

SIMULATION OF BRIGHTNESS TEMPERATURES FOR THE MICROWAVE
RADIOMETER ON THE AQUARIUS/SAC-D MISSION

by

SALMAN SAEED KHAN

B.S. National University of Computer & Emerging Sciences, 2004

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the School of Electrical Engineering & Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2009

ABSTRACT

Microwave radiometers are highly sensitive receivers capable of measuring low levels of natural blackbody microwave emissions. Remote sensing by satellite microwave radiometers flying on low-earth, polar orbiting, satellites can infer a variety of terrestrial and atmospheric geophysical parameters for scientific and operational applications, such as weather and climate prediction.

The objective of this thesis is to provide realistic simulated ocean brightness temperatures for the 3-channel Microwave Radiometer (MWR), which will be launched in May 2010 on the joint NASA/CONAE Aquarius/SAC-D Mission. These data will be used for pre-launch geophysical retrieval algorithms development and validation testing. Analyses are performed to evaluate the proposed MWR measurement geometry and verify the requirements for spatial/temporal sampling. Finally, a preliminary study is performed for the post-launch inter-satellite radiometric calibration using the WindSat polarimetric radiometer on the Coriolis satellite.

Dedicated to my beloved father,
Professor Ahmad Saeed Khan (1935-2009).

The setting sun sinks at the horizon,
Its weakening glow is not meant to die,
It is merely an end to a day well spent,
It is merely an end to a life well lived,
For some it is the start of a lonesome night,
It is only true 'cause they can't see the light,
Alas my friends, if only you could comprehend,
Every sunset is a sunrise at the end.

Yours as always,

Salman.

ACKNOWLEDGMENTS

I would like to thank Dr. W. Linwood Jones for offering me an opportunity to work in the Central Florida Remote Sensing Lab (CFRSL) under his guidance. It has improved my skills in a multitude of areas. The CFRSL family has also been very supportive and cooperative; they have made me feel at home since I joined in October 2008. I would like to thank all the lab members for their support, but specifically, I would like to mention that my colleague and now a good friend of mine, Mr. Spencer Farrar, has put a lot of energy in helping me with my thesis and I can safely say that without his assistance it would have been a very difficult task; I thank you for your help. I would also like to thank my committee members for their guidance, specifically, Sergio for his valuable input during his spring visit to CFRSL and Mr. Johnson for his time and the kind words he had for me when my father passed away. All my friends in Orlando, specifically, Azer, Nazim, Sami, Kamran & Ali, I will never forget your support when I was facing the toughest time of my life with the sad demise of my beloved father.

Finally, I would express my love to my precious mother; I love you more than the world and it is only because of your love and prayers that I have been achieving whatever I aim for, and my brother who has shown that he is a strong man, just like dad was; I thank you for your support and I love my niece, nephew, and my sister Kishu.

I would also wish to acknowledge that the CFRSL is a university partner with Analytical Graphics Inc. and that the use of Satellite Tool Kit in this thesis was for educational purposes only.

TABLE OF CONTENTS

LIST OF FIGURES	vi
LIST OF TABLES	viii
LIST OF ACRONYMS/ABBREVIATIONS	ix
CHAPTER 1.INTRODUCTION	1
CHAPTER 2. SATELLITES AND INSTRUMENTS	5
2.1 Aquarius/SAC-D.....	5
2.1.1 Aquarius Instrument.....	8
2.1.2 Microwave Radiometer (MWR)	11
2.1.3 Yaw Steering.....	17
2.2 Coriolis/WindSat.....	19
2.3 Inter-Satellite Relative Orbital Periods.....	25
2.4 Inter-Satellite Radiometric Spatial/Temporal Collocations.....	28
CHAPTER 3. SIMULATED MICROWAVE RADIOMETER DATASET	30
3.1 MWR Tb Simulation Requirements	31
3.2 WindSat IDR.....	38
3.3 MWR Footprint Analysis.....	44
3.4 Normalization of WindSat Brightness Temperatures to MWR Incidence Angles	51
3.4.1 Adjustments for Incidence Angle	52
3.4.2 Dependence on Environmental Parameters	64
CHAPTER 4. RESULTS AND VALIDATION	77
4.1 Spatial/Temporal Validation.....	77
4.2 Radiometric Validation.....	81
4.3 Yaw Steering Results	83
CHAPTER 5. CONCLUSIONS & FUTURE WORK	99
APPENDIX A CONICAL SCANNING GEOMETRY	103
APPENDIX B RADIATIVE TRANSFER THEORY	108
APPENDIX C MATLAB SCRIPT.....	113
LIST OF REFERENCES	167

