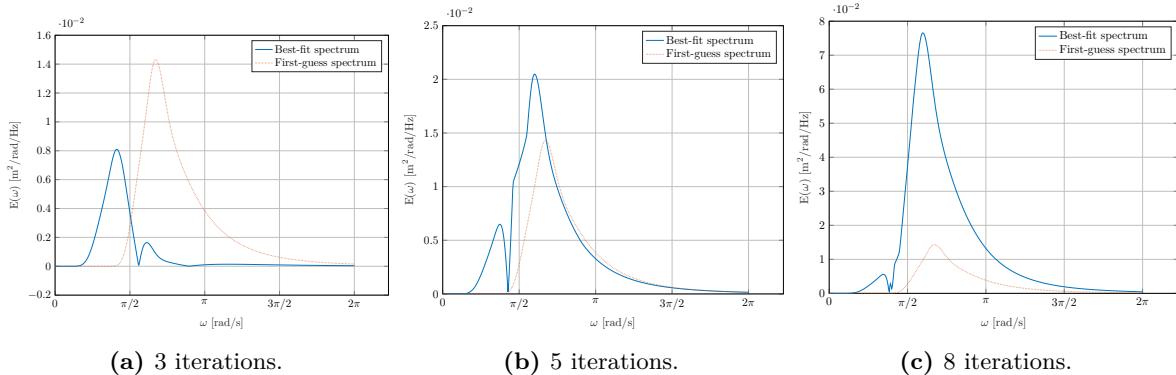
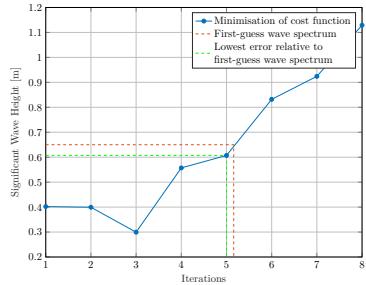


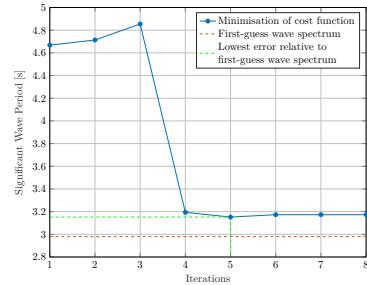
Figure 5.10: Comparison of best-fit and first-guess one-dimensional wave spectra obtained using the **HH** minimisation technique for different numbers of iterations.

The verification of the calculation of **SWH** was done using 3 and 8 iterations, shown in Figure 5.10a, and 5.10c respectively. The spectra were compared to the calculated values for the respective iterations shown in Figure 5.11a. The calculated **SWH** of iteration 3 was lowest, whereas for iteration 8, it was at its maximum. This can be seen in Figure 5.10a, where the smallest spectral peak was achieved, and in Figure 5.10c, where the highest spectral peak was achieved. This verified the calculation of the **SWH** parameter from the best-fit one-dimensional wave spectrum.

The **SWH** parameter was verified using iterations 3 and 5, shown in Figure 5.10a, and 5.10b respectively. The way in which a larger and smaller **SWP** value impacted the shape of the spectra was used to verify the calculation. Remembering that period and frequency have an inverse relationship, it was expected that the largest **SWP** would correspond to a wave spectra with a spectral peak at a lower ω value. This was seen in Figure 5.10a, which had the lowest ω_{peak} value and corresponded to the highest **SWP** value in Figure 5.11b. The opposite effect was seen in Figure 5.10b and verified the calculation of the **SWP** from the best-fit one-dimensional wave spectrum.



(a) Significant Wave Height.



(b) Significant Wave Period.

Figure 5.11: Comparison of best-fit wave parameters over each iteration of the **HH** minimisation technique compared to first-guess values obtained from **NCEP** wave data.

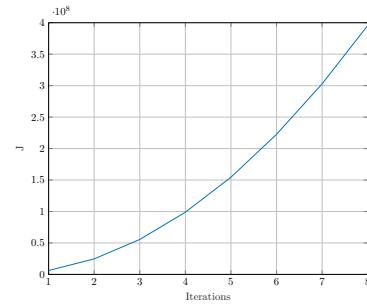


Figure 5.12: Cost function minimisation over 8 iterations.

Verification of the cost function minimisation was done by plotting the cost function output parameter, J , over each iteration. This plot is seen in Figure 5.12.

In the literature, the **HH** inversion technique cost function minimised in 3-4 iterations [47, 59, 60, 68]. However when implemented, it did not minimise, as seen in Figure 5.12, and in Figure 5.11a, where after 5 iterations the height increases with each subsequent iteration. Therefore the implemented cost function could not be validated. This undesirable result was attributed to the best-fit wave spectrum generated by the `costFunctionCalculation` function.

The cost function value impacted the calculation of wave parameters. Consequently, the cost function's incorrect maximisation affected the accuracy of the best-fit wave parameters. In Figure 5.10a, the best-fit wave spectra were smaller than the first-guess wave spectrum. However, when iterations were greater than 5, Figure 5.10b and Figure 5.10c showed that the best-fit wave spectra were taller than the first-guess wave spectrum. All best-fit wave spectra after 5 iterations increase in height. Again, compared to the literature [59], this was an unexpected result and was attributed to the incorrect cost function implementation.

The **SWH** was calculated using the area under the spectrum, as described in Section 4.7. The unexpected increase in height of the best-fit wave spectrum also resulted in an increase in the total area underneath the curve. Subsequently, the calculated **SWH** also unexpectedly increased as seen in Figure 5.11a.

Due to the unexpected behaviour of the cost function, an additional step was added to find the number of iterations that produced the most accurate **SWH**. To find this number, the **RE** between the best-fit and first-guess wave parameters for each iteration was calculated as

$$\text{RE} [\%] = \frac{\text{Best-Fit Wave Parameter} - \text{First-Guess Wave Parameter}}{\text{First-Guess Wave Parameter}} \quad (5.1)$$

where *First-Guess Wave Parameter* was either of the first-guess parameters, $\hat{H}_{1/3}$ or $\hat{T}_{1/3}$, and *Best-Fit Wave Parameter* was the respective best-fit parameter, $H_{1/3}$ or $T_{1/3}$. A negative value represented a best-fit parameter less than the first-guess parameter. These values are record in Table 5.4.

Iterations	$\hat{H}_{1/3}$ [m]	$H_{1/3}$ [m]	$RE_{H_{1/3}}$ [%]
1	0.65	0.402	-38.15
2	0.65	0.399	-38.62
3	0.65	0.299	-54.00
4	0.65	0.557	-14.31
5	0.65	0.607	-6.62
6	0.65	0.832	28.00
7	0.65	0.924	42.15
8	0.65	1.129	73.69

Table 5.4: RE after each iteration of the cost function.

The RE values in Table 5.4 were compared, and it was found that 5 iterations produced the smallest RE. Therefore, it was expected that visually, the SWH value at 5 iterations would be closest to the first-guess value for SWH.

Subsequently, the error for 5 iterations was plotted in Figure 5.11a. The first-guess line in Figure 5.11a shows the expected value, however, it takes 5.2 iterations to achieve this value. Since iterations were discrete whole numbers, the closest iteration value was 5, and this was confirmed by the calculation of the RE in Table 5.4. Therefore, this further validated choosing 5 iterations to achieve results with minimal error.

Figure 5.11b shows that the period minimised at 4 iterations and remained relatively constant for subsequent iterations. Performing the previous analysis on SWP for all iterations, it was also found that the least RE occurred at 5 iterations. This is shown in Figure 5.11b.

Therefore, it was determined that 5 iterations produced the least RE in both SWH and SWP. This value was therefore used for case study testing in Chapter 6 of this report.

5.5 Summary

- The following steps of the pipeline were verified as correct: Taking a unique SAR data transect of the full scene; generation of a one-dimensional JONSWAP wave spectrum, the deep water approximation; generation of a directional distribution function, and as a result the generation of the two-dimensional wave spectra; generation of a negative wave-number spectrum; generation of an equivalent HH SAR spectrum of ocean waves; calculation of power spectral expansions, and their associated coefficients; and the implementation of the HH inversion technique for a single iteration.
- The implementation of the HH cost function was incorrect for multiple iterations and increased for each subsequent iteration. The SWP was found to minimise, whereas, the Significant Wave Height (SWH) had the lowest RE for 5 iterations.
- It was decided that **5 iterations** would be used for case study testing in Chapter 6.

Chapter 6

Pipeline Case Study Testing

This chapter detailed the testing procedure used to test the entire implementation of the pipeline. Due to the fact that ground-truth data from the CSIR directional wave buoy could not be obtained in time to test the full pipeline, multiple locations were analysed in order to test the pipeline's implementation. Thus, this chapter will detail two various testing setups for different locations, as well as discuss the obtained results. The two setups investigated the performance of the pipeline for different conditions, namely depth and sea state.

The results discussion involved analysing the first-guess wave and SAR spectra for each location and comparing these to the best-fit spectra obtained after minimising the cost function described in Equation 3.61 for 5 iterations. The results from each location were compared to establish the efficacy of the pipeline and its performance under different conditions.

6.1 Testing Setup

Due to the aforementioned lack of ground-truth data, the pipeline had to be tested for multiple locations with differing features. This was done through investigating the pipeline's implementation at locations surrounding Cape Town, at different depths and on different dates. Each date represented different sea state conditions. This was done to verify the implementation at multiple ocean depths, which would impact the wave spectra at these locations. This allowed the robustness of the pipeline to be tested for multiple depths and locations.

The chosen locations are shown in Table 6.1, along with the associated geographical location and depth. These locations were plotted using Google Earth in Figure 6.1 with bathymetric contour data shown.

Location Tag	Location	Depth [m]
CS_1	34.25°S, 18.25°E	130
CS_2	34°S, 18.25°E	80
CS_3	34.25°S, 18.75°E	50
CS_4	34.5°S, 18.75°E	125

Table 6.1: Case study testing locations and associated depths.



Figure 6.1: Geographic points of the four case study testing locations. The contours shown in colour represent the respective depths below sea level: 25 m, 50 m, 100 m, 200 m, 250 m, 500 m, and 750 m. Bathymetric contour data sourced and adapted from OpenDem.

After deciding the four locations, the dates of the investigation needed to be decided. These dates were limited by the dates for which [S1A](#) data was available. [S1A](#) data over the False Bay and Cape Point region was available at intervals of roughly 5 days. [NCEP](#) wave data was only available for 10 days, backdated from the current date. Due to this fact, only two dates could be investigated. Oct. 15, 2023, and Oct. 20, 2023, were chosen as the dates of investigation for the case study testing.

Case study testing was conducted by generating the one-dimensional [JONSWAP](#) wave spectra for each location, and from this, the associated two-dimensional wave-number spectrum was generated. This two-dimensional wave spectrum was used as the first-guess wave spectrum. Additionally, these wave spectra were used to generate the associated first-guess [SAR](#) spectrum of these ocean waves.

6.2 Depth Investigation

Due to the lack of ground-truth data, relationships between [SWH](#) and [SWP](#) at varying depths were investigated. As discussed in Section 3.1, depth impacts wave parameters through the dispersion relationship 3.19. As such, waves in shallower water were expected to have greater [SWH](#) values, and smaller [SWP](#) values. The trends observed for the [RE](#) between the first-guess and best-fit wave parameters were analysed for the [SWP](#) to establish the accuracy of the pipeline, relative to the first-guess wave spectrum.

The date for this investigation was chosen as October 20, 2023 due to improved coverage east of the Cape Point peninsula, and south of False Bay. This allowed deeper water locations to be investigated and the subsequent depth effects analysed.

The full scene [SAR](#) data obtained from [S1A](#) for the investigation is shown in Figure 6.2 with the transects used, highlighted and labelled.

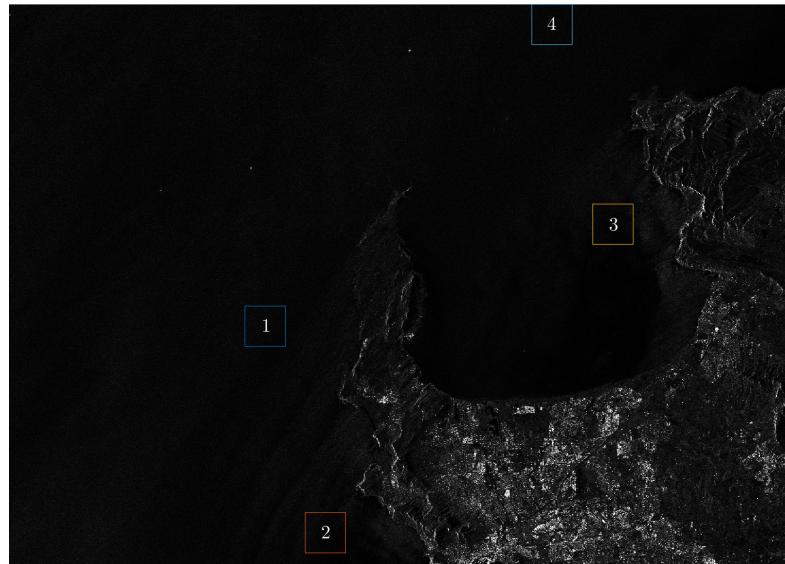


Figure 6.2: S1A SAR image over Cape Town acquired on October 20, 2023, at 18:56 UTC. Transects used in case study testing are highlighted and labelled.

6.2.1 Results

The four transects shown in Figure 6.2 were passed through the pipeline along with their respected one-dimensional JONSWAP modelled wave spectra. These data are shown in Table 6.2 where the $\hat{\cdot}$ notation represents first-guess values.

Location	Depth	$H_{1/3} \hat{[m]}$	$T_{1/3} \hat{[s]}$
34.25°S, 18.25°E	130 m	1.34	5.17
34°S, 18.25°E	80 m	1.48	5.32
34.25°S, 18.75°E	50 m	1.59	5.41
34.5°S, 18.75°E	125 m	1.28	4.74

Table 6.2: First-guess wave parameters for locations of interest for October 20, 2023. Sourced from NCEP wave data.

The cost function was minimised for 5 iterations, due to the reasons discussed in Section 5.4.2, and produced the one-dimensional wave spectra shown in Figure 6.3 for each location. Both the first-guess JONSWAP and output best-fit wave spectra were plotted on the same set of axes for visual comparison. Plots over the full range ω axis can be found in Appendix C.31.

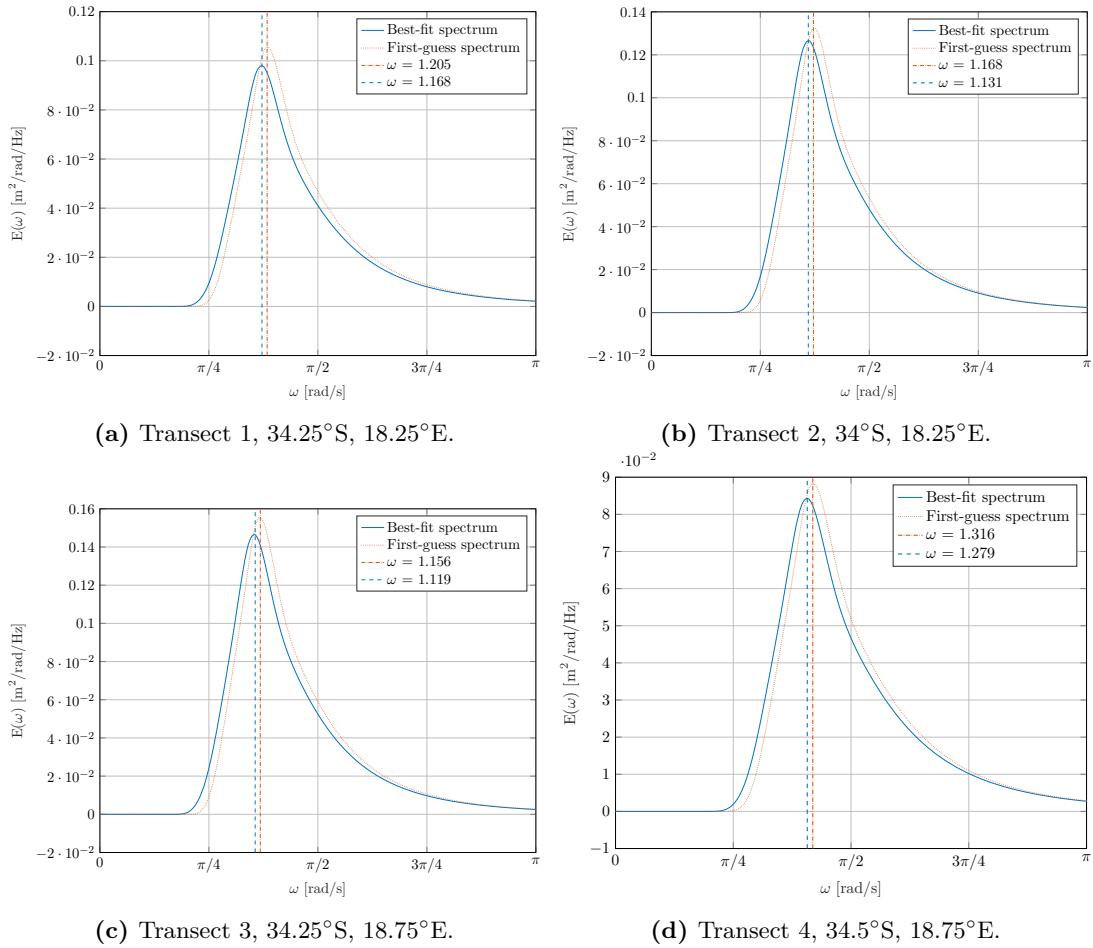


Figure 6.3: Best-fit and first-guess wave spectra after 5 iterations for all transects.

The best-fit parameters for $H_{1/3}$ and $T_{1/3}$ were extracted by the methods described in Section 4.7.2, and are shown in Table 6.3.

Transect	Location	Depth	$\hat{H}_{1/3}$ [m]	$\hat{T}_{1/3}$ [s]	$H_{1/3}$ [m]	$T_{1/3}$ [s]
1	34.25°S, 18.25°E	130 m	1.34	5.17	1.0522	5.1100
2	34°S, 18.25°E	80 m	1.48	5.32	1.1786	5.2766
3	34.25°S, 18.75°E	50 m	1.59	5.41	1.2590	5.3346
4	34.5°S, 18.75°E	125 m	1.28	4.74	1.0155	4.6678

Table 6.3: Best-fit wave parameters for four defined transect locations for October 20, 2023. Generated using the pipeline mlx after 5 iterations.

6.2.2 Discussion

As discussed in Section 5.4.2, the Best-fit Significant Wave Height (**BSWH**) parameters were not reliable. However, the trends in terms of these best-fit parameters are discussed. From Table 6.3, the largest First-guess Significant Wave Height (**FSWH**) occurred in the shallowest water in transect 3. This relationship was maintained for the **BSWH**. The deepest water, in transect 1 and 4, had the smallest **FSWH** values.

The RE of these data in Table 6.3, when compared to the first-guess values, are shown in Table

C.3 and confirm that whilst the **BSWH** parameters are inaccurate, they are still precise for different depths.

The **SWP** was minimised, as discussed in Section 5.4.2. This was used to determine the accuracy of the **SWP** at varying depths relative to the first-guess wave spectrum. Considering the dispersion relationship, it was expected that shallower water would produce larger **SWP** values. This relationship was seen for the first-guess parameters in Table 6.2. Furthermore, this relationship was observed for the best-fit wave parameters, as transect 3 estimated the largest **SWP**, in the shallowest water of the investigation.

The deepest water, in transect 1, did not have the smallest **SWP** for either the first-guess or best-fit parameters. Whilst transect 1 was in marginally deeper water than transect 4, this difference in wave period was attributed to the rate of change of the continental shelf, seen in Figure 6.1 through the proximity of the 200 m and 100 m contour lines. The rate of change for transect 1 was greater than that of transect 4 as the 200 m contour was closer to the 100 m contour than in the case of transect 4. This phenomenon is described in Chapter 5.4 of [4].

Therefore, it was determined that the relationship between depth and **SWP** held for the Best-fit Significant Wave Period (**BSWP**) parameters. Further investigation of the **SWP** parameters was conducted using the relative error between the best-fit and first-guess wave parameters.

Table C.3 shows the raw data obtained using Equation 5.1. The maximum **RE** for **SWP** occurred in transect 4, equal to **-1.52 %**. The minimum **RE** occurred for transect 2 of **-0.82 %**. This gave a spread of the **RE** as **0.70 %** for varying depths. Transect 2 and 3 were classified as shallow water, and transects 1 and 4 as deep water. This gave the average **RE** for deep and shallow water as **-1.34 %** and **-1.11 %** respectively. Whilst the **RE** for deep water, was greater than shallow water, the difference was only **0.23 %**. Therefore, it was determined that the **BSWP** was accurately determined by the pipeline.

The analysis of the **SWH** and **SWP** at varying depths indicated that the pipeline was precise and accurate for these test cases. This was determined despite the inaccuracies described in Section 5.4.2 and the absence of ground-truth data. It was therefore proven that ocean depth did not impact the pipeline's performance.

6.3 Sea State Investigation

Due to the lack of ground-truth data, relationships between **SWH** and **SWP** for varying dates, and as a result, sea states were investigated. The trends observed for the **RE** between the first-guess and best-fit wave parameters were analysed for the **SWP** to establish the accuracy of the pipeline, relative to the first-guess wave spectrum.

The dates for this investigation were chosen as October 20, 2023 and October 15, 2023. As previously discussed, October 20, 2023 had improved coverage when compared to October 15, 2023. This meant that transect 1 in Section 6.2 did not appear in the **S1A** data for October 15, 2023. As such, only transects 2 through 4 were used for the analysis of the pipeline's performance for varying sea states.

The two full scene **SAR** data obtained from **S1A** for this investigation is shown in Figure 6.4 with the transects used, highlighted and labelled.

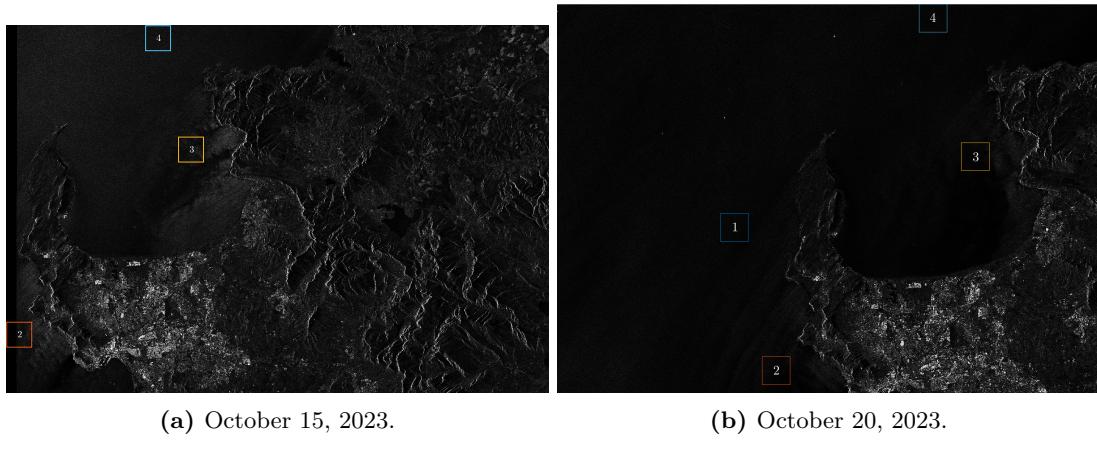


Figure 6.4: SIA SAR image over Cape Town. Acquired on October 15, 2023 at 17:26 UTC, and October 20, 2023, at 18:56 UTC. Transects used in case study testing are highlighted and labelled.

The three transects displayed in Figure 6.2 were processed by the pipeline, along with their respective one-dimensional JONSWAP modelled wave spectra, generated using NCEP data. These data are shown in Table 6.4.

Location	Date	$H_{1/3}$ [m]	$T_{1/3}$ [s]
34°S, 18.25°E	Oct. 15, 2023	1.03	8.77
	Oct. 20, 2023	1.48	5.32
34.25°S, 18.75°E	Oct. 15, 2023	0.46	8.95
	Oct. 20, 2023	1.59	5.41
34.5°S, 18.75°E	Oct. 15, 2023	0.85	9.01
	Oct. 20, 2023	1.28	4.74

Table 6.4: First-guess wave parameters for locations of interest for October 15, 2023 and October 20, 2023. Sourced from [NCEP](#) wave data.

The [HH](#) cost function was minimised for 5 iterations and produced the one-dimensional wave spectra shown in Figure 6.5 for each location. Both dates' spectra are plotted for a visual comparison of the varying sea state on the two distinct dates. Plots of the full range ω axis can be found in Figure C.15.

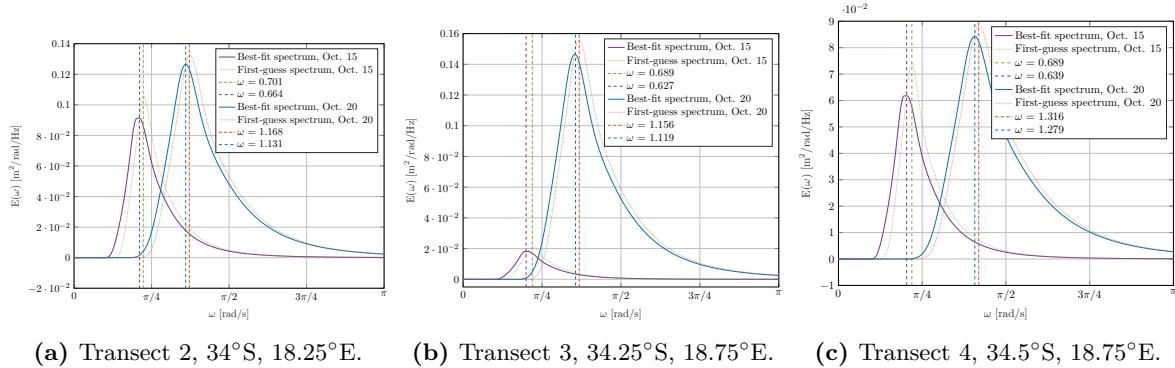


Figure 6.5: Best-fit and first-guess wave spectra after 5 iterations for all transects.

The best-fit parameters for SWH and SWP, were extracted using the methods described in

Section ?? and are shown in Table 6.5.

Transect	Location	Date	$\hat{H}_{1/3}$ [m]	$\hat{T}_{1/3}$ [s]	$H_{1/3}$ [m]	$T_{1/3}$ [s]
2	34°S, 18.25°E	Oct. 15, 2023	1.03	8.77	0.8139	8.9898
		Oct. 20, 2023	1.48	5.32	1.1786	5.2766
3	34.25°S, 18.75°E	Oct. 15, 2023	0.46	8.95	0.3603	9.5186
		Oct. 20, 2023	1.59	5.41	1.2590	5.3346
4	34.5°S, 18.75°E	Oct. 15, 2023	0.85	9.01	0.6641	9.3356
		Oct. 20, 2023	1.28	4.74	1.0155	4.6678

Table 6.5: Best-fit wave parameters for three defined transect locations for October 15, 2023 and October 20, 2023. Generated using the pipeline mlx after 5 iterations.

6.3.1 Discussion

Figure 6.5 shows that for all locations, the sea state varied significantly. The mean of all parameters on each date was determined and is seen in Table C.1. $H_{1/3(15)}$ and $H_{1/3(20)}$ were determined to be 0.78 and 1.45 respectively and $T_{1/3(15)}$ and $T_{1/3(20)}$ were determined to be 7.71 and 5.16 respectively. From these data, as well as a visual comparison of these data in Figure 6.5, it was determined that the sea states were substantially different to one another, and therefore, analysis on differing sea state could be conducted.

Again, due to the issues discussed in Section 5.4.2, the BSWH parameters were not reliable. From Table 6.5, it can be seen that all FSWHs on Oct. 15, 2023, were lower than the FSWH on Oct. 20, 2023. This trend between the two dates was maintained for the respective BSWH.

Furthermore, comparing the RE of the SWH for these two dates relative to the first-guess wave parameters, shown in Table C.2, revealed that whilst the BSWH was inaccurate, it remained precise for differing sea states.

Analyses of the SWP was conducted, as this parameter was minimised. The relationship between depth and SWP was expected to be maintained regardless of the sea state as well as maintaining a consistent, small RE with respect to the first-guess wave parameters.

As discussed in Section 6.2.2, the relationship between depth and SWP was maintained for the BSWP for Oct. 20, 2023. Similar analysis with respect to the deepest and shallowest water for Oct. 15, 2023 was conducted, and resulted in the same relationship being determined to hold.

Therefore, it was determined that the relationship between depth and SWP held for multiple sea states. Further investigation of the SWP parameters was conducted using the RE between the best-fit and first-guess wave parameters.

Table C.2, shows the raw data obtained using Equation 5.1. For Oct. 20, 2023, the minimum RE occurred in transect 2, and for Oct. 15, 2023, the minimum RE occurred in transect 2 as well. Furthermore, for Oct. 20, 2023, the maximum RE occurred in transect 4, and for Oct. 15, 2023, the maximum RE occurred for transect 3. The spread of the RE for Oct. 20, 2023, was **-0.82 %**, whereas, for Oct. 15, 2023 it was **3.84 %**. The average RE for Oct. 20, 2023 was **-1.22 %**, whereas the average RE for Oct. 15, 2023 was **4.16 %**.

Interestingly, the RE for Oct. 20, 2023 was a negative value, which meant that the best-fit wave parameters were always less than the first-guess wave parameters. In contrast, the RE for Oct. 15, 2023 was positive. This meant that the best-fit wave parameters were always greater than

the first-guess wave parameters.

Therefore, it was determined that for large **SWPs**, the **BSPW** overestimated the value with respect to the First-guess Significant Wave Period (**FSWP**). Conversely, for smaller **SWPs**, the **BSPW** was marginally underestimated with respect to the **FSWP**.

Whilst this was an interesting result of the **BSPW**, the analysis of the **SWH** and **SWP** for different sea states showed that the pipeline was precise and accurate for these test cases. This was proven despite the inaccuracies described in Section 5.4.2 and the absence of ground-truth data. Therefore, it was proven that whilst sea state did slightly impact the pipeline's performance, the best-fit parameters determined were within an acceptable range, less than 5%.

Further analysis into the value at which the **BSPW** had minimal **RE** would have been useful, however, owing to the limitation with respect to the overlap in time of both wave and **S1A** data, this was not possible.

Chapter 7

Conclusions

The main objective of this project was to design and implement a pipeline to extract wave parameters from **SAR** data. The lack of year-round, in-situ data in the **MIZ** necessitated the use of remote sensing. This project was used to showcase a foundational implementation which informs the broader research goal; to develop an Antarctic sea ice parameter extraction pipeline. Thereby, improving accessibility of data for the **MIZ**.

This report began by investigating the relevant literature and theoretical background concerning ocean waves and **SAR** in Chapters 2 and 3 respectively. This investigation aimed to frame the context of the problem described in Section 1.2. The literature clearly demonstrated that accessing sea ice parameters in Antarctica is vital to improving climate, and wave attenuation and dispersion models for the region. Further investigation into relevant literature revealed the potential of remote sensing techniques for extracting sea ice and wave parameters. While the literature analysis revealed successfully implemented wave parameter extraction algorithms, as well as wave dispersion and attenuation models, it was found that no complete pipeline had been developed for the Antarctic region.

Following the extensive literature and theoretical analysis, Chapter 4 introduced the core of this project: the modular design of a wave parameter extraction pipeline in MATLAB. The five main blocks of the pipeline were identified as: pre-processing, metadata extraction, wave spectrum generation, **SAR** spectrum calculation, and inversion. Each block was designed using multiple sub-blocks implemented as MATLAB functions.

The pre-processing block was designed in a hybrid manner. Using the **SNAP** Desktop tool, **GRD S1A** data was pre-processed before being imported into MATLAB and subdivided into 512x512 transects. Metadata was then extracted, followed by the implementation of the **HH** technique in subsequent blocks. This involved using **NCEP** data to generate the first-guess wave spectrum along with the associated **SAR** spectrum. The final block in the pipeline implemented the **HH** inversion technique.

Chapter 5 followed, and verified the implementation of multiple sub-blocks within the designed pipeline. These tests utilised sample **S1A** data from Oct. 2, 2023, with the aim of confirming the desired function of each sub-block. Unfortunately, the final sub-block of the pipeline did not function as intended.

The error in this implementation was attributed to the calculation of the **HH** cost function. This error led to maximisation, as opposed to the expected minimisation seen in the literature. The analysis of this error revealed that 5 iterations produced the least **RE** with respect to the first-guess wave parameters. This was determined to be an acceptable compromise for accurate analysis of the best-fit wave parameters.

Acknowledging this inaccuracy, Chapter 6 analysed the results of the entire pipeline, whilst

investigating its effectiveness in various case studies. These compared the pipeline's reliability for varying ocean depths and different sea states. Regrettably, access to [CSIR](#) directional wave data was not obtained in time for analysis of these case studies. Instead, the relationships defined in Chapter 3 were used to assess the reliability and accuracy of the pipeline.

In these case studies, it was found that the pipeline's performance remained consistent regardless of ocean depth. Furthermore, the second case study found that while the pipeline had less than 5% [RE](#) for varying sea states, it overestimated [SWP](#) values for larger [FSWP](#).

In summary, this project successfully fulfilled its objectives through designing and implementing a pipeline that could be used to extract wave parameters from [SAR](#) data. Hence, this preliminary investigation can provide valuable insights into the development of a sea ice parameter extraction pipeline. Notably, the error with regards to the cost function proved to be the only problem in the pipeline's implementation. However, the largest limitation to this project was the inability to test the pipeline's accuracy due to a lack of ground-truth data.

Due to the project's limited scope and time frame, several aspects remained unexplored. The subsequent section is intended to present recommendations for future pipeline development.

7.1 Recommendations for Future Work

- Improve the implementation of the cost function, so that minimisation is achieved.
- Investigate the reliability and accuracy of the pipeline using different sized open ocean transects. In this report, only 512x512 sized transects were investigated.
- Obtain ground-truth directional wave data from the [CSIR](#) that can be used to further test the pipeline's accuracy.
- Use [SAR](#) data from various satellites that operate at different bands to assess robustness.

7.1.1 Pre-processing

- Scripts should be developed for the seamless, automated retrieval of [S1A](#) data.
- Enhance the pre-processing component of the pipeline by using the industry standard [SNAP](#) Python library.
- Investigate alternative and additional pre-processing techniques for [S1A](#) data to improve pipeline accuracy. These may include but are not limited to Ellipsoid Correction, Multi-looking, and Radiometric Calibration using gamma and beta options. [Add citation to snap reference manual]

7.1.2 [SAR](#) Imaging of Ocean Waves

Since this project served as an initial exploration into the development of a wave parameter extraction pipeline, only the [HH](#) technique was implemented. Future versions of this pipeline should include the implementation of other wave parameter extraction algorithms. The following research papers, [62, 63, 79], are all centred around the [HH](#) technique but optimise different aspects of the process. All of these algorithms are discussed in Chapter 2.

Appendix A

Additional Theory Definitions

This appendix serves as additional theoretical context for the introductory theory provided in Chapter 3.

A.1 Pre-processing Techniques

A.1.1 Speckle

Speckle, often noted for its "salt-and-pepper" appearance [6] in SAR images, results from the interference of multiple scatters within a resolution cell which leads to varying intensity values. Even when imaging a uniform surface, speckle exists due to phase variations amongst scatterers within the resolution cell, with brighter image regions exhibiting more intense speckle [5, 6]. Speckle is not an error in a SAR image, but a consequence of imaging with finite resolution. Higher resolution weakens the speckle effect as it reduces the number of scatterers within the resolution cell [6]. Unprocessed intensity VV GRD data is shown in Figure A.1 and speckle can be seen in both sub-scenes.

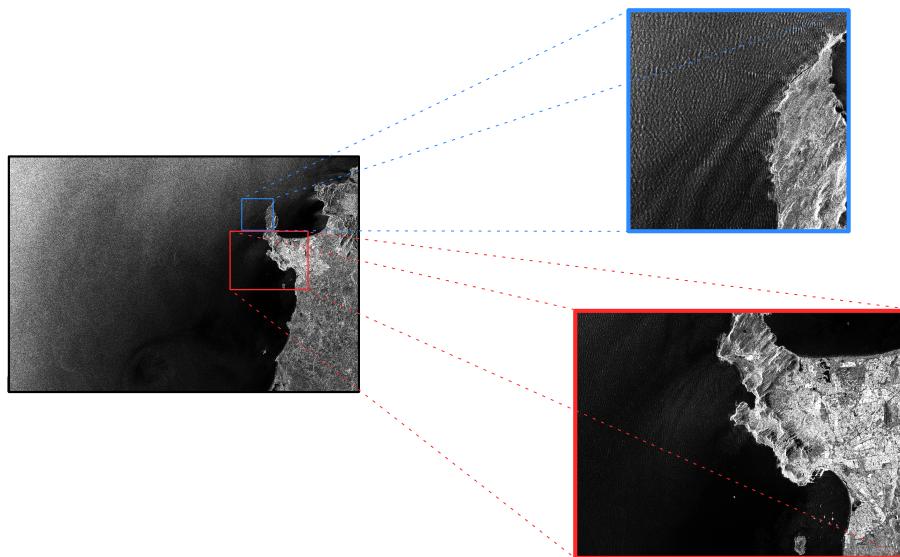


Figure A.1: Unprocessed GRD S1A data.

A.1.2 Thermal Noise

Thermal noise significantly affects SAR image quality. It represents inherent background energy generated by the radar receiver channel and creates a noise threshold. When a received signal falls below this threshold, it becomes indistinguishable from thermal noise, which impacts low-intensity scatterers, such as new ice and open water [80], especially in cross-polarised applications. In these cases, the thermal noise can significantly impact the interpretability of the SAR data [80]. Unlike speckle, which appears as grainy patterns and can be reduced through filtering, thermal noise is characterised by a uniform background noise level. In the context of S1A, thermal noise is composed of two additive noise sources related to antenna movement and scalloping noise. It predominantly affects TOPSAR imaging mode, especially for cross-polarised images.

Removal of S1A thermal noise results in improved image quality and can be visually seen in Figure A.2 when compared to Figure A.1, particularly over land regions in the second sub-scene.

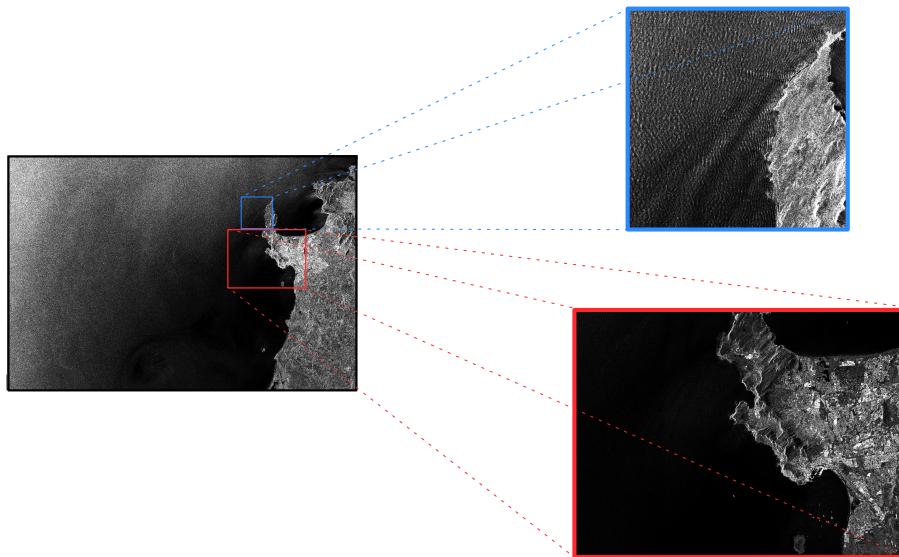


Figure A.2: GRD S1A data with thermal noise removed using SNAP.

A.1.3 Radiometric Calibration

In SAR imaging, σ_0 represents the backscatter coefficient, which indicates the radar signal's strength reflected by scatterers. σ_0 is a dimensionless number that accounts for variations in factors like incidence angle, wavelength, polarisation, and surface properties. Radiometric calibration addresses Radar Cross Section (RCS) variations, ensuring that pixel values in SAR images are both qualitatively representative and quantitatively aligned with σ_0 . This process corrects radiometric bias present in raw SAR data, enabling its use in quantitative applications like target recognition and environmental monitoring. Calibration factors in the scattering area, antenna gain pattern, and range spread loss, enhance the accuracy and reliability of SAR data.