LIST OF FIGURES

Figure 1: Annual ocean mean salinity with colorbar in practical salinity units (PSS) [4].....	7
Figure 2: Aquarius/SAC-D Satellite Mission [5].....	8
Figure 3: Aquarius Instrument Configuration [9].....	9
Figure 4: Aquarius Beam Pointing [6].....	10
Figure 5: MWR Instrument Configuration [5].....	12
Figure 6: MWR Forward Beam (Ka-Band) Geometry [10].	13
Figure 7: MWR Beam Geometry & Swath Overlap [11].	14
Figure 8: Target Footprints of Forward Looking (Ka-Band) Beams [10].	15
Figure 9: MWR Swath Coverage per Beam [10].....	16
Figure 10: Timing Diagram [10].....	17
Figure 11: Aquarius/SAC-D yaw steering (left panel) and without yaw steering (right panel). ..	18
Figure 12: WindSat Mechanical Configuration [12].	20
Figure 13: WindSat Feed-bench with multi-frequency/polarization horns [2].....	21
Figure 14: WindSat Earth Projected Beams for multiple antenna scans [2].....	22
Figure 15: WindSat Geometry & Pointing Angles [2].	23
Figure 16: WindSat Conical Scanning geometry.....	24
Figure 17: WindSat Swath Angular Distributions for different frequencies [12].	25
Figure 18: Aquarius & Coriolis Ground-Tracks in-Phase.	27
Figure 19: Aquarius & Coriolis Ground-Tracks out of Phase.	27
Figure 20: Example of Ascending/Descending Collocations for MWR and WindSat.....	28
Figure 21: Collocation swath in 45 hours.	29
Figure 22: Azimuthal Translation of MWR to Equivalent WindSat.	32
Figure 23: MWR Swath [10].	33
Figure 24: WindSat K/Ka Band Scan Geometry [10].	35
Figure 25: EDR Azimuth Limits.....	35
Figure 26: Nineteen Simulated MWR Beams.	37
Figure 27: Conical Scanning IDR Pixels at 37 GHz for two antenna scans.....	39
Figure 28: Conical Scanning IDR Pixels at 23.8 GHz for two antenna scans.....	40
Figure 29: IDR Pixels at 37 GHz & 23.8 GHz.	41
Figure 30: Sensor Instantaneous Field of View (IFOV) [17].	44
Figure 31: Elliptical & Rectangular MWR IFOVs.....	45
Figure 32: IDR Spatial Resolution at 23.8 GHz.	47
Figure 33: IDR Spatial Resolution at 37 GHz.	47
Figure 34: IDR beam-fills in Rectangular MWR IFOVs.....	48
Figure 35: IDR Scan Skip.....	49
Figure 36: Flowchart of MWR Brightness Temperature Simulation Algorithm.....	50
Figure 37: Density plot, binned averages & linear fit at 23.8 GHz V pol (52°).	53
Figure 38: Comparison of Linear Fit to the 45° Line at 23.8 GHz V pol (52°).	54
Figure 39: Density plot, binned averages & linear fit at 23.8 GHz V pol (58°).	55
Figure 40: Comparison of Linear Fit to the 45° Line at 23.8 GHz V pol (58°).	56
Figure 41: Density plot, binned averages & linear fit at 37 GHz V pol (52°).	57
Figure 42: Comparison of Linear Fit to the 45° Line at 37 GHz V pol (52°).	58