Radiometric calibration of S1A GRD data can be visually seen in Figure A.3 over mountainous regions in both sub-scenes. This is due to the correction of backscatter bias which doesn't account for elevation changes.

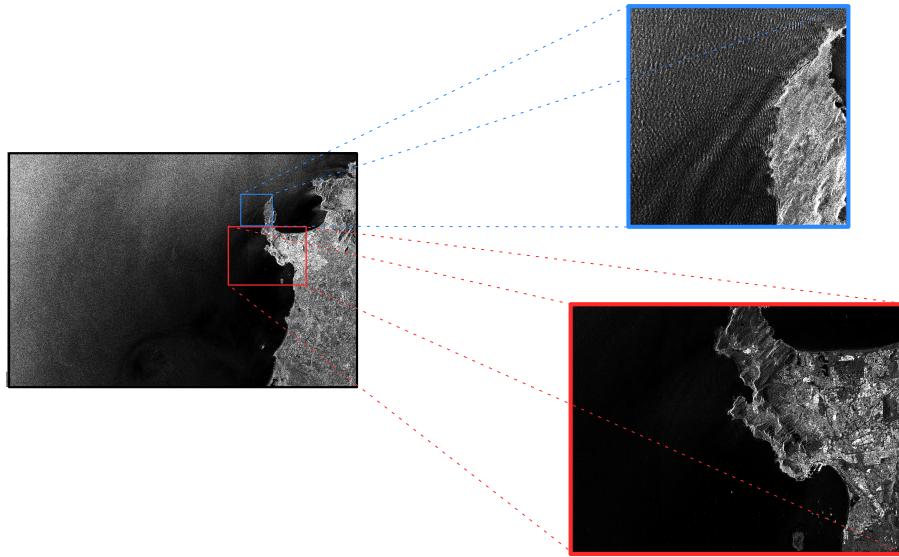


Figure A.3: Radiometrically calibrated GRD S1A data done using SNAP.

A.2 Polarisation

Polarisation is a fundamental aspect of SAR imaging. Polarisation refers to the orientation of the plane in which an EM wave oscillates as it propagates. Linear polarisation maintains a constant orientation along the wave's path, while circular and elliptical polarisations involve changing oscillation plane orientations, forming shapes like ellipses or circles [6].

Traditionally, most SAR systems used single-polarised sensors, predominantly operating in HH (horizontal transmit, horizontal receive) or VV (vertical transmit, vertical receive) polarisation. However, newer SAR sensors offer dual-polarisation or quad-polarisation capabilities. These sensors can transmit both horizontally and vertically polarised waveforms and receive signals from both polarisations. Understanding the polarisation of a SAR image is crucial as it affects how radar signals interact with objects on the ground, influencing the recorded radar brightness for specific polarisations.

To interpret polarimetric SAR data effectively, it is helpful to consider how different types of scatterers interact with different polarizations. Rough surface scatterers, double-bounce scatterers, and volume scatterers make up the three primary categories of scatterers in a scene. Different polarimetric channels exhibit varying preferences for these scattering types, and these preferences can be used to analyse and classify scattering types.

A.2.1 Scattering

Scattering is important to understand how satellite sensors detect and interpret EM waves. Scattering on the Earth's surface involves two primary processes: surface and volume scattering. Surface roughness plays a critical role in EM backscatter. A flat or specular surface¹ behaves like a mirror, reflecting radiation back to the sensor only when the surface aligns perpendicularly with the sensor's line of sight. This occurs with flat or tilted surfaces, such as mountains or ocean surface waves. In contrast, rougher surfaces scatter a portion of incident EM radiation

¹A specular surface reflects an EM wave off its surface at an angle equal to the incidence angle.

back toward the sensor.

When the surface has a regular structure, such as waves with a specific wavelength, it can result in positive interference, a phenomenon known as Bragg scattering. Bragg scattering is of particular interest in the context of ocean waves and is caused by short ripple waves on the ocean surface [59]. This phenomenon is characterised by the constructive interference of incoming and backscattered EM radiation due to the regular structure of waves. This creates high backscatter values. The key factor for Bragg scattering's occurrence is the consistency between the wavelength of the electromagnetic waves and the spacing between wave crests.

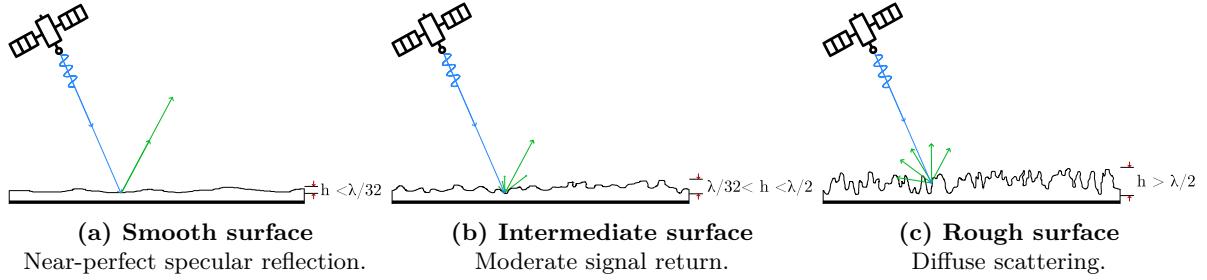


Figure A.4: Comparison of surface roughness and surface backscatter. Adapted from [6].

A.3 Hasselmann and Hasselmann (HH) Variable Definitions

The variables used in Equation 3.62 are defined as

$$\delta P = \hat{P}(\underline{k}) - P_n(\underline{k}) \quad (\text{A.1})$$

$$\delta E_k = \hat{E}(\underline{k}) - E_n(\underline{k}) \quad (\text{A.2})$$

$$A_k = W_k^2 + 2\mu \quad (\text{A.3})$$

$$B_k = W_k + W_{-k} \quad (\text{A.4})$$

$$W_k = |T_k^S|^2 \exp [-k_x^2 \xi_n'^2] \quad (\text{A.5})$$

Appendix B

SAR Spectrum Plots

B.1 Pipeline Verification

B.1.1 Power Spectral Expansions

The plots shown in Figure 5.6 without a spectral threshold, and zoomed-out versions are all shown in Figure B.1 through Figure B.3.

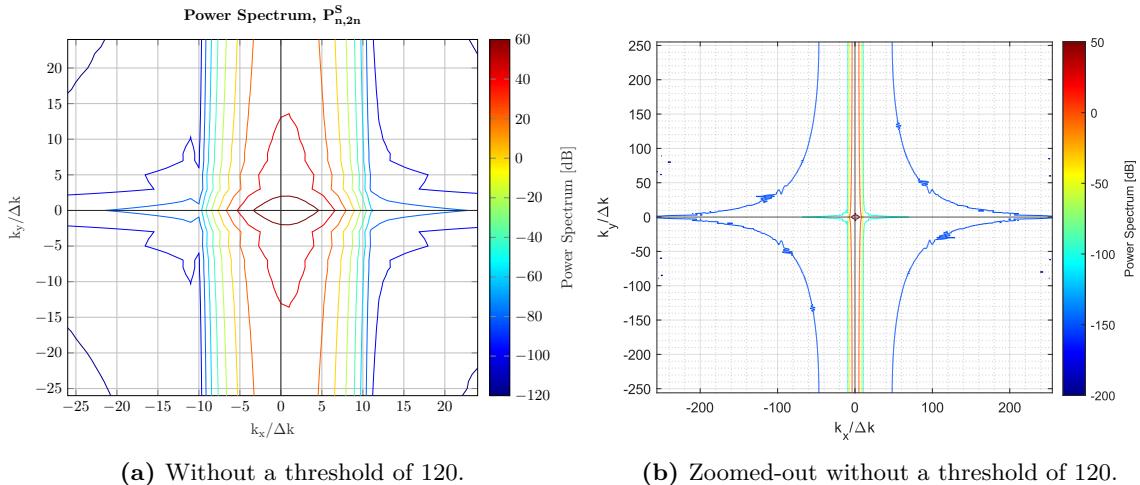


Figure B.1: Spectral expansion of $P_{n,2n}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the `spectralExpansion2n`, and `SARSpectrumPlot` functions.

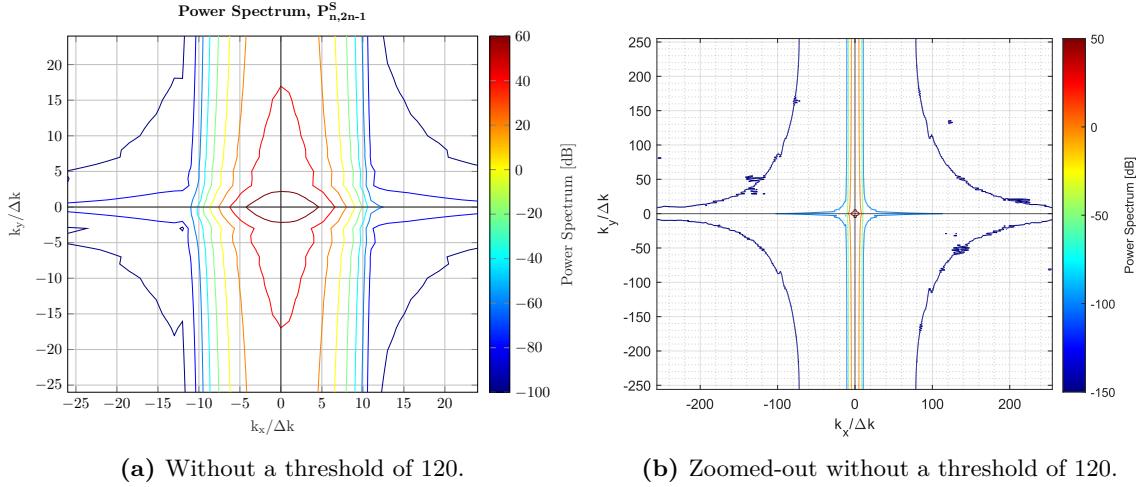


Figure B.2: Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the `spectralExpansion2n_1`, and `SARspectrumPlot` functions.

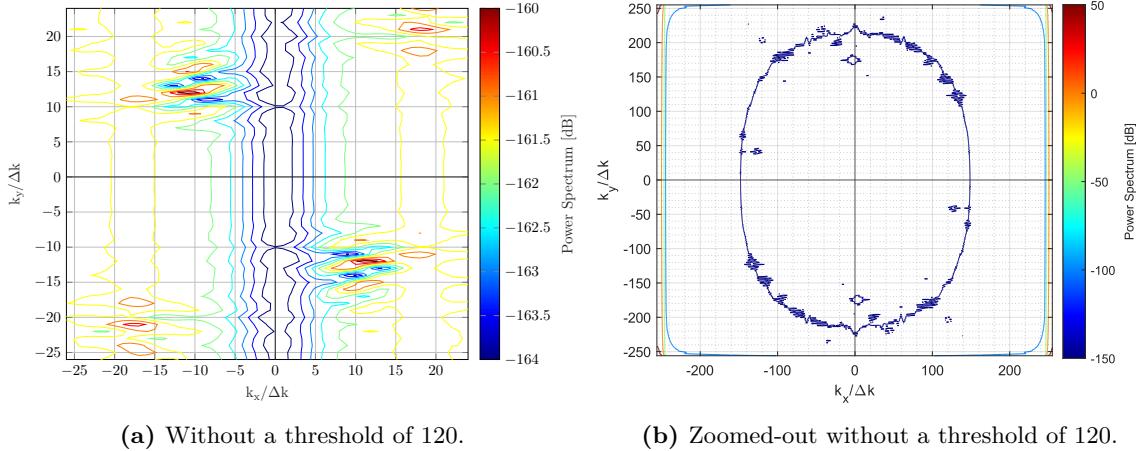


Figure B.3: Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views. Generated using NCEP wave data, the `spectralExpansion2n_2`, and `SARspectrumPlot` functions.

Figure B.3a shows that all of the seen values were very small, and this makes sense, as when applying the threshold of 120 in Section 5.3.1, this results in an empty plot.

B.1.2 Power Spectral Multiplication with Coefficient

The plots shown in Figures 5.6 and 5.9 without a spectral threshold, and zoomed-out versions are all shown in Figure B.4 through Figure B.6.

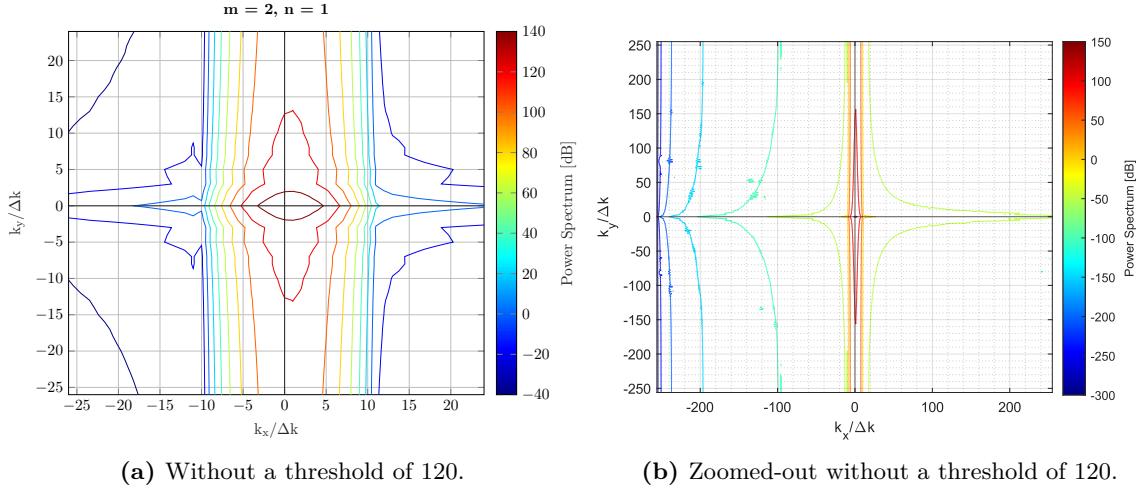


Figure B.4: Spectral expansion of $P_{n,2n}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x \beta)$.

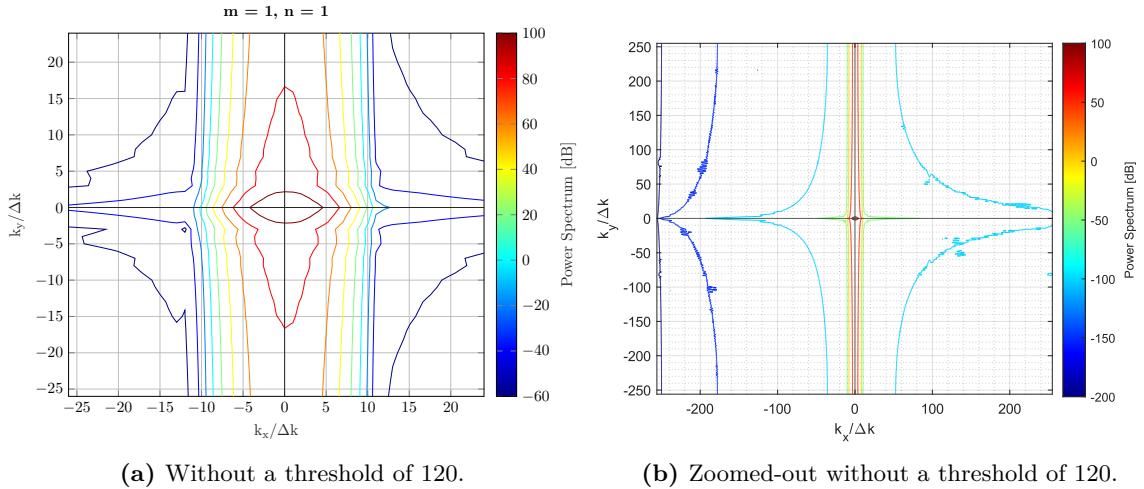
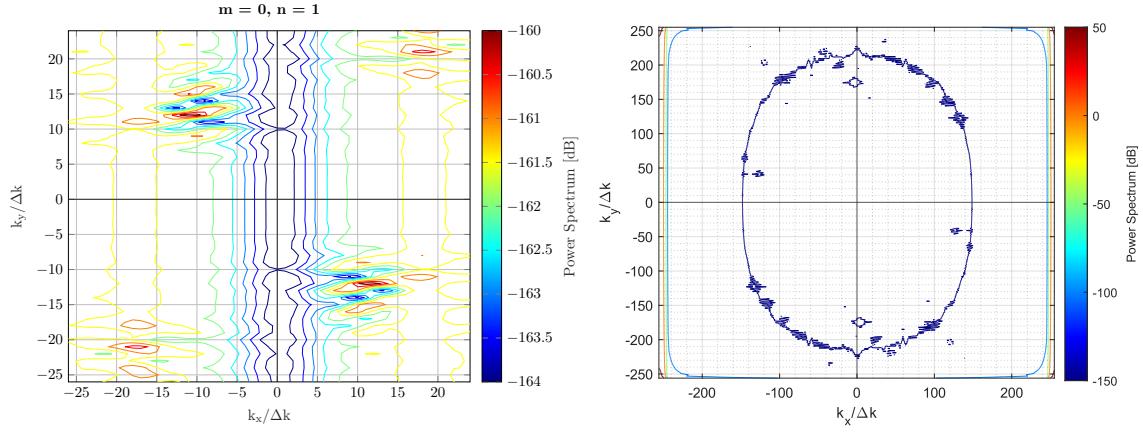


Figure B.5: Spectral expansion of $P_{n,2n-1}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x \beta)^2$.



(a) Without a threshold of 120.

(b) Zoomed-out without a threshold of 120.

Figure B.6: Spectral expansion of $P_{n,2n-2}^S$ for the wave spectrum shown in Figure 5.5a, from 02-Oct-2023 at 34° S, 18° E in different views after multiplication with $(k_x\beta)^0 = 1$.

Comparing Figure B.6 with Figure B.3, shows no change in the SAR spectrum. This verifies the implementation of the multiplication.

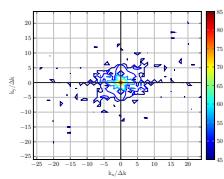
Comparing Figure B.4b and Figure B.5b to their spectra prior to multiplication in Figure B.1b and Figure B.2b respectively, the difference in the coefficients discussed in Section 5.3.1 can be seen. The components at approximately $k_x = -100$ in Figure B.4b can be seen to become straight and follow the k_y axis grid. This verifies the multiplication by $(k_x\beta)$. In contrast, the spectral components at approximately $k_x = 50$ in Figure B.5b, can be seen to follow a more parabolic shape than that of Figure B.4b. This verifies the multiplication by $(k_x\beta)^2$.

Appendix C

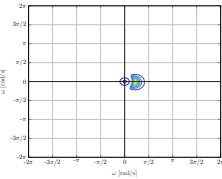
Case Study Testing

A tabulated overview of all relevant figures which detail the evolution of pipeline spectra is shown initially for both dates, and all transects, before all plots are shown in the relevant sections.

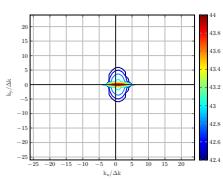
2. Observed SAR



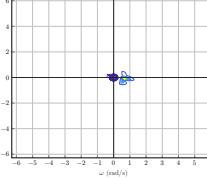
First-guess wave



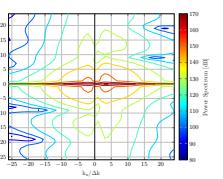
First-guess SAR



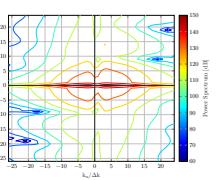
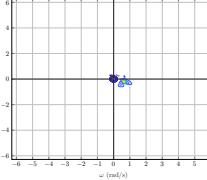
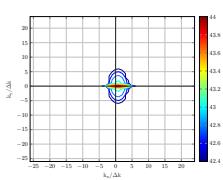
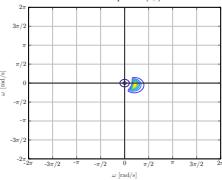
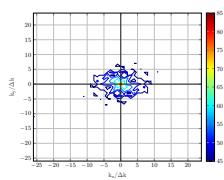
Best-fit wave



Best-fit SAR



3.



4.

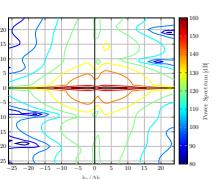
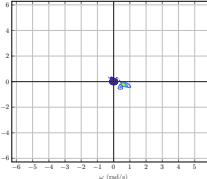
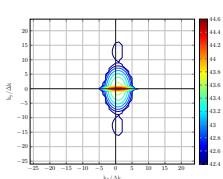
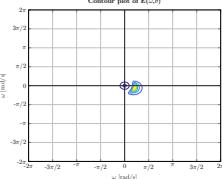
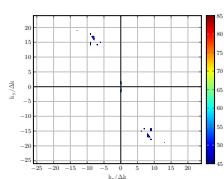
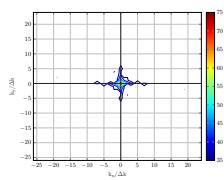
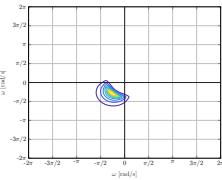


Figure C.1: Observed S1A SAR spectra, first-guess JONSWAP modelled wave spectrum, calculated first-guess SAR spectrum, best-fit wave spectrum, and best-fit SAR spectrum for all three transect locations, with labelled rows, as depicted in Figure 6.4a, for Oct. 15, 2023.

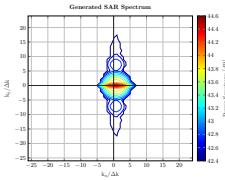
1. Observed SAR



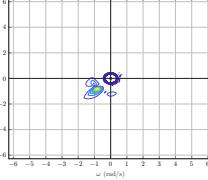
First-guess wave



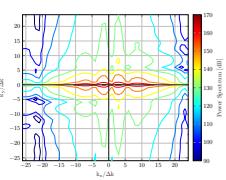
First-guess SAR



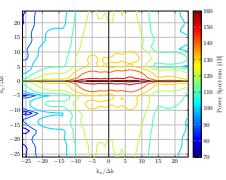
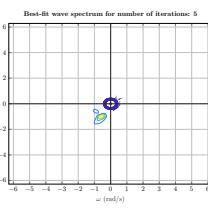
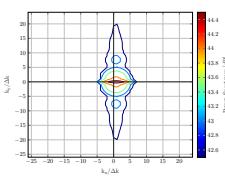
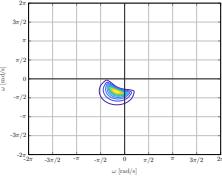
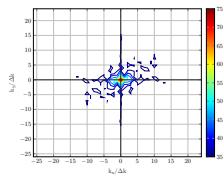
Best-fit wave



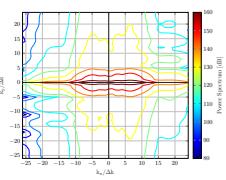
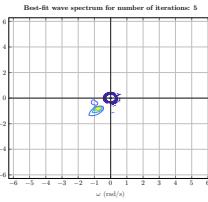
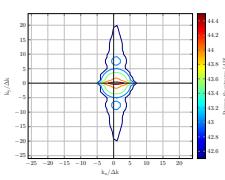
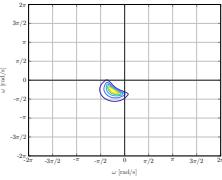
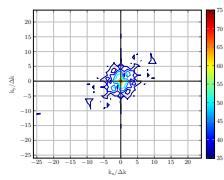
Best-fit SAR



2.



3.



4.

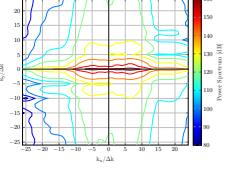
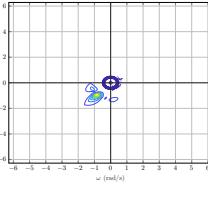
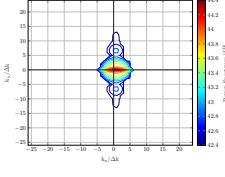
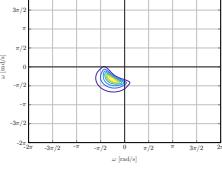
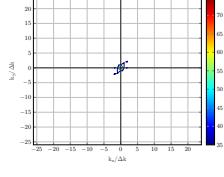


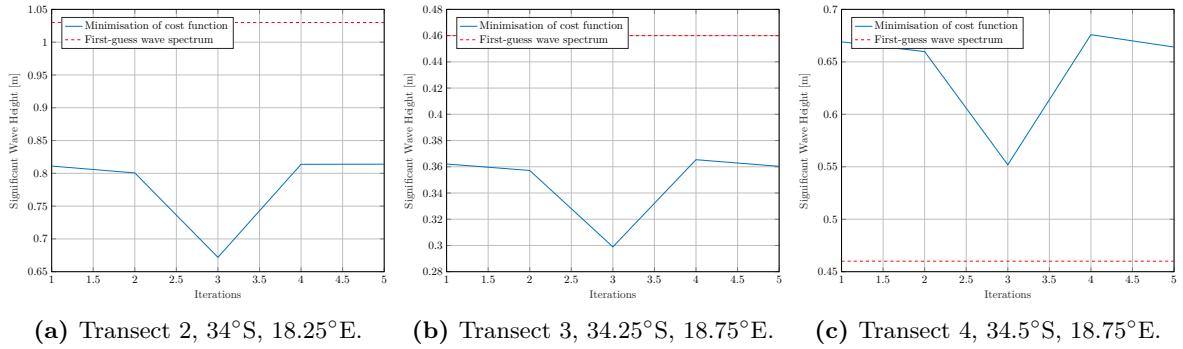
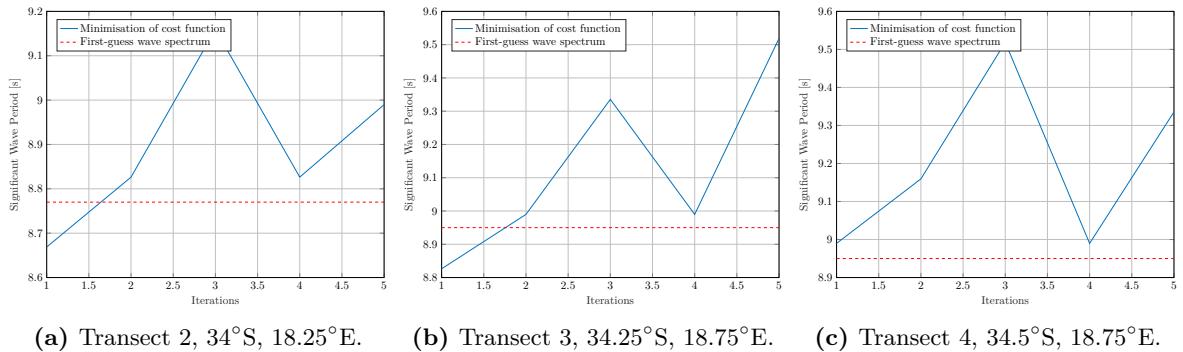
Figure C.2: Observed S1A SAR spectra, first-guess JONSWAP modelled wave spectrum, calculated first-guess SAR spectrum, best-fit wave spectrum, and best-fit SAR spectrum for all four transect locations, with labelled rows, as depicted in Figure 6.2, for Oct. 20, 2023.

C.1 October 15, 2023

This section contains additional data and plots from October 15, 2023, which are included to aid the case study testing done in Chapter 6.

C.1.1 Output Parameters

The best-fit values for significant wave height and period are plotted for each iteration in Figure C.3 and Figure C.4 respectively.

**Figure C.3:** Best-fit significant wave height per iteration for Oct. 20, 2023.**Figure C.4:** Best-fit significant wave period per iteration for Oct. 20, 2023.

The mean values of the **SWH** and **SWP** are shown in Table C.1.

Date	$\hat{H}_{1/3}$ [m]	$\hat{T}_{1/3}$ [s]
Oct. 15, 2023	0.78	7.71
Oct. 20, 2023	1.45	5.16

Table C.1: Mean values of **SWH** and **SWP** for both sea states.

The relative error of the data in Table 6.5, when compared to the data in Table 6.4 was calculated using Equation 5.1, and are shown in Table C.2.

Transect	Location	Depth	$RE_{H_{1/3}}$ [%]	$RE_{T_{1/3}}$ [%]
2	34°S, 18.25°E	Oct. 15, 2023	-20.36	-0.82
		Oct. 20, 2023	-20.98	2.51
3	34.25°S, 18.75°E	Oct. 15, 2023	-20.82	-1.39
		Oct. 20, 2023	-21.67	6.35
4	34.5°S, 18.75°E	Oct. 15, 2023	-20.66	-1.52
		Oct. 20, 2023	-21.87	3.61

Table C.2: **RE** for all transects for October 15, 2023 and October 20, 2023 after 5 iterations.

C.1.2 SAR Spectra Plots

Additional SAR spectra plots are shown in this section. These plots aid the case study testing done in Chapter 6.

Observed SAR Transects

Each individual transect is shown in Figure C.5, with a colour bar representing the intensity value measured.

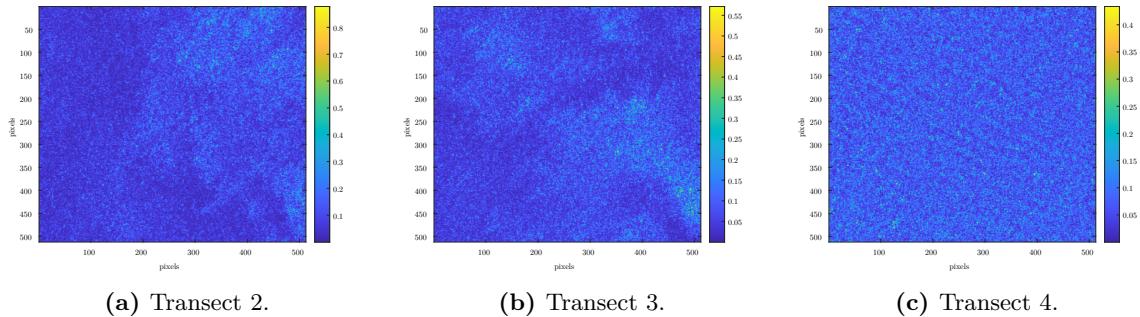


Figure C.5: Full resolution SAR data transects, from Figure 6.4a. These transects were used to calculate the SAR spectra in Figure C.7.

Conventional greyscale plots of SAR transect data is shown in Figure C.6.

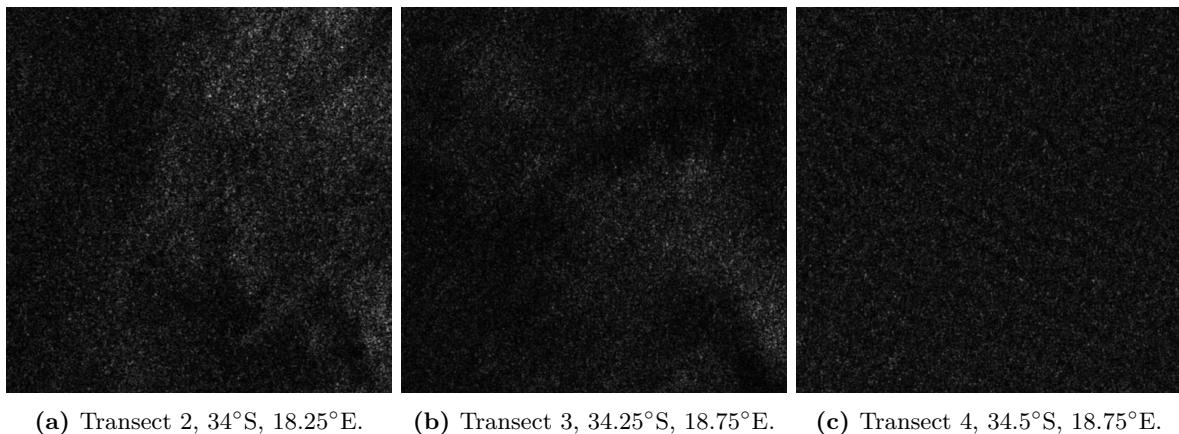


Figure C.6: Full resolution SAR data transects, from Figure 6.4a. These transects were used to calculate the SAR spectra in Figure C.7.

The intensity SAR spectra of SAR transect data is shown in Figure C.7.

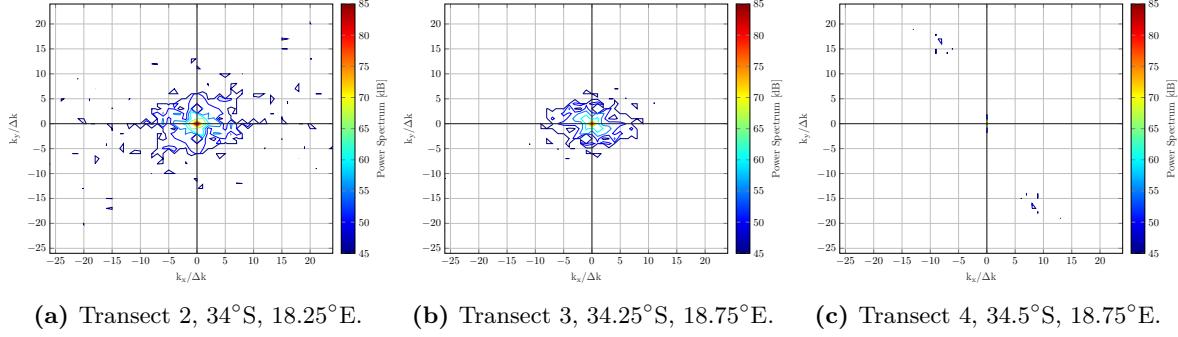


Figure C.7: Observed SAR intensity spectra of data transects, from Figure 6.4a.

The plots shown in Figure C.7 without a spectral threshold, and zoomed-out versions are all shown in Figure C.8 and Figure C.9 respectively.

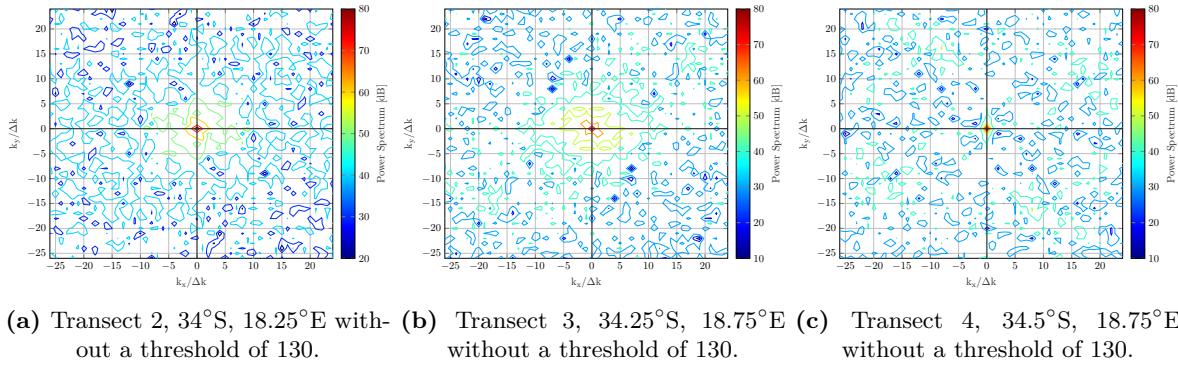


Figure C.8: Observed SAR intensity spectra of data transects, from Figure 6.4a,
without a spectral threshold.

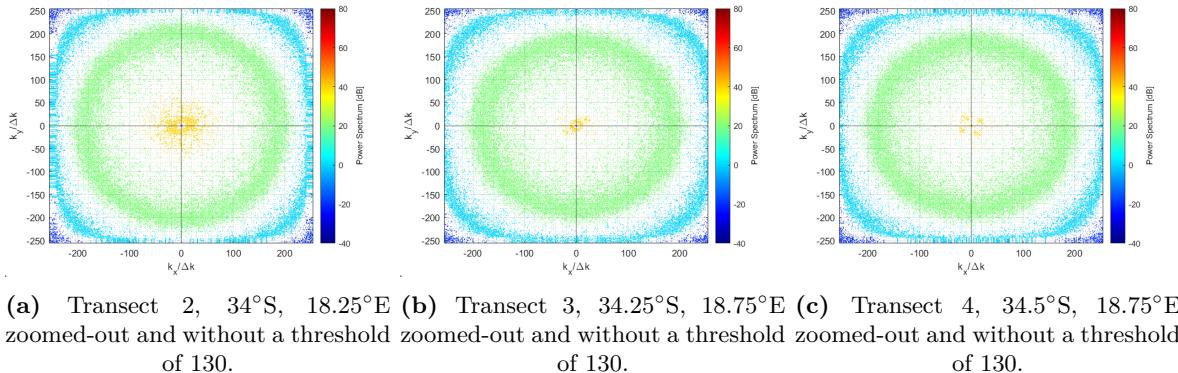


Figure C.9: Full view of observed SAR intensity spectra of data transects, from
Figure 6.4a, without a spectral threshold.

HH Generated SAR Spectra

The HH generated equivalent SAR spectra of wave spectra data is shown in Figure C.10.

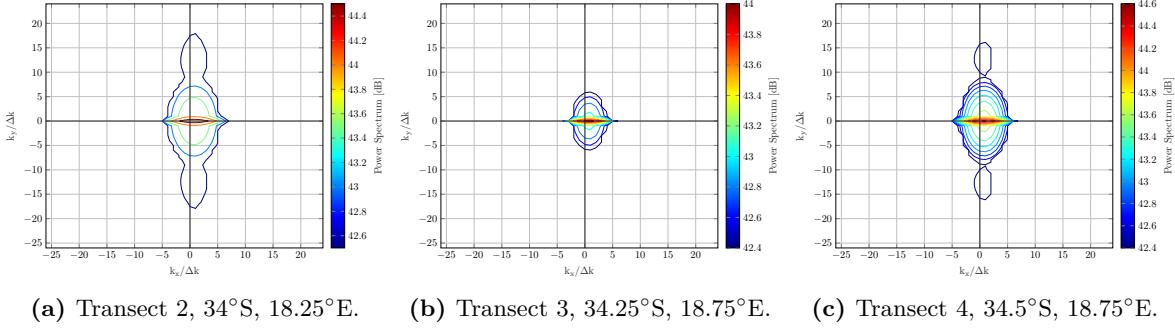


Figure C.10: Generated HH SAR spectra of wave spectra, from Figure 6.4a.

The plots shown in Figure C.10 without a spectral threshold, are shown in Figure C.11.

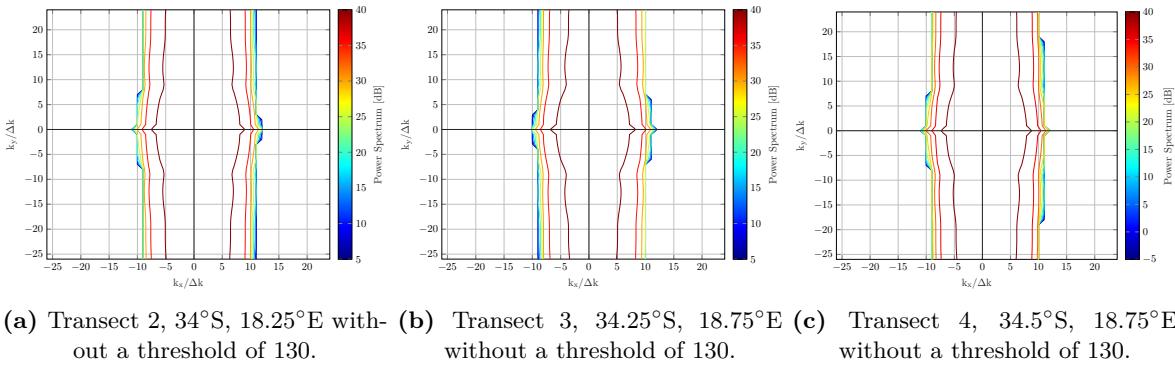


Figure C.11: Generated HH SAR spectra of wave spectra, from Figure 6.5, without a spectral threshold.

Best-fit SAR Spectra

The best-fit SAR spectra after 5 iterations is shown in Figure C.12.