Figure 43: Density plot, binned averages& linear fit at 37 GHz V pol (58°).	59
Figure 44: Comparison of Linear Fit to the 45° Line at 37 GHz V pol (58°).	60
Figure 45: Density plot, binned averages & linear fit at 37 GHz H pol (52°)	61
Figure 46: Comparison of Linear Fit to the 45° Line at 37 GHz H pol (52°).	62
Figure 47: Density plot, binned averages & linear fit at 37 GHz H pol (58°).	63
Figure 48: Comparison of Linear Fit to the 45° Line at 37 GHz H pol (58°).	64
Figure 49: Dependence on wind speed, bias density plot & binned averages at 23.8 GHz V.....	65
Figure 50: Dependence on water vapor, bias density plot & binned averages at 23.8 GHz V.....	66
Figure 51: Dependence on SST, bias density plot & binned averages at 23.8 GHz V.....	67
Figure 52: Dependence on CLW, bias density plot & binned averages at 23.8 GHz V.....	68
Figure 53: Dependence on wind speed, bias density plot & binned averages at 37 GHz V.....	69
Figure 54: Dependence on water vapor, bias density plot & binned averages at 37 GHz V.....	70
Figure 55: Dependence on SST, bias density plot & binned averages at 37 GHz V.....	71
Figure 56: Dependence on CLW, bias density plot & binned averages at 37 GHz V.....	72
Figure 57: Dependence on wind speed, bias density plot & binned averages at 37 GHz H.....	73
Figure 58: Dependence on water vapor, bias density plot & binned averages at 37 GHz H.....	74
Figure 59: Dependence on SST, bias density plot & binned averages at 37 GHz H.....	75
Figure 60: Dependence on CLW, bias density plot& binned averages at 37 GHz H.....	76
Figure 61: IDR Beam Fills in MWR Pixels at 36.5 GHz.	78
Figure 62: IDR Beam Fills in MWR Pixels at 23.8 GHz.	79
Figure 63: Collocated MWR 36.5 (Red) and 23.8 GHz (Green) Scans.	80
Figure 64: MWR 36.5 GHz Swath at 58° (Red) & 52° (Green).	81
Figure 65: Earth Gridded T_b s at 37 GHz V pol (53°).....	82
Figure 66: Earth Gridded T_b Difference at 37 GHz V pol (53°).	83
Figure 67: Absolute Yaw vs Latitude.	84
Figure 68: Satellite Velocity Azimuth.	86
Figure 69: Beam Compass Azimuth Angle for Ascending Flight.....	87
Figure 70: Beam Compass Azimuth Angle for Descending Flight.	87
Figure 71: Beam 1 & 8 Collocation Separation at Ascending & Descending Node.	88
Figure 72: Forward/Aft Beam 1 Collocation Separation.....	89
Figure 73: Forward/Aft Beam 2 Collocation Separation.....	90
Figure 74: Forward/Aft Beam 3 Collocation Separation.....	91
Figure 75: Forward/Aft Beam 4 Collocation Separation.....	92
Figure 76: Forward/Aft Beam 5 Collocation Separation.....	93
Figure 77: Forward/Aft Beam 6 Collocation Separation.....	94
Figure 78: Forward/Aft Beam 7 Collocation Separation.....	95
Figure 79: Forward/Aft Beam 8 Collocation Separation.....	96
Figure 80: Forward Aft Beams.	97
Figure 81: A Snapshot of 3 Forward Beams Collocating.	97
Figure 82: All Forward Beams Collocating with Different Aft Scans.	98
Figure 83: Conical Scanning Sensor Geometry (3-D and Polar View) [17].	104
Figure 84: Conical Scanning Sensor Geometry [17].	105
Figure 85: Conical Scanning Sensor Geometry (Satellite Sub-Track Plane View) [17].	107
Figure 86: Radiative Transfer Theory Concept.	110

LIST OF TABLES

Table 1: Random Error Reduction with Averaging Interval	6
Table 2: MWR & WindSat Similarities.....	26
Table 3: Equivalent WindSat Azimuth Angles.....	34
Table 4: Simulated MWR Beams	38
Table 5: IDR Fields.....	41
Table 6: Elliptical IFOV Dimensions	45
Table 7: Rectangular IFOV Dimensions.....	48