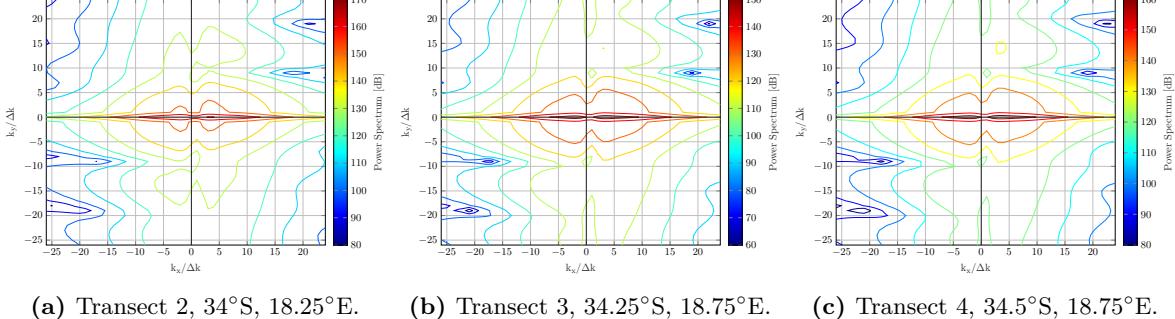
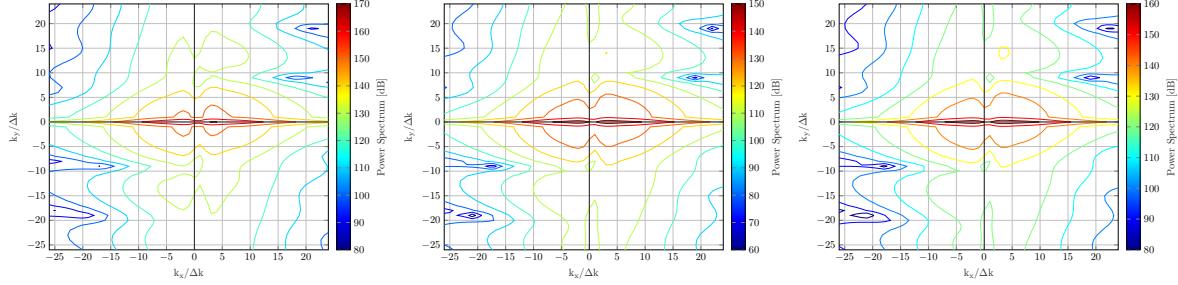


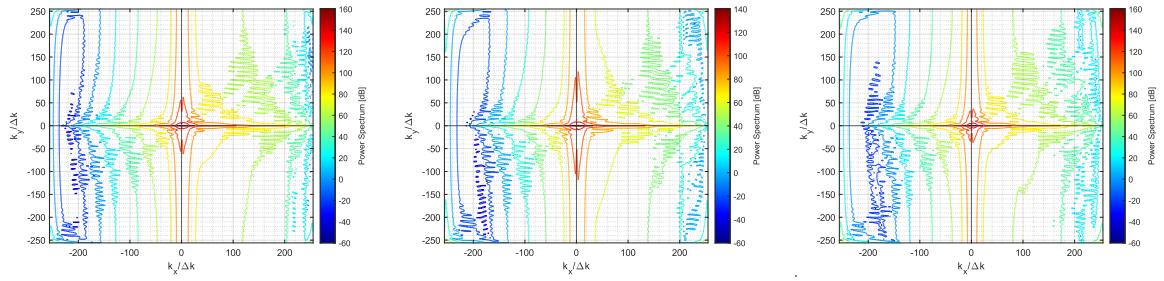
Figure C.12: Best-fit SAR spectra.

The plots shown in Figure C.12 without a spectral threshold, and zoomed-out versions are all shown in Figure C.13 and Figure C.14 respectively.



(a) Transect 2, 34°S, 18.25°E with- (b) Transect 3, 34.25°S, 18.75°E (c) Transect 4, 34.5°S, 18.75°E
out a threshold of 130. without a threshold of 130. without a threshold of 130.

Figure C.13: Best-fit HH SAR spectra, without a spectral threshold.



(a) Transect 2, 34°S, 18.25°E (b) Transect 3, 34.25°S, 18.75°E (c) Transect 4, 34.5°S, 18.75°E
zoomed-out and without a threshold zoomed-out and without a threshold zoomed-out and without a threshold
of 130. of 130. of 130.

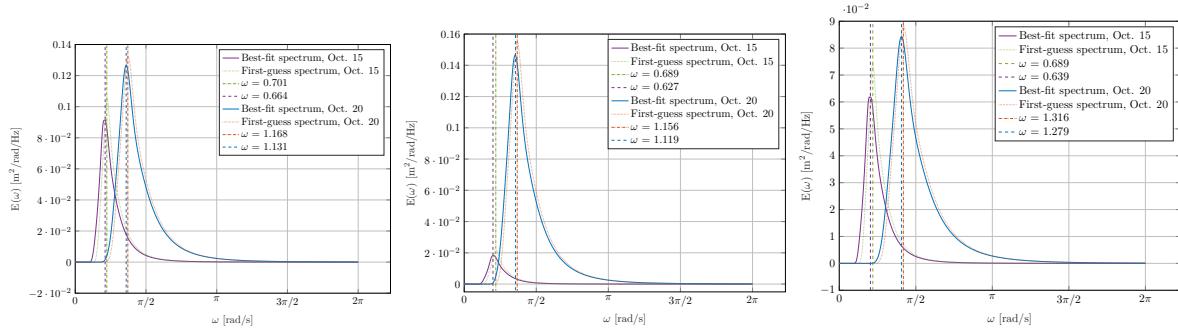
Figure C.14: Full view of best-fit SAR spectra, without a spectral threshold.

C.1.3 Wave Spectra Plots

Additional wave spectra plots are shown in this section. These plots aid the case study testing done in Chapter 6.

Full range One-dimensional Wave Spectra

Zoomed out version of the plots produced in Figure 6.5 can be found in Figure C.15.



(a) Transect 2, 34°S, 18.25°E. (b) Transect 3, 34.25°S, 18.75°E. (c) Transect 4, 34.5°S, 18.75°E.

Figure C.15: Best-fit and first-guess wave spectra after 5 iterations for all transects. Zoomed out version plots of Figure 6.5

Best-fit Frequency-direction Wave Spectra

The associated frequency-directional wave spectra for the best-fit one-dimensional wave spectra in Figure 6.5 are shown in Figure C.16.

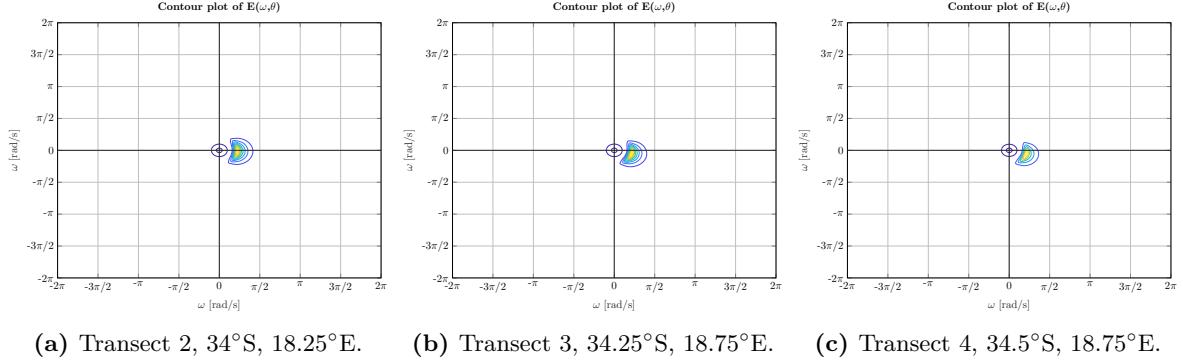


Figure C.16: Two-dimensional first-guess frequency-direction wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.

Best-fit Wave-number Wave Spectra

The associated frequency-directional wave spectra for the best-fit one-dimensional wave spectra in Figure 6.5 are shown in Figure C.17.

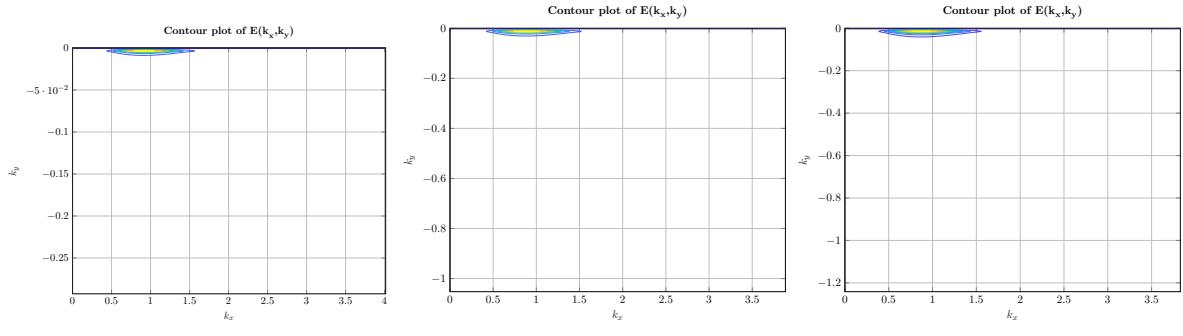


Figure C.17: Two-dimensional first-guess wave-number wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.

C.2 October 20, 2023

This section contains additional data and plots from October 20, 2023, which are included to aid the case study testing done in Chapter 6.

C.2.1 Output Parameters

The best-fit values for significant wave height and period are plotted for each iteration in Figure C.18 and Figure C.19 respectively.

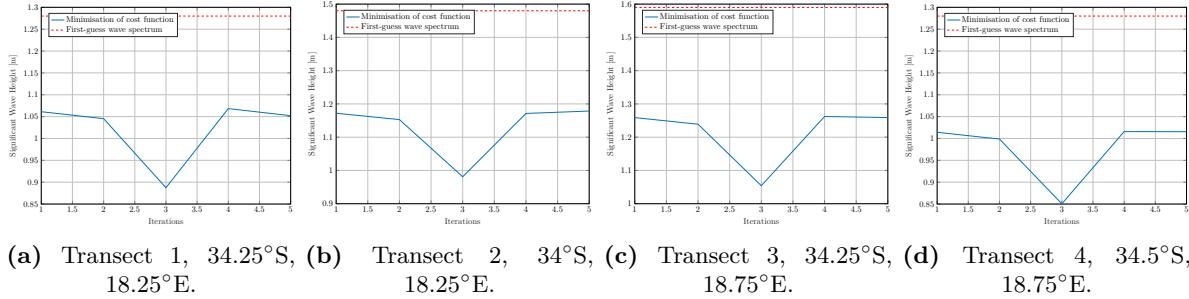


Figure C.18: Best-fit significant wave height per iteration for Oct. 20, 2023.

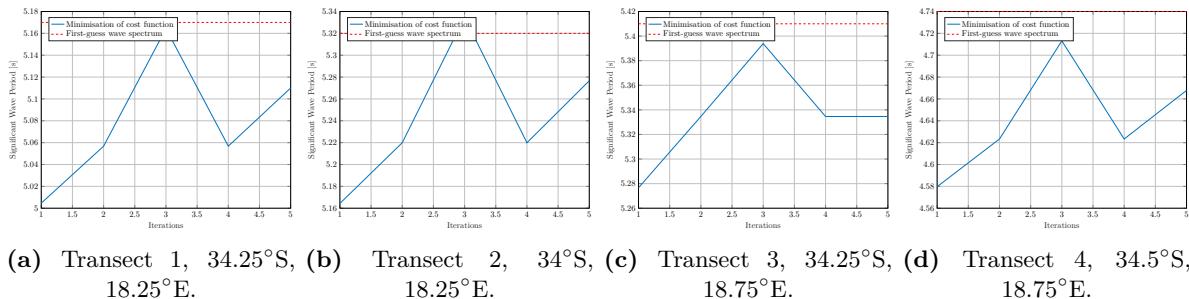


Figure C.19: Best-fit significant wave period per iteration for Oct. 20, 2023.

The relative error of the data in Table 6.3, when compared to the data in Table 6.2 was calculated using Equation 5.1, and are shown in Table C.3.

Transect	Location	Depth	RE _{H_{1/3}} [%]	RE _{T_{1/3}} [%]
1	34.25°S, 18.25°E	130 m	-21.48	-1.16
2	34°S, 18.25°E	80 m	-20.36	-0.82
3	34.25°S, 18.75°E	50 m	-20.82	-1.39
4	34.5°S, 18.75°E	125 m	-20.66	-1.52

Table C.3: RE for all transects for October 20, 2023 after 5 iterations.

C.2.2 SAR Spectra Plots

Additional SAR spectra plots are shown in this section. These plots aid the case study testing done in Chapter 6.

Observed SAR Transects

Each individual transect is shown in Figure C.20, with a colour bar representing the intensity value measured.

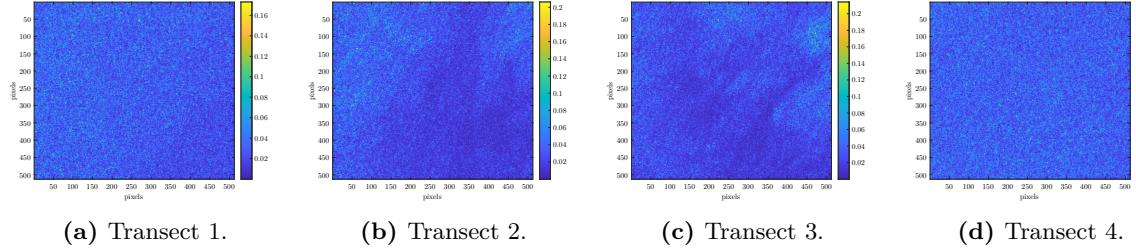
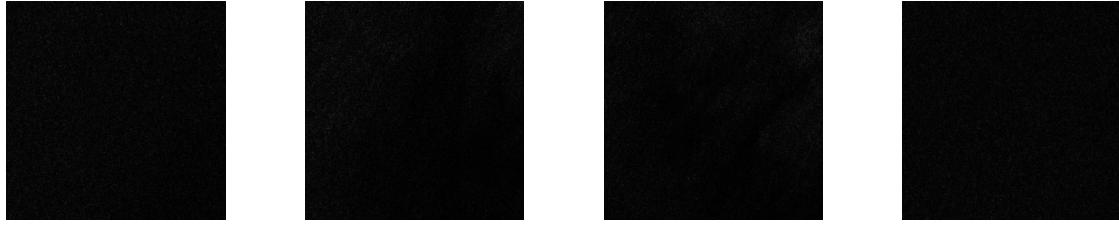


Figure C.20: Full resolution **SAR** data transects, from Figure 6.2. These transects were used to calculate the **SAR** spectra in Figure C.22.

Conventional greyscale plots of **SAR** transect data is shown in Figure C.21.



(a) Transect 1, 34.25°S, 18.25°E. (b) Transect 2, 34°S, 18.25°E. (c) Transect 3, 34.25°S, 18.75°E. (d) Transect 4, 34.5°S, 18.75°E.

Figure C.21: Full resolution **SAR** data transects, from Figure 6.2. These transects were used to calculate the **SAR** spectra in Figure C.22.

The intensity **SAR** spectra of **SAR** transect data is shown in Figure C.22.

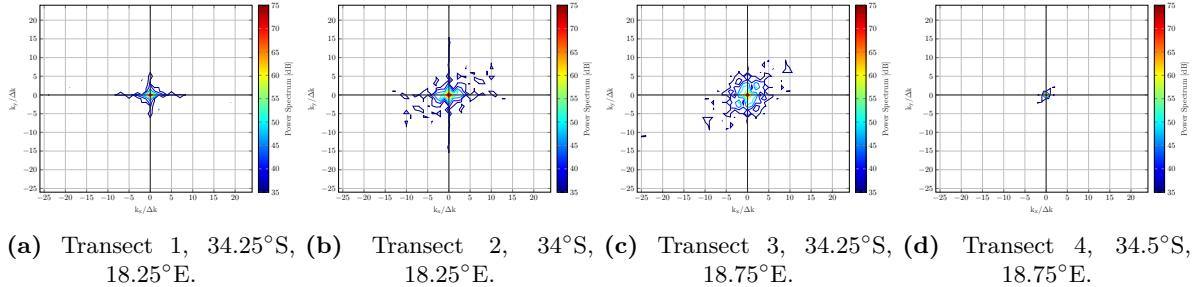
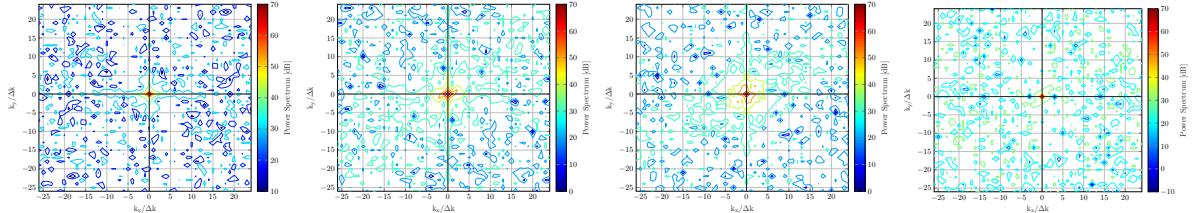


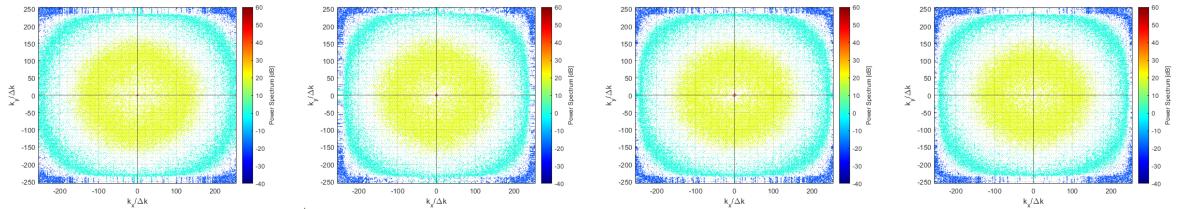
Figure C.22: Observed **SAR** intensity spectra of data transects, from Figure 6.2.

The plots shown in Figure C.22 without a spectral threshold, and zoomed-out versions are all shown in Figure C.23 and Figure C.24 respectively.



(a) Transect 1, 34.25°S, (b) Transect 2, 34°S, (c) Transect 3, 34.25°S, (d) Transect 4, 34.5°S, 18.25°E without a thresh- 18.25°E without a thresh- 18.75°E without a thresh- 18.75°E without a thresh- old of 130. old of 130. old of 130. old of 130.

Figure C.23: Observed SAR intensity spectra of data transects, from Figure 6.2, without a spectral threshold.

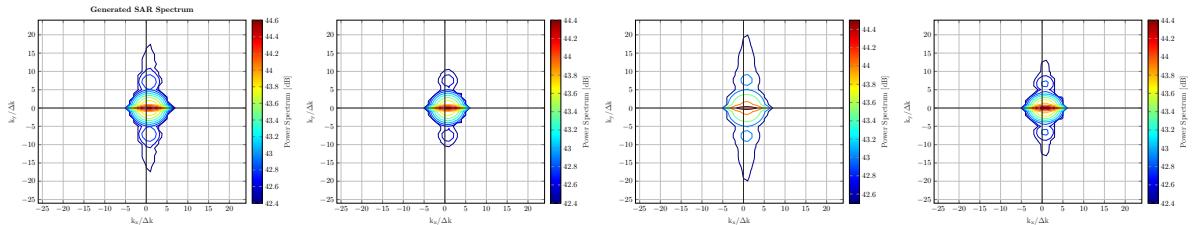


(a) Transect 1, 34.25°S, (b) Transect 2, 34°S, (c) Transect 3, 34.25°S, (d) Transect 4, 34.5°S, 18.25°E zoomed-out and 18.25°E zoomed-out and 18.75°E zoomed-out and 18.75°E zoomed-out and without a threshold of 130. without a threshold of 130. without a threshold of 130. without a threshold of 130.

Figure C.24: Full view of observed SAR intensity spectra of data transects, from Figure 6.2, without a spectral threshold.

HH Generated SAR Spectra

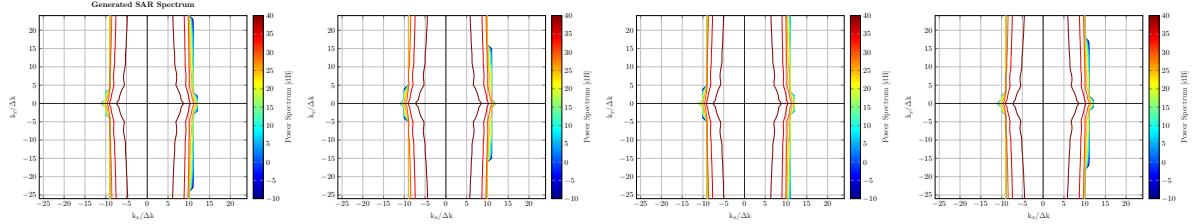
The HH generated equivalent SAR spectra of wave spectra data is shown in Figure C.25.



(a) Transect 1, 34.25°S, (b) Transect 2, 34°S, (c) Transect 3, 34.25°S, (d) Transect 4, 34.5°S, 18.25°E. 18.25°E. 18.75°E. 18.75°E.

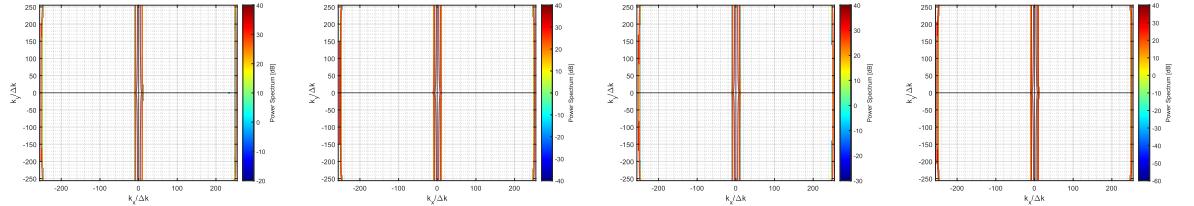
Figure C.25: Generated HH SAR spectra of wave spectra, from Figure 6.3.

The plots shown in Figure C.25 without a spectral threshold, and zoomed-out versions are all shown in Figure C.26 and Figure C.27 respectively.



(a) Transect 1, 34.25°S , (b) Transect 2, 34°S , (c) Transect 3, 34.25°S , (d) Transect 4, 34.5°S , 18.25°E without a threshold. (a) Transect 1, 18.25°E without a threshold. (b) Transect 2, 18.75°E without a threshold. (c) Transect 3, 18.75°E without a threshold. (d) Transect 4, 18.75°E without a threshold.

Figure C.26: Generated HH SAR spectra of wave spectra, from Figure 6.3, without a spectral threshold.

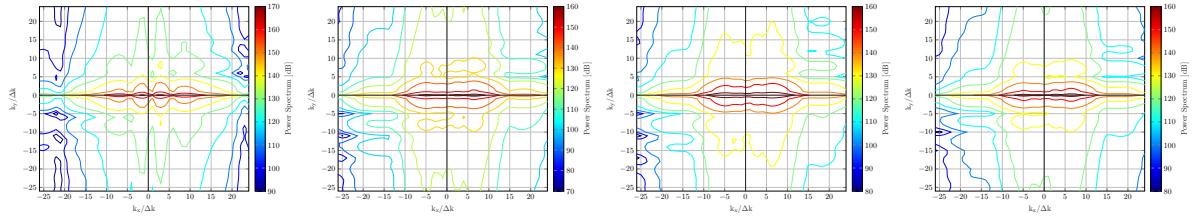


(a) Transect 1, 34.25°S , (b) Transect 2, 34°S , (c) Transect 3, 34.25°S , (d) Transect 4, 34.5°S , 18.25°E zoomed-out and 18.25°E zoomed-out and 18.75°E zoomed-out and 18.75°E zoomed-out and without a threshold of 130. without a threshold of 130. without a threshold of 130. without a threshold of 130.

Figure C.27: Full view of generated HH SAR spectra of wave spectra, from Figure 6.3, without a spectral threshold.

Best-fit SAR Spectra

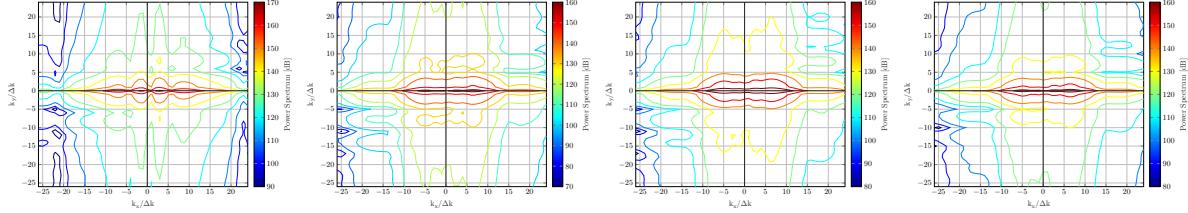
The best-fit SAR spectra after 5 iterations is shown in Figure C.28.



(a) Transect 1, 34.25°S , (b) Transect 2, 34°S , (c) Transect 3, 34.25°S , (d) Transect 4, 34.5°S , 18.25°E . (a) Transect 1, 18.25°E . (b) Transect 2, 18.75°E . (c) Transect 3, 18.75°E . (d) Transect 4, 18.75°E .

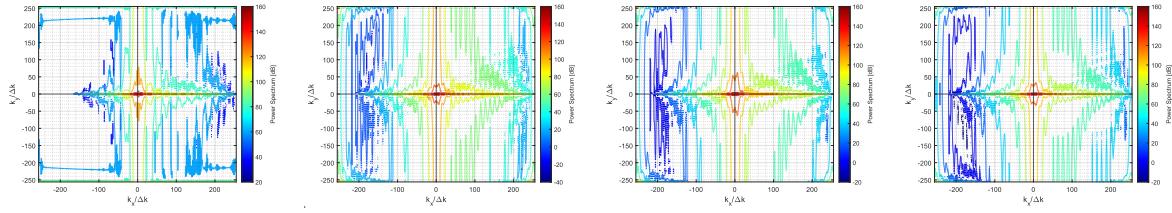
Figure C.28: Best-fit SAR spectra.

The plots shown in Figure C.28 without a spectral threshold, and zoomed-out versions are all shown in Figure C.29 and Figure C.30 respectively.



(a) Transect 1, 34.25°S , (b) Transect 2, 34°S , (c) Transect 3, 34.25°S , (d) Transect 4, 34.5°S , 18.25°E without a thresh- 18.25°E without a thresh- 18.75°E without a thresh- 18.75°E without a thresh-
old of 130. old of 130. old of 130. old of 130.

Figure C.29: Best-fit HH SAR spectra, without a spectral threshold.



(a) Transect 1, 34.25°S , (b) Transect 2, 34°S , (c) Transect 3, 34.25°S , (d) Transect 4, 34.5°S , 18.25°E zoomed-out and 18.25°E zoomed-out and 18.75°E zoomed-out and 18.75°E zoomed-out and without a threshold of 130. without a threshold of 130. without a threshold of 130. without a threshold of 130.

Figure C.30: Full view of best-fit SAR spectra, without a spectral threshold.

C.2.3 Wave Spectra Plots

Additional wave spectra plots are shown in this section. These plots aid the case study testing done in Chapter 6.

Full range One-dimensional Wave Spectra

Zoomed out version of the plots produced in Figure 6.3 can be found in Figure C.31.

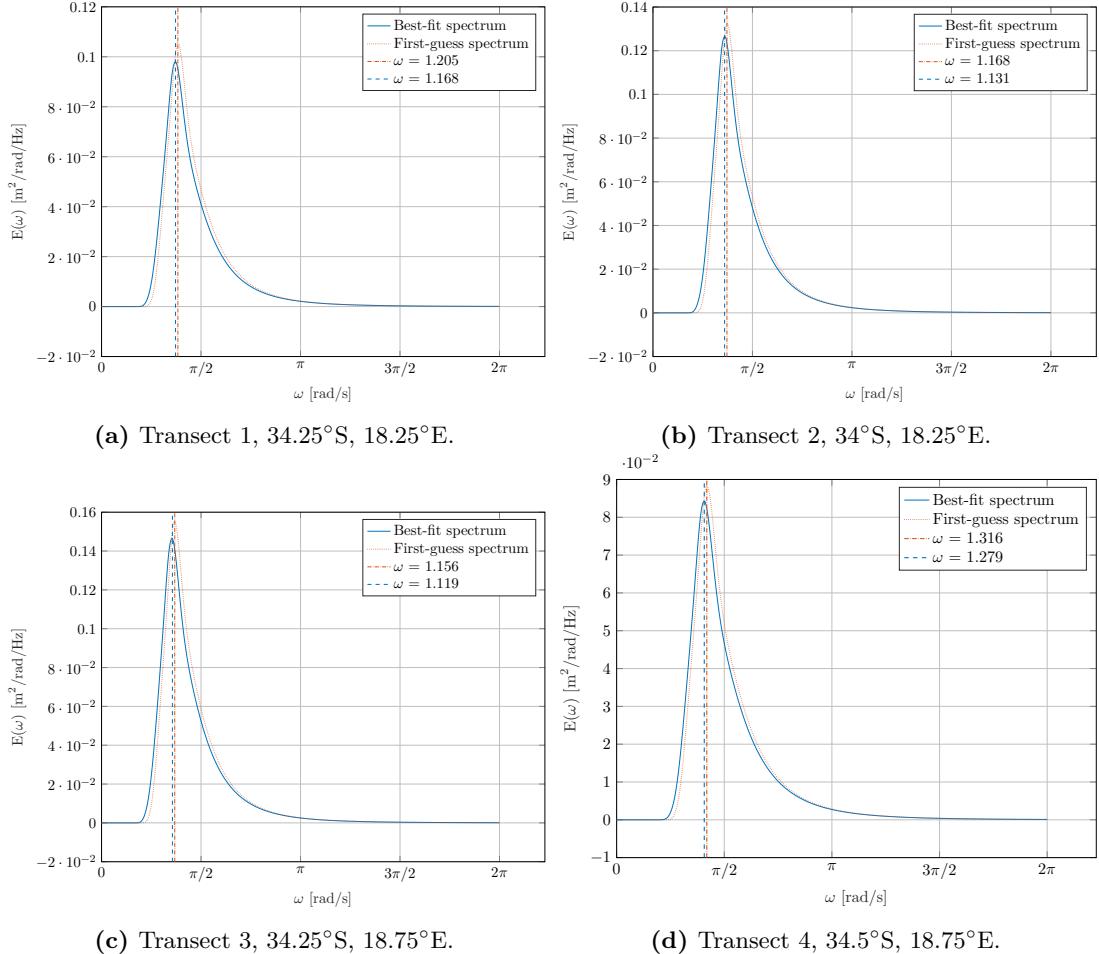


Figure C.31: Best-fit and first-guess wave spectra after 5 iterations for all transects. Zoomed out version plots of Figure 6.3.

Best-fit Frequency-direction Wave Spectra

The associated frequency-directional wave spectra for the best-fit one-dimensional wave spectra in Figure 6.3 are shown in Figure C.32.

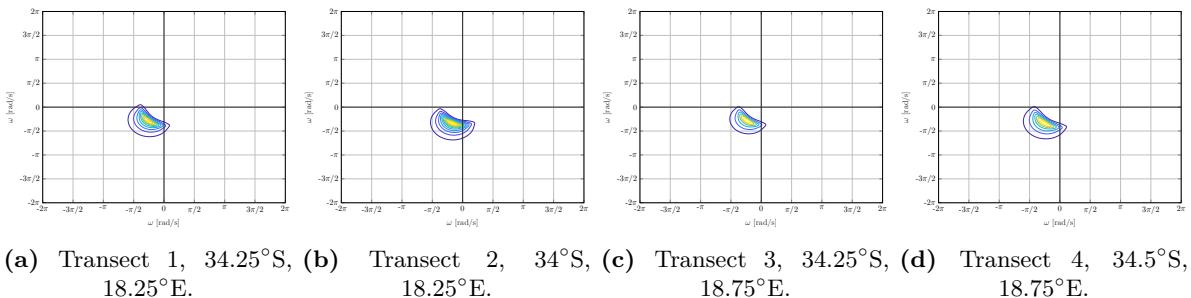
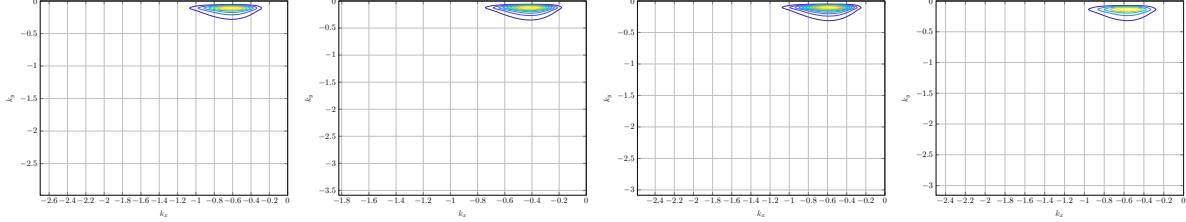


Figure C.32: Two-dimensional first-guess frequency-direction wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.

Best-fit Wave-number Wave Spectra

The associated frequency-directional wave spectra for the best-fit one-dimensional wave spectra in Figure 6.3 are shown in Figure C.33.



- (a) Transect 1, 34.25°S, 18.25°E. (b) Transect 2, 34°S, 18.25°E. (c) Transect 3, 34.25°S, 18.75°E. (d) Transect 4, 34.5°S, 18.75°E.

Figure C.33: Two-dimensional first-guess wave-number wave spectra for each individual testing location generated using wave data sourced from NCEP for Oct. 20, 2023.

Appendix D

Sentinel-1A Metadata Values

Table D.1 shows all tabulated attribute names that can be used with the `filterAttributesNetCDF` function. Additionally, Table D.1 is available as a .csv file on this project's [GitHub](#) here.

Attribute Name	Description	Unit
PRODUCT	Product name	
PRODUCT_TYPE	Product type	
SPH_DESCRIPTOR	Description	
MISSION	Satellite mission	
ACQUISITION_MODE	Acquisition mode	
antenna_pointing	Right or left facing	
BEAMS	Beams used	
SWATH	Swath name	
PROC_TIME	Processed time	utc
Processing_system_identifier	Processing system identified	
orbit_cycle	Cycle	
REL_ORBIT	Track	
ABS_ORBIT	Orbit	
STATE_VECTOR_TIME	Time of orbit state vector	utc
VECTOR_SOURCE	State vector source	
incidence_near		deg
incidence_far		deg
slice_num	Slice number	
data_take_id	Data take identifier	
first_line_time	First zero doppler azimuth time	utc
last_line_time	Last zero doppler azimuth time	utc
first_near_lat		deg
first_near_long		deg
first_far_lat		deg
first_far_long		deg
last_near_lat		deg
last_near_long		deg
last_far_lat		deg
last_far_long		deg
PASS	ASCENDING or DESCENDING	
SAMPLE_TYPE	DETECTED or COMPLEX	
mds1_tx_rx_polar	Polarization	
mds2_tx_rx_polar	Polarization	
mds3_tx_rx_polar	Polarization	

Table D.1 continued from previous page

Attribute Name	Description	Unit
mds4_tx_rx_polar	Polarization	
polsar_data	Polarimetric Matrix	flag
algorithm	Processing algorithm	
azimuth_looks		
range_looks		
range_spacing	Range sample spacing	m
azimuth_spacing	Azimuth sample spacing	m
range_window_type	Range window type	
range_window_coefficient	Range window coefficient	
pulse_repetition_frequency	PRF	Hz
radar_frequency	Radar frequency	MHz
line_time_interval		s
total_size	Total product size	MB
num_output_lines	Raster height	lines
num_samples_per_line	Raster width	samples
subset_offset_x	X coordinate of UL corner of subset in original image	samples
subset_offset_y	Y coordinate of UL corner of subset in original image	samples
srgr_flag	SRGR applied	flag
avg_scene_height	Average scene height ellipsoid	m
map_projection	Map projection applied	flag
is_terrain_corrected	Orthorectification applied	flag
DEM	Digital Elevation Model used	
geo_ref_system	Geographic reference system	
lat_pixel_res	Pixel resolution in geocoded image	deg
lon_pixel_res	Pixel resolution in geocoded image	deg
slant_range_to_first_pixel	Slant range to 1st data sample	m
ant_elev_corr_flag	Antenna elevation applied	flag
range_spread_comp_flag	Range spread compensation applied	flag
replica_power_corr_flag	Replica pulse power correction applied	flag
abs_calibration_flag	Product calibrated	flag
calibration_factor	Calibration constant	dB
chirp_power	Chirp power	
inc_angle_comp_flag	Incidence angle compensation applied	flag
ref_inc_angle	Reference incidence angle	
ref_slant_range	Reference slant range	
ref_slant_range_exp	Reference slant range exponent	
rescaling_factor	Rescaling factor	flag
bistatic_correction_applied		flag
range_sampling_rate	Range sampling rate	MHz
range_bandwidth	Bandwidth total in range	MHz
azimuth_bandwidth	Bandwidth total in azimuth	Hz
multilook_flag	Multilook applied	flag
coregistered_stack	Coregistration applied	flag
external_calibration_file	External calibration file used	
orbit_state_vector_file	Orbit file used	
metadata_version	AbsMetadata version	

Table D.1 continued from previous page

Attribute Name	Description	Unit
centre_lat		deg
centre_lon		deg
centre_heading		deg
centre_heading2		deg
Orbit_State_Vectors:orbit_vectorx:time	Time of x number orbit state vector	utc
Orbit_State_Vectors:orbit_vectorx:x ¹ _pos	x position of x number orbit state vector	m
Orbit_State_Vectors:orbit_vectorx:y_pos	y position of x number orbit state vector	m
Orbit_State_Vectors:orbit_vectorx:z_pos	z position of x number orbit state vector	m
Orbit_State_Vectors:orbit_vectorx:x_vel	x velocity of x number orbit state vector	m/s
Orbit_State_Vectors:orbit_vectorx:y_vel	y velocity of x number orbit state vector	m/s
Orbit_State_Vectors:orbit_vectorx:z_vel	z velocity of x number orbit state vector	m/s
Band_xx ² _yy ³ :swath	Swath name	
Band_xx_yy:polarization	Polarization	
Band_xx_yy:annotation	metadata file	
Band_xx_yy:band_names	corresponding bands	
Band_xx_yy:first_line_time	First zero doppler azimuth time	utc
Band_xx_yy:last_line_time	Last zero doppler azimuth time	utc
Band_xx_yy:line_time_interval	Time per line	s
Band_xx_yy:num_output_lines	Raster height	lines
Band_xx_yy:num_samples_per_line	Raster width	samples
Band_xx_yy:sample_type	DETECTED or COMPLEX	
Band_xx_yy:calibration_factor	Calibration constant	

Table D.1: S1A NetCDF exported metadata values. Attribute Names given can be used with filterAtt function.¹x here denotes the number of the orbit vector to request metadata for.²xx here denotes the capture mode of S1A data. Examples include IW or EW. More detail on S1A capture modes can be found in Section 3.2.2.³yy here denotes the polarisation of S1A data. Examples include VV or VH. More detail on S1A polarisation can be found in Section A.2.

Appendix E

MATLAB Live Script

This appendix contains a copy of the unrun `pipeline mlx` script which is classed as the pipeline for this project.

Wave Parameter Extraction Pipeline

This MATLAB LiveScript serves as a live document to show the implementation of the SAR wave parameter extraction pipeline with additional documentation. Documentation notes are included before code blocks which explain the code which follows.

Table of Contents

Prerequisites.....	1
Software Prerequisites.....	1
SNAP ESA.....	1
MATLAB Prerequisites.....	2
M_Map.....	2
Obtain SAR Data.....	2
Pre-Process SAR Data.....	2
Thermal Noise Calibration.....	2
Radiometric Calibration.....	3
Export Incidence Angle.....	3
Exporting as NetCDF.....	3
Take transects and subdivide.....	4
View SAR Data.....	4
Define Transect Parameters.....	4
Plot SAR data with transects shown.....	4
Show individual transects.....	5
Generate Wave Spectra.....	5
Download Associated Wave Data.....	5
Obtain wave first-guess wave parameters for the region of interest.....	6
Generate One-dimensional Wave Spectra.....	7
Generate Two-dimensional Wave Spectra.....	7
SAR Spectrum Calculation.....	8
Generate observed SAR Spectra	8
Hasselmann Procedure.....	9
Iterate the inversion procedure.....	9
Best-fit Data Plots.....	11
Significant Wave Height.....	11
Significant Wave Period.....	11
Two-dimensional wave spectra.....	12
Best-fit SAR Spectrum.....	12
One-dimensional Wave Spectum.....	12
Comparison between Best-fit and First-guess Wave Spectra.....	12
Cost Function, J.....	13
Output Parameters.....	13

Prerequisites

Software Prerequisites

SNAP ESA

SNAP is a tool developed by the European Space Agency (ESA) to process SAR data obtained from Sentinel Satellites. SNAP is used to pre-process data in this pipeline and is essential to install.