LIST OF ACRONYMS/ABBREVIATIONS

CFRSL	Central Florida Remote Sensing Lab
CONAE	Argentine Space Agency
EDR	Environmental Data record (WindSat)
EIA	Earth Incidence Angle
GDAS	Global Data Assimilation System
IDR	Intermediary Data Record (WindSat)
IFOV	Instantaneous Field of View
MWR	Microwave Radiometer
NOAA	National Oceanic & Atmospheric Administration
PSU	Practical Salinity Unit
RTM	Radiative Transfer Model
SAC-D	Satelite de Aplicaciones Centificas – D
SST	Sea Surface Temperature
TDR	Temperature Data Record

CHAPTER 1.INTRODUCTION

The Aquarius/SAC-D Mission (AQ) is a very exciting science mission of discovery, which has the potential to revolutionize our understanding of the ocean surface salinity and the marine hydrological cycle. This is a cooperative international scientific mission developed between the National Aeronautics and Space Administration (NASA) of United States of America (USA) and the Comisión Nacional de Actividades Espaciales (CONAE) of Argentina, which includes instrument contributions from Agenzia Spaziale Italiana (ASI), Centre National d'Etudes Spatiales (CNES), and the Canadian Space Agency (CSA) [1]. The satellite bus (SAC-D service platform) will be provided by CONAE and the launch (using a Boeing Delta-II vehicle), will be provided by NASA in May 2010, from the Western Test Range (WTR) at Vandenberg Air Force Base (VAFB).

The Aquarius Mission focuses on understanding the interactions between the global water cycle, ocean circulation and climate through the measurement of Sea Surface Salinity (SSS). The prime instrument of the mission, an L-band radiometer/scatterometer known as Aquarius, provides global measurements of salt concentration (salinity) at the ocean surface. These data are needed to study ocean circulation and heat capacity of the ocean, which in turn affects Earth's climate and the water cycle.

The current measurements of salinity at the sea surface and deep ocean have been acquired from ships and buoy observations over the past 125 years [2]. While these measurements provide useful climate modeling information, their application in global oceanic models is very limited

because of the lack of global sampling. Since the majority of salinity measurements have been obtained from ships, the resulting sampling has largely been confined to shipping lanes and mostly during the summer. Further, approximately 25% of the globe has not been sampled and greater than 73% have fewer than 10 observations. Thus, the Aquarius mission will advance Earth science by providing a global coverage of SSS measurements that will relate salinity variations to climatic changes and will study the effect of these variations on ocean circulation. After launch, Aquarius is expected to provide SSS measurements for a period of at least three years, which will give an unprecedented view of the ocean's role in climate change.

There are eight science instruments on the SAC-D service platform; however, only the Aquarius instrument (provided jointly by NASA's Goddard Space Flight Center and Jet Propulsion Laboratory) and the Microwave Radiometer (MWR) provided by CONAE are relevant to this thesis. The Aquarius instrument comprises a passive microwave radiometer operating in the protected "radio astronomy" frequency band at 1.413 GHz and an active scatterometer in the space-radar frequency band at 1.26 GHz [2]. The measurement of L-band brightness temperature by the Aquarius radiometer is the primary means for remote sensing SSS, and the ocean backscatter measurements from the scatterometer provides a critical roughness correction for the passive SSS retrievals. In addition, the MWR, operating at K/Ka dual polarized bands, measures the surface brightness temperature in the frequency range sensitive to geophysical parameters over the ocean such as surface wind speed, rain rate, water vapor, cloud liquid water, and sea ice. These geophysical products are useful to the Aquarius microwave radiometer for key brightness temperature corrections required to derive accurate sea surface salinity estimates.

The objective of this thesis is to provide realistic simulated ocean brightness temperatures (T_b) for the 3-channel Microwave Radiometer (MWR) on Aquarius/SAC-D (AQ). These data will be used for pre-launch geophysical retrieval algorithms development and science data processing system validation testing. Analyses are performed to evaluate the proposed MWR measurement geometry and verify the requirements for spatial/temporal sampling. Finally, a preliminary study is performed for the post-launch inter-satellite radiometric calibration using the WindSat polarimetric radiometer on the Coriolis satellite.