To install SNAP please do the following:

- Download SNAP [here](#) for your OS distribution
- Choose only the Sentinel Toolboxes installer
- Install SNAP and follow the onscreen instructions. It is only necessary to install the Sentinel-1 Toolbox

MATLAB Prerequisites

M_Map

M_Map is a mapping package for MATLAB which allows data plots on different world maps.

To use M_Map please do the following:

- Download the zipped package [here](#) and
- Add the extracted folder to your MATLAB path

Note: A copy of the M_Map package is included on the [git repo here](#).

Obtain SAR Data

SAR data from Sentinel-1A can be downloaded from [Copernicus SciHub](#). In order to download data, you need to register an account on the site.

Once you have registered your account, you can search for SAR data on the site by choosing the appropriate filters and region. The region you search is set by right-clicking and drawing a rectangle, otherwise you can left-click and draw the vertices of any polygon. Recommended filters are:

1. **Sensing Period:** As desired
2. **Satellite Platform:** S1A
3. **Product Type:** GRD
4. **Polarisation:** As desired

After searching with your filters, you will see all the available data within this region. Clicking on a red footprint on the map, shows a preview of these data, and the data can be downloaded from the pop-up.

Note: These data are in excess of 1GB, so large amounts of storage space is required. Use an external hard drive if possible.

Pre-Process SAR Data

After downloading SAR data from [Copernicus SciHub](#), open [SNAP](#) and import the data. These data can be viewed as follows:

1. Click the '+' next to the dataset
2. Click the '+' next to the *Bands* folder
3. Open the desired band to preview the image

Thermal Noise Calibration

This needs to be done first

To remove thermal noise, click *Radar -> Radiometric -> S1 Thermal Noise Removal*. This will bring up a window where the following needs to be done.

1. Ensure the file will save as *BEAM-DIMAP*
2. Check that *Remove Thermal Noise* is checked in the *Processing Parameters* window
3. Select a specific polarisation (If desired)
4. Rename the output file (*Target Product*) if desired
5. Check the *Open in SNAP* checkbox

After clicking *Run*, SNAP will begin removing thermal noise from the selected data and once complete will open in the *Product Explorer* sidebar.

Radiometric Calibration

To radiometrically calibrate the SAR data, ensure that the **noise calibrated product is selected**, then click *Radar -> Radiometric -> Calibrate* This will bring up a window where the following needs to be done.

1. Ensure the file will save as *BEAM-DIMAP*
2. Check that **only Output sigma0 band** is checked in the *Processing Parameters* window
3. Select a specific polarisation (If desired)
4. Rename the output file (*Target Product*) if desired
5. Check the *Open in SNAP* checkbox

After clicking *Run*, SNAP will begin radiometric calibration from the selected data and once complete will open in the *Product Explorer* sidebar.

Export Incidence Angle

In order to access the incidence angle of each pixel in the the SAR data, the following process needs to be followed.

1. Click the '+' next to the *Tie-Point Grids* folder
2. Right-click on the *incident_angle* band
3. Select *Band Maths*
4. Rename the band to "Incidence_Angle" and click OK

Exporting as NetCDF

In order to export these pre-processed SAR data in a format supported by MATLAB, the following process needs to be followed.

1. Click *File*
2. Hover over the *Export* option, and select **NetCDF4-BEAM**
3. If your file explorer is opened, click on *Subset...* on the right hand side of the dialog box
4. This will bring up a new window with four different menus.
5. You can take a Spatial Subset of the image using either pixel or geographical coordinates in the *Spatial Subset* menu

6. Ensure that under the *Band Subset* menu, your desired bands are selected, as well as your created ***Incidence_Angle*** band.
7. Ensure that under the *Metadata Subset* menu, all options are selected
8. After checking all of these parameters, click OK, and name the file as desired and save it in your desired location

Take transects and subdivide

After exporting the SAR data as a NetCDF file, use the following commands to import the data and metadata into MATLAB.

```
filepath = "C:\Users\ryanj\Downloads\150ct_subset_tnr_Cal.nc";
% Import data values
ncImport = ncinfo(filepath);
% Update the band to import as required
sarData = ncread(filepath, 'Sigma0_VV');
metadata = ncinfo(filepath, 'metadata');
incidenceAngleFull = ncread(filepath, 'Incidence_Angle');
latGrid = ncread(filepath, 'Lat');
lonGrid = ncread(filepath, 'Lon');
```

View SAR Data

Choose whether to display a greyscale image of the imported SAR data.

```
plotSARData = false;
if plotSARData
    figure;
    imshow(sarData);
end
```

Define Transect Parameters

Enter the number of desired transects to take, the start point (in pixels), along with the angle at which to take the transects. The angle is calculated from the positive x-axis clockwise.

```
topLeftLat = -34;
topLeftLon = 18.25;
[topLeftx, topLefty] = latToPixel(latGrid, lonGrid, topLeftLat, topLeftLon);
numTransects = 1;
th = 30;
[transectData, transectPositions] =
get512Transects(sarData, 0, topLefty, th, numTransects);
[incidenceAngle, incidencePositions] =
get512Transects(incidenceAngleFull, 1, 1, th, numTransects);
```

Plot SAR data with transects shown

The original data, with shown transects can be plotted.

```
plotFullScene = true;
if plotFullScene
    disp('Generating plot... ');
    figure;
    imshow(sarData)
    hold on;
    for i = 1:numTransects

        annotate512Transect(transectPositions(i,2),transectPositions(i,1),i,'red','white',0)
    ;
    end

    title('Sentinel-1A data with transects displayed')
    hold off
end
disp('Done');
```

Show individual transects

Choose which transects to plot. Enter the values as a comma seperated list.

```
transectsToPlot = "1";
strTransects = strsplit(transectsToPlot,',');
transectsToPlot = str2double(strTransects);
% Check if any element in the vector is greater than n
if any(transectsToPlot > numTransects)
    error('At least one number given is greater than the number of transects,
%.0f.\n', numTransects);
end

rowsToPlot = ceil(length(transectsToPlot)/2);

figure;
disp('Generating plot... ');
for i = 1:length(transectsToPlot)
    subplot(rowsToPlot,2,i);
    transectNum = transectsToPlot(i);
    imshow(transectData(:,:,transectNum));
    title(['Transect ', num2str(transectNum)])
end
disp('Done');
```

Generate Wave Spectra

Download Associated Wave Data

```
func = helperFunctions;
```

```

captureDate = func.getCaptureDate(metadata);
[noaaDateStr, noaaHourStr] = func.getNOAAParams(captureDate);
noaaUrl = func.getNOAAUrl(noaaDateStr, noaaHourStr);

```

Set the name for the saved downloaded .grib2 file.

```

waveDownloadName = "wave_data";
% Check file ends in .grib2

waveDataExtension = '.grib2';

if ~endsWith(waveDownloadName, waveDataExtension)
    waveDownloadName = strcat(waveDownloadName, waveDataExtension);
end
waveFilePath = downloadNOAAWaveFile(noaaUrl, waveDownloadName);

```

Set the location of wgrib and the downloaded .grib2 filepath. Choose if you'd like to plot your downloaded data along with the type to plot.

```

GribPath = "C:\Users\ryanj\Downloads\15_Oct_Wave_data.grib2";
wgrib2Path = "C:\Users\ryanj\Downloads\wgrib2.exe";
[GribPath,~,~] = fileparts(GribPath);
GribPath = strcat(GribPath, '\');
[wgrib2Path,~,~] = fileparts(wgrib2Path);
waveStruct = getGribStruct(wgrib2Path,GribPath);

```

```

noaaPlot = true;
noaaValToPlot = 'significantWaveHeight';
if noaaPlot
    disp('Generating plot...');
    noaaDataPlot('miller',waveStruct,noaaValToPlot);
    disp('Done');
end

```

Obtain wave first-guess wave parameters for the region of interest

Set the location at which to generate wave spectra for as well as the data resolution.

```

latitude = -34.204;
longitude = 18.28666944;
resolution = 0.25;
[gridLat, gridLon] = createLatLonGrid(latitude, longitude, resolution);
startLat = max(gridLat);
startLon = min(gridLon);
endLat = min(gridLat);
endLon = max(gridLon);
waveVals = getSubsetWaveVals(waveStruct,startLat,startLon,endLat,endLon);

```

Generate One-dimensional Wave Spectra

Define and instantiate variables for the one-dimensional wave spectra

```
imageSize = size(transectData,1);
%imageSize = 512;
w = linspace(0,2*pi,imageSize)';
f = linspace(0,1,imageSize)';
Hs = waveVals.significantWaveHeight(2,2);
T0 = waveVals.significantWavePeriod(2,2);
w0 = 2*pi./T0;
f0 = 1./T0;
gammaVal = 1.3713;
multipleWaveSpectra = false;
if multipleWaveSpectra
    S = generateMultipleJONSWAP(waveVals,gammaVal,w,1);
else
    S = generateSingleJONSWAP(Hs,w0,gammaVal,w);
end

plot1DWaveSpectra = true;
displaywPeak = false;
plotLats = false;
plotLons = false;
if plot1DWaveSpectra
    disp('Generating plot...');

waveSpectrumPlot(waveVals,S,w,multipleWaveSpectra,displaywPeak,plotLats,plotLons);
    disp('Done');
end
```

Generate Two-dimensional Wave Spectra

Define the ocean depth at the location of interest.

```
g = 9.81;
[D,theta] = generateDirectionalDistribution(waveVals,w,1);
E = generate2DWaveSpectrum(S,D);
if size(E,3) == 9
    E = E(:,:,:5,1);
else
    E = E(:,:,:,:,1);
end
d = 70;
%% Calculate E(k_x,k_y)
k = (w.^2./g)';
k_x = k.*cos(waveVals.direction(2,2));
k_y = k.*sin(waveVals.direction(2,2));
[E_k,k] = waveNumberSpectrum(E,w,k,d);
[E_k_inv,k_inv] = waveNumberSpectrum(E,w,fliplr(-k),d);
k_x_inv = k_inv.*cos(waveVals.direction(2,2));
```

```

k_y_inv = k_inv.*sin(waveVals.direction(2,2));
plot2DWaveSpectra = true;
wOrK = 1;
contourOrSurf =1;
if plot2DWaveSpectra
    if wOrK
        disp('Generating plot...');
        twoDWaveSpectrumPlot(E_k,wOrK,contourOrSurf,w,theta,k_x,k_y);
        %matlab2tikz(['..plots/contour_E_k.tex'])
        disp('Done');
    else
        disp('Generating plot...');
        twoDWaveSpectrumPlot(E,wOrK,contourOrSurf,w,theta,k_x,k_y);
        disp('Done');
    end
end

```

SAR Spectrum Calculation

Generate observed SAR Spectra

The SAR Spectrum of each transect can be plotted as well as the full scene spectrum after taking a 2D FFT of transectData. The same transects are used as defined in the previous section.

```

% Spectral bandwidth calculation
func = helperFunctions;
if multipleWaveSpectra
    S_1 = S(:,:,5);
else
    S_1 = S;
end
S_1 = S_1(2:end);
S_1 = func.resize(S_1(2:end),w);
for i=2:length(w)
    dw(i) = w(i)-w(i-1);
end
dw = mean(dw);
S_norm = S_1 / (trapz(S_1).*dw);
spectralMean = trapz(w.*S_norm).*dw;
spectralVar = trapz((w-spectralMean).^2.*S_norm).*dw;
spectralBW = sqrt(spectralVar);
plotTransectSARSpectrum = false;
transectsToPlot = "1";
strTransects = strsplit(transectsToPlot,',');
transectsToPlot = str2double(strTransects);
% Check if any element in the vector is greater than n
if any(transectsToPlot > numTransects)
    error('At least one number given is greater than the number of transects,
%.0f.\n', numTransects);
end

```

```

spectrumThreshold =35;
%57 has been good previously
if plotTransectSARSpectrum
    disp('Generating plot...');
    for i = 1:length(transectsToPlot)
        transectNum = transectsToPlot(i);
        intensityFFT = abs(fftshift(fft2(transectData(:,:,:transectNum)))); 
        SARSpectrumPlot(intensityFFT,spectrumThreshold,spectralBW,1);
        title(['Transect ', num2str(transectNum), ' observed SAR Spectrum'])
        %matlab2tikz(['../plots/results/observedSARSpectrum_',num2str(i),'.tex'])
    end
    disp('Done');
end

```

Hasselmann Procedure

The generated SAR spectrum, as derived by Hasselmann and Hasselmann, generates a SAR spectrum of the generated wave spectrum.

```

nonLinOrder = 1;
incidenceAngle = incidenceAngle(:,:,:1);
r = ones(size(incidenceAngle));
intensityFFT = abs(fftshift(fft2(transectData(:,:,:1)))); 
P_s_pipeline =
generateSARSpectrumOceanWaves(k,k_y,k_x,E_k,metadata,incidenceAngle,r,nonLinOrder,w)
;
plotGeneratedSARSpectrum = false;
if plotGeneratedSARSpectrum
    spectrumThresholdGen = 120;
    SARSpectrumPlot(20*log10(abs(P_s_pipeline)),spectrumThresholdGen,spectralBW,0);
    title('Generated SAR Spectrum')
    disp('Done');
    %matlab2tikz(['../plots/generatedSARSpectrum_Verify.tex'])
end

```

Iterate the inversion procedure

Define the number of iterations to minimise the cost function for before clicking "Run".

```

nonLinOrder = 1;
incidenceAngle = incidenceAngle(:,:,:1);
r = ones(size(incidenceAngle));
P_s_lin =
imageVarianceSpectrum(k,k_x,k_y,k_inv,k_x_inv,k_y_inv,E_k,E_k_inv,metadata,incidence
Angle);
E_k_inv = func.resize(E_k_inv(:,1:end-7),E_k);
[p_s_coeff,beta,xi_sqrt,~] = quasilinearCoeff(k,k_y,k_x,E_k,metadata,incidenceAngle);

```

```

p_s_coeff_inv =
quasilinearCoeff(k_inv,k_y_inv,k_x_inv,E_k_inv,metadata,incidenceAngle);
mu = calculateWeight(intensityFFT);
B = calculateNormalisationConstant(E_k);

E_k = func.resize(E_k(2:end,2:end),incidenceAngle);
E_k_inv = func.resize(E_k_inv(2:end,2:end),incidenceAngle);

look = func.getLook(metadata);
look = func.look(look);
polarisation = func.getPolarisation(metadata);
%k_y = func.resize(k_y,th(1,:));
k_l = func.kl(look,k_y);
k_l_inv = func.kl(look,k_y_inv);
omega = func.omega(k);
mu_Th = 0.5;
Th_k = func.hydroMTF(omega,mu_Th,k,k_y);
Th_k = func.resize(Th_k(2:end),omega);
Tt_k = func.tiltMTF(polarisation,k_l,incidenceAngle);
TR_k = func.rarMTF(Tt_k,Th_k);

Th_k_inv = func.hydroMTF(omega,mu_Th,k_inv,k_y_inv);
Th_k = func.resize(Th_k_inv(2:end),omega);
Tt_k_inv = func.tiltMTF(polarisation,k_l_inv,incidenceAngle);
TR_k_inv = func.rarMTF(Tt_k_inv,Th_k_inv);
TR_k_inv = func.resize(TR_k_inv(:,1:end-1),E_k);

Tv_k = func.rangeVelocityTF(omega,incidenceAngle,k_l,k);
Tv_k = func.resize(Tv_k(:,2:end),E_k);

Tvb_k = func.velocityBunchingMTF(beta,k_x,Tv_k);
Tvb_k = func.resize(Tvb_k(:,2:end),E_k);

Tv_k_inv = func.rangeVelocityTF(omega,incidenceAngle,k_l,fliplr(-k));
Tv_k_inv = func.resize(Tv_k_inv(:,2:end-3),E_k);

Tvb_k_inv = func.velocityBunchingMTF(beta,k_x_inv,Tv_k_inv);
Tvb_k_inv = func.resize(Tvb_k_inv(:,2:end-5),E_k);

Ts_k_inv = func.sarImagingMTF(TR_k_inv,Tvb_k_inv);
Ts_k = func.sarImagingMTF(TR_k,Tvb_k);

[M, N] = size(intensityFFT);
dx = spectralBW;
dy = spectralBW;
kx = (-M/2:M/2-1) / (M * dx);
ky = (-N/2:N/2-1) / (N * dy);
% Calculate Δk (the spacing between k values)
dk_x = kx(2) - kx(1);
dk_y = ky(2) - ky(1);

```

```

P_s_pipeline =
generateSARSpectrumOceanWaves(k,k_y,k_x,E_k,metadata,incidenceAngle,r,nonLinOrder,w)
;
iterations = 5;
iterationVec = linspace(1,iterations,iterations);
[J,En,Pn,sigWaveHeight,sigWavePeriod] =
inversion(iterations,P_s_pipeline,intensityFFT,E_k,E_k_inv,k,k_x,k_y,B,mu,Ts_k,Ts_k_
inv,P_s_lin,p_s_coeff,p_s_coeff_inv,dk_x,dk_y,metadata,incidenceAngle,r,w,theta);
[En_w, En_w_th] = waveNumberConvert(En,k,w,theta,D,d);

```

Best-fit Data Plots

All output best-fit plots can be toggled and displayed.

Significant Wave Height

```

% Plot waveHeight
plotBFHeight = false;
if plotBFHeight
    figure;
    plot(iterationVec,sigWaveHeight, 'DisplayName','Minimisation of cost function');
    yline(waveVals2.significantWaveHeight,'LineStyle','--','Color',
'r','DisplayName','First-guess wave spectrum');
    ylabel('Significant Wave Height [m]')
    xlabel('Iterations')
    grid on;
    grid minor;
    legend('show','Location','northwest')
    %matlab2tikz(['..../plots/results/150ct/
sigWaveHeightMinimisation2_5iterations_150ct.tex']);
end

```

Significant Wave Period

```

plotBFPeriod = false;
if plotBFPeriod
    figure;
    plot(iterationVec,sigWavePeriod, 'DisplayName','Minimisation of cost function');
    yline(waveVals2.significantWavePeriod,'LineStyle','--','Color',
'r','DisplayName','First-guess wave spectrum');
    ylabel('Significant Wave Period [s]')
    xlabel('Iterations')
    grid on;
    grid minor;
    legend('show','Location','northwest')
    %matlab2tikz(['..../plots/results/150ct/
sigWavePeriodMinimisation2_5iterations_150ct.tex']);
end

```

Two-dimensional wave spectra

```
plotMinimised2DWaveSpectrum = false;
if plotMinimised2DWaveSpectrum
    wOrK = 1;
    contourOrSurf = 1;
    if wOrK
        disp('Generating plot... ');
        twoDWaveSpectrumPlot(En,wOrK,contourOrSurf,w,theta,k_x,k_y);
        disp('Done');
    else
        disp('Generating plot... ');
        twoDWaveSpectrumPlot(En_w_th,wOrK,contourOrSurf,w,theta,k_x,k_y);
        disp('Done');
    end
end
```

Best-fit SAR Spectrum

```
plotMinimisedSARSpectrum = false;
if plotMinimisedSARSpectrum
    spectrumThresholdMin = 130;
    disp('Generating plot... ')
    SARSpectrumPlot(20*log10(abs(Pn)),spectrumThresholdMin,spectralBW);
    title('Best-fit SAR Spectrum')
    disp('Done');
end
```

One-dimensional Wave Specturm

```
plotEnw = false;
if plotEnw
    figure;
    plot(w,En_w);
    grid on;
    xlabel('\omega [rad/s]')
    ylabel('E(\omega) [m^2/rad/Hz]')
    grid minor;
    set(gca,'XTick',0:pi/2:2*pi)
    set(gca,'XTickLabel',{'0','\pi/2','\pi','3\pi/2','2\pi'})
    set(gca,'FontSize',12)
end
```

Comparison between Best-fit and First-guess Wave Spectra

```
plotBFandFGwave = false;
BFandFGwPeak = false;
if plotBFandFGwave
    figure;
    plot(w,En_w,'DisplayName','Best-fit spectrum');
```

```

hold on;
plot(w,S, 'DisplayName', 'First-guess spectrum');
maxIndexS = find(S == max(S));
maxIndexEn = find(En_w == max(En_w));
title(['One-dimensional wave spectrum, E(\omega) at ',
num2str(waveVals.latitude(1,1)), 'S, ', num2str(waveVals.longitude(1,1)), 'E'])
xlabel('omega [rad/s]')
ylabel('E(omega) [m^2/rad/Hz]')
grid on;
if BFandFGwPeak
    xline(w(maxIndexS),LineWidth=1,DisplayName=['\omega =
',num2str(round(w(maxIndexS),3))],Color="#D95319",LineStyle="--");
    xline(w(maxIndexEn),LineWidth=1,DisplayName=['\omega =
',num2str(round(w(maxIndexEn),3))],Color="#0072BD",LineStyle="--");
end
hold off;
legend('show');
xlabel('omega [rad/s]')
ylabel('E(omega) [m^2/rad/Hz]')
grid on;
grid minor;
set(gca,'XTick',0:pi/2:2*pi)
set(gca,'XTickLabel',{'0','\pi/2','\pi','3\pi/2','2\pi'})
set(gca,'FontSize',12)
%matlab2tikz(['../plots/results/150ct/oneDWaveSpec2_5iterations_150ct.tex']);
end

```

Cost Function, J

```

plotJ = false;
if plotJ
    figure;
    plot(iterationVec,J)
    xlabel('Iterations')
    ylabel('J')
    grid on;
    grid minor;
    %matlab2tikz(['../plots/results/150ct/J2_5iterations_150ct.tex']);
end

```

Output Parameters

The best-fit significant wave height and period can be saved as .csv files. These variables are also available in the Workspace.

```

saveWaveParams = false;
writematrix(sigWaveHeight, 'SignificantWaveHeight.csv');
writematrix(sigWavePeriod, 'SignificantWavePeriod.csv');

```

If the pipeline has not been run yet, click "Run".

Bibliography

- [1] M. Scott, “Understanding climate: Antarctic sea ice extent,” Mar. 2023, Accessed on Aug. 4, 2023. [Online]. Available: <https://www.climate.gov/news-features/understanding-climate/understanding-climate-antarctic-sea-ice-extent>
- [2] R. Torres, P. Snoeij, D. Geudtner, D. Bibby, M. Davidson, E. Attema *et al.*, “GMES Sentinel-1 mission,” *Remote Sensing of Environment*, vol. 120, pp. 9–24, 2012.
- [3] I. Garkusha and V. Hnatushenko, “A technology for building a seamless mosaic coverage of the Antarctica coasts with various-time images from Sentinel-1,” in *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*. IEEE, 2020, pp. 747–750.
- [4] L. H. Holthuijsen, *Waves in Oceanic and Coastal Waters*. Cambridge University Press, 2007.
- [5] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, “A tutorial on synthetic aperture radar,” *IEEE Geoscience and remote sensing magazine*, vol. 1, no. 1, pp. 6–43, 2013.
- [6] F. Meyer, *Spaceborne Synthetic Aperture Radar - Principles, Data Access, and Basic Processing Techniques*, 1st ed. SERVIT, 2019, vol. 1, p. 21–63.
- [7] G. Spreen and S. Kern, “Chapter 9: Methods of satellite remote sensing of sea ice,” in *Sea Ice*, D. N. Thomas, Ed. John Wiley & Sons, 2017, pp. 239–260.
- [8] M. Bourbigot, “Sentinel-1 product defintion,” Mar 2016. [Online]. Available: <https://sentinels.copernicus.eu/documents/247904/1877131/Sentinel-1-Product-Definition.pdf/6049ee42-6dc7-4e76-9886-f7a72f5631f3?t=1461673251000>
- [9] R. A. Massom and S. E. Stammerjohn, “Antarctic sea ice change and variability - physical and ecological implications,” *Polar Science*, vol. 4, pp. 149–186, 2010.
- [10] M. C. Kennicutt, D. Bromwich, D. Liggett, B. Njåstad, L. Peck, and S. R. Rintoul, “Sustained Antarctic research: A 21st century imperative,” *One Earth*, vol. 1, pp. 95–113, 2019.
- [11] I. Simmonds, “Comparing and contrasting the behaviour of Arctic and Antarctic sea ice over the 35 year period 1979-2013,” *Annals of Glaciology*, vol. 56, pp. 18–28, 2015.
- [12] T. Maksym, S. E. Stammerjohn, S. Ackley, and R. Massom, “Antarctic sea ice- A polar opposite?” *Oceanography*, vol. 25, pp. 140–151, 2012.
- [13] A. Alberello, M. Onorato, L. Bennetts, M. Vichi, C. Eayrs, K. MacHutchon *et al.*, “Brief communication: Pancake ice floe size distribution during the winter expansion of the Antarctic marginal ice zone,” *The Cryosphere*, vol. 13, pp. 41–48, 2019.
- [14] M. J. Doble, G. De Carolis, M. H. Meylan, J. R. Bidlot, and P. Wadhams, “Relating wave attenuation to pancake ice thickness, using field measurements and model results,” *Geophysical Research Letters*, vol. 42, pp. 4473–4481, 2015.

- [15] B. Bronselaer, M. Winton, S. M. Griffies, W. J. Hurlin, K. B. Rodgers, O. V. Sergienko *et al.*, “Change in future climate due to Antarctic meltwater,” *Nature*, vol. 564, p. 53–58, 2018.
- [16] K. R. Gamache, J. R. Giardino, N. R. Regmi, and J. D. Vitek, “Chapter 12 - The Impact of Glacial Geomorphology on Critical Zone Processes,” in *Developments in Earth Surface Processes*, J. R. Giardino and C. Houser, Eds. Elsevier, 2015, pp. 363–395.
- [17] W. M. J. Lazeroms, A. Jenkins, G. H. Gudmundsson, and R. S. W. van de Wal, “Modelling present-day basal melt rates for Antarctic ice shelves using a parametrization of buoyant meltwater plumes,” *The Cryosphere*, vol. 12, pp. 49–70, 2018.
- [18] Y. Dong, A. Varquez, and M. Kanda, “Global anthropogenic heat flux database with high spatial resolution,” *Atmospheric Environment*, vol. 150, pp. 276–294, 2017.
- [19] M. P. Meredith and M. A. Brandon, “Chapter 8: Oceanography and sea ice in the Southern Ocean,” in *Sea Ice*, D. N. Thomas, Ed. John Wiley & Sons, 2017, p. 216–238.
- [20] J. C. Comiso, R. A. Gersten, L. V. Stock, J. Turner, G. J. Perez, and K. Cho, “Positive trend in the Antarctic sea ice cover and associated changes in surface temperature,” *Journal of Climate*, vol. 30, pp. 2251–2267, 2017.
- [21] K. E. Taylor, R. J. Stouffer, and G. A. Meehl, “An overview of CMIP5 and the experiment design,” *Bulletin of the American Meteorological Society*, vol. 93, pp. 485 – 498, 2012.
- [22] C. L. Parkinson, “Southern Ocean sea ice and its wider linkages: Insights revealed from models and observations,” *Antarctic Science*, vol. 16, pp. 387–400, 12 2004.
- [23] M. Gherardi and M. C. Lagomarsino, “Characterizing the size and shape of sea ice floes,” *Scientific Reports*, vol. 5, 5 2015.
- [24] P. Wadhams, “The Seasonal Ice Zone,” *The Geophysics of Sea Ice*, p. 825–991, 1986.
- [25] J. Brouwer, A. D. Fraser, D. J. Murphy, P. Wongpan, A. Alberello, A. Kohout *et al.*, “Altimetric observation of wave attenuation through the Antarctic marginal ice zone using ICESat-2,” *Cryosphere*, vol. 16, pp. 2325–2353, 2022.
- [26] M. Vichi, C. Eayrs, A. Alberello, A. Bekker, L. Bennetts, D. Holland *et al.*, “Effects of an explosive polar cyclone crossing the Antarctic Marginal Ice Zone,” *Geophysical Research Letters*, vol. 46, p. 5948–5958, 2019.
- [27] A. Gordon, “Bottom water formation,” *Encyclopedia of Ocean Sciences*, p. 334–340, 2001.
- [28] S. S. Jacobs, “Bottom water production and its links with the thermohaline circulation,” *Antarctic Science*, vol. 16, p. 427–437, 2004.
- [29] V. A. Squire, “Ocean wave interactions with sea ice: A reappraisal,” *Annual Review of Fluid Mechanics*, vol. 52, pp. 37–60, 2020.
- [30] F. Montiel, A. L. Kohout, and L. A. Roach, “Physical drivers of ocean wave attenuation in the marginal ice zone,” *Journal of Physical Oceanography*, vol. 52, pp. 889 – 906, 2022.
- [31] S. Stammerjohn, R. Massom, D. Rind, and D. Martinson, “Regions of rapid sea ice change: An inter-hemispheric seasonal comparison,” *Geophysical Research Letters*, vol. 39, 2012.
- [32] M. G. Asplin, R. Galley, D. G. Barber, and S. Prinsenberg, “Fracture of summer perennial sea ice by ocean swell as a result of Arctic storms,” *Journal of Geophysical Research: Oceans*, vol. 117, p. C06025, 2012.

- [33] J. E. Stopa, F. Ardhuin, J. Thomson, M. M. Smith, A. Kohout, M. J. Doble *et al.*, "Wave attenuation through an Arctic marginal ice zone on 12 October 2015: 1. Measurement of wave spectra and ice features from Sentinel 1a," *Journal of Geophysical Research: Oceans*, vol. 123, pp. 3619–3634, 2018.
- [34] I. R. Young, E. Fontaine, Q. Liu, and A. V. Babanin, "The wave climate of the Southern Ocean," *Journal of Physical Oceanography*, vol. 50, pp. 1417 – 1433, 2020.
- [35] A. Alberello, L. G. Bennetts, M. Onorato, M. Vichi, K. MacHutchon, C. Eayrs *et al.*, "Three-dimensional imaging of waves and floes in the marginal ice zone during a cyclone," *Nature Communications*, vol. 13, 2022.
- [36] W. M. Organization, "Sea ice nomenclature," World Meteorological Organization, Tech. Rep., March 2014, Accessed Aug. 6, 2023. [Online]. Available: https://library.wmo.int/doc_num.php?explnum_id=4651
- [37] M. Vichi, "A statistical definition of the Antarctic marginal ice zone," *The Cryosphere*, pp. 1–23, 2021, Preprint.
- [38] M. J. Doble, M. D. Coon, and P. Wadhams, "Pancake ice formation in the Weddell Sea," *Journal of Geophysical Research: Oceans*, vol. 108, pp. 3209–3221, 2003.
- [39] A. Alberello, A. Dolatshah, L. G. Bennetts, M. Onorato, F. Nelli, and A. Toffoli, "A Physical Model of Wave Attenuation in Pancake Ice," *International Journal of Offshore and Polar Engineering*, vol. 31, pp. 263–269, 2021.
- [40] P. Wadhams and J. Wilkinson, "The physical properties of sea ice in the Odden ice tongue," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 46, no. 6-7, pp. 1275–1300, 1999.
- [41] W. Dierking, "Sea ice monitoring by Synthetic Aperture Radar," *Oceanography*, vol. 26, pp. 100–111, 2013.
- [42] G. Aulicino, G. Fusco, S. Kern, and G. Budillon, "Estimation of sea-ice thickness in Ross and Weddell Seas from SSM/I Brightness Temperatures," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, pp. 4122–4140, 2014.
- [43] P. Wadhams, M. A. Lange, and S. F. Ackley, "The ice thickness distribution across the Atlantic sector of the Antarctic Ocean in midwinter," *Journal of Geophysical Research: Oceans*, vol. 92, pp. 14 535–14 552, 1987.
- [44] D. B. Clarke and S. F. Ackley, "Sea ice structure and biological activity in the Antarctic marginal ice zone," *Journal of Geophysical Research: Oceans*, vol. 89, pp. 2087–2095, 1984.
- [45] M. A. Lange and H. Eicken, "Textural characteristics of sea ice and the major mechanisms of ice growth in the weddell sea," *Annals of Glaciology*, vol. 15, p. 210–215, 1991.
- [46] G. De Carolis and D. Desiderio, "Dispersion and attenuation of gravity waves in ice: a two-layer viscous fluid model with experimental data validation," *Physics Letters A*, vol. 305, pp. 399–412, 2002.
- [47] P. Wadhams, F. Parmiggiani, and G. De Carolis, "The use of SAR to measure ocean wave dispersion in frazil-pancake icefields," *Journal of physical oceanography*, vol. 32, pp. 1721–1746, 2002.
- [48] M. Belmonte Rivas, I. Otosaka, A. Stoffelen, and A. Verhoef, "A scatterometer record of sea ice extents and backscatter: 1992–2016," *The Cryosphere*, vol. 12, pp. 2941–2953, 2018.

- [49] W. Dierking, "Radar signatures of frazil and pancake ice-a review," *Report, Department of Electromagnetic Systems, Technical University of Denmark, No. 695*, 2001.
- [50] R. Ricker, S. Hendricks, D. K. Perovich, V. Helm, and R. Gerdes, "Impact of snow accumulation on CryoSat-2 range retrievals over Arctic sea ice: An observational approach with buoy data," *Geophysical Research Letters*, vol. 42, pp. 4447–4455, 2015.
- [51] W. Emery, C. Fowler, and J. Maslanik, "Arctic sea ice concentrations from special sensor microwave imager and advanced very high resolution radiometer satellite data," *Journal of Geophysical Research: Oceans*, vol. 99, pp. 18 329–18 342, 1994.
- [52] R. E. Brandt, S. G. Warren, A. P. Worby, and T. C. Grenfell, "Surface albedo of the Antarctic sea ice zone," *Journal of Climate*, vol. 18, pp. 3606–3622, 2005.
- [53] NASA, "Types of orbits," Accessed on Aug. 6, 2023. [Online]. Available: <https://solarsystem.nasa.gov/basics/chapter5-1/>
- [54] E. Attema, M. Davidson, P. Snoeij, B. Rommen, and N. Flouri, "Sentinel-1 mission overview," in *2009 IEEE International Geoscience and Remote Sensing Symposium*, vol. 1. IEEE, 2009, pp. 1–36.
- [55] D. Geudtner, R. Torres, P. Snoeij, M. Davidson, and B. Rommen, "Sentinel-1 system capabilities and applications," in *2014 IEEE Geoscience and Remote Sensing Symposium*. IEEE, 2014, pp. 1457–1460.
- [56] P. Potin, B. Rosich, N. Miranda, P. Grimont, I. Shurmer, A. O'Connell *et al.*, "Copernicus Sentinel-1 constellation mission operations status," in *IGARSS 2019-2019 IEEE international geoscience and remote sensing symposium*. IEEE, 2019, pp. 5385–5388.
- [57] G. C. Hulley, D. Ghent, F. M. Götsche, P. C. Guillevic, D. J. Mildrexler, and C. Coll, "3 - Land Surface Temperature," in *Taking the Temperature of the Earth*, G. C. Hulley and D. Ghent, Eds. Elsevier, 2019, pp. 57–127.
- [58] R. Torres, P. Snoeij, M. Davidson, D. Bibby, and S. Lokas, "The Sentinel-1 mission and its application capabilities," in *2012 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2012, pp. 1703–1706.
- [59] K. Hasselmann and S. Hasselmann, "On the nonlinear mapping of an ocean wave spectrum into a synthetic aperture radar image spectrum and its inversion," *Journal of Geophysical Research*, vol. 96, pp. 10 713–10 729, 1991.
- [60] S. Hasselmann, C. Brüning, K. Hasselmann, and P. Heimbach, "An improved algorithm for the retrieval of ocean wave spectra from synthetic aperture radar image spectra," *Journal of Geophysical Research: Oceans*, vol. 101, pp. 16 615–16 629, 1996.
- [61] A. Fernandes, Y. Sarma, H. Menon, and P. Vethamony, "Wave directional spectrum from SAR imagery," *Journal of the Indian Society of Remote Sensing*, vol. 28, pp. 9–18, 2000.
- [62] C. Mastenbroek and C. F. De Valk, "A semiparametric algorithm to retrieve ocean wave spectra from synthetic aperture radar," *Journal of Geophysical Research: Oceans*, vol. 105, pp. 3497–3516, 2000.
- [63] G. Engen and H. Johnsen, "SAR-ocean wave inversion using image cross spectra," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, pp. 1047–1056, 1995.
- [64] G. De Carolis, P. Olla, and F. De Santi, "SAR image wave spectra to retrieve the thickness of grease-pancake sea ice using viscous wave propagation models," *Scientific Reports*, vol. 11, 2021.

- [65] F. De Santi, G. De Carolis, P. Olla, M. Doble, S. Cheng, H. H. Shen *et al.*, “On the ocean wave attenuation rate in grease-pancake ice, a comparison of viscous layer propagation models with field data,” *Journal of Geophysical Research: Oceans*, vol. 123, pp. 5933–5948, 2018.
- [66] J. B. Keller, “Gravity waves on ice-covered water,” *Journal of Geophysical Research: Oceans*, vol. 103, pp. 7663–7669, 1998.
- [67] G. Aulicino, P. Wadhams, and F. Parmiggiani, “SAR pancake ice thickness retrieval in the Terra Nova Bay (Antarctica) during the PIPERS expedition in winter 2017,” *Remote Sensing*, vol. 11, 2019.
- [68] P. Wadhams, F. F. Parmiggiani, G. de Carolis, D. Desiderio, and M. J. Doble, “SAR imaging of wave dispersion in Antarctic pancake ice and its use in measuring ice thickness,” *Geophysical Research Letters*, vol. 31, 2004.
- [69] W. J. Pierson Jr and L. Moskowitz, “A proposed spectral form for fully developed wind seas based on the similarity theory of SA Kitaigorodskii,” *Journal of Geophysical Research*, vol. 69, no. 24, pp. 5181–5190, 1964.
- [70] K. Hasselmann, T. P. Barnett, E. Bouws, H. Carlson, D. E. Cartwright, K. Enke *et al.*, “Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP).” *Ergaenzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A*, 1973.
- [71] K. Hasselmann, “On the spectral dissipation of ocean waves due to white capping,” *Boundary-Layer Meteorology*, vol. 6, pp. 107–127, 1974.
- [72] “Ieee standard letter designations for radar-frequency bands,” *IEEE Std 521-2019 (Revision of IEEE Std 521-2002)*, pp. 1–15, 2020.
- [73] I. G. Cumming and F. H. Wong, “Digital processing of synthetic aperture radar data,” *Artech house*, vol. 1, no. 3, pp. 108–110, 2005.
- [74] A. C. Frery, J. Wu, and L. Gomez, *Data Acquisition*. IEEE Press, 2022, pp. 1–41.
- [75] P. Huang and W. Xu, “A new spaceborne burst Synthetic Aperture Radar imaging mode for Wide Swath Coverage,” *Remote Sensing*, vol. 6, no. 1, pp. 801–814, 2014. [Online]. Available: <https://www.mdpi.com/2072-4292/6/1/801>
- [76] F. De Zan and A. Monti Guarnieri, “Topsar: Terrain observation by progressive scans,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 9, pp. 2352–2360, 2006.
- [77] “Deciaml degrees,” Feb 2023, Accessed on Sept. 18, 2023. [Online]. Available: https://en.wikipedia.org/wiki/Decimal_degrees
- [78] B. Gweba, “Developing a methodology for the assessment of wave energy along the South African Coast,” Master’s thesis, University of Stellenbosch, Stellenbosch, South Africa, March 2018, Available: <https://scholar.sun.ac.za/handle/10019.1/103308>.
- [79] M. Portabella, A. Stoffelen, and J. A. Johannessen, “Toward an optimal inversion method for synthetic aperture radar wind retrieval,” *Journal of Geophysical Research: Oceans*, vol. 107, 2002.
- [80] F. D. Carsey, *Microwave remote sensing of sea ice*. American Geophysical Union, 1992.