The simulated brightness temperature data of MWR will be derived from actual WindSat brightness temperature measurements, which is presently operating on board the Department of Defense (DoD) Coriolis satellite [2]. This dataset will be produced from a four-month data collection of a high spatial resolution WindSat T_b product, the Intermediary Data Record (IDR). MWR shares a number of similarities with the WindSat microwave radiometer. The similarities between MWR and WindSat's frequencies, swath overlap and orbit, make WindSat IDR a suitable input data product for the preparation of simulated MWR brightness temperature data. The earth incidence angles of the two instruments are, however, different. Because the emissivity of ocean is a function of the incidence angle, WindSat brightness temperatures have to be normalized to MWR brightness temperatures via an incidence angle correction, which is derived from a microwave radiative transfer model.

This thesis is organized as follows: Chapter 2 describes the MWR and WindSat orbits and radiometer measurement geometries, which include antenna instantaneous fields of view and swath widths. Chapter 3 analyzes the requirements and presents an algorithm to prepare the

MWR simulated dataset from the WindSat IDR data product. Chapter 4 presents the spatial/temporal and radiometric validation of the simulated dataset and the results of yaw steering verification for the Aquarius satellite. Chapter 5 gives the conclusions and recommendations for future work.

CHAPTER 2. SATELLITES AND INSTRUMENTS

The WindSat polarimetric radiometer on board the Department of Defense (DoD) Coriolis satellite has been used as a data source to simulate the T_b s for the 3-channel Microwave Radiometer (MWR) on Aquarius/SAC-D (AQ) mission. A detailed description of the WindSat & AQ mission objectives, orbits, the relevant instrument configurations, and scan geometry has been presented in this chapter. Also, an inter-satellite orbital analysis is performed, which aids in presenting a preliminary study of the spatial/temporal radiometric collocations between the two satellite sensors.

2.1 Aquarius/SAC-D

The Aquarius/SAC-D mission is being developed as a partnership between the United States (NASA) and Argentina (CONAE) [3]. The Aquarius salinity instrument, the data processing system, and the launch into orbit are the primary contributions from NASA. CONAE, on the other hand, is responsible for building the spacecraft, providing a number of other science payloads, mission operations and the data downlink. The primary purpose of the mission is to accurately measure salinity and provide global ocean salinity maps every 7 days.

The Aquarius salinity measurements will provide information to study the interactions between the water cycle (marine rainfall and evaporation, melting and freezing of ice, and river run-off), the ocean circulation, and the climate. To achieve this objective, a salinity measurement accuracy

of 0.2 psu (Practical Salinity Unit) is required to monitor the seasonal and inter-annual variation of salinity in the open ocean.

As noted in Chapter 1, historically the ocean salinity measurements have been very sparse and irregular. Climatology maps based on the available historical data are interpolated over ~1000 Km scales. The Aquarius/SAC-D mission will provide salinity measurements with a resolution of ~150 Km which is approximately an order of magnitude improvement.

The mean SSS dynamic range is ~5 psu (32.5 to 37.5 in the open ocean) as shown in Figure 1. The blue end of the color scale corresponds to salinity measurements of 33 psu and below, while the red color maps to measurements of 37.4 psu and above. The 0.2 psu accuracy of salinity measurements in a 28 day averaging interval, attributed to the Aquarius mission, yields a 25:1 signal/error relative to mean field. This accuracy and signal/error ratio improves with an increase in the averaging interval as shown in Table 1. For the three years that the Aquarius/SAC-D mission is expected to provide salinity measurements, the accuracy and the signal/error relative to mean field will increase to 0.03 and 190:1, respectively.

Table 1: Random Error Reduction with Averaging Interval

	Instantaneous	7 Days	28 Days	90 Days	1 Year	3 Years
Global RMS (psu)	0.87	0.33	0.20	0.09	0.05	0.03
Mean Signal/Error	6	15	25	55	110	190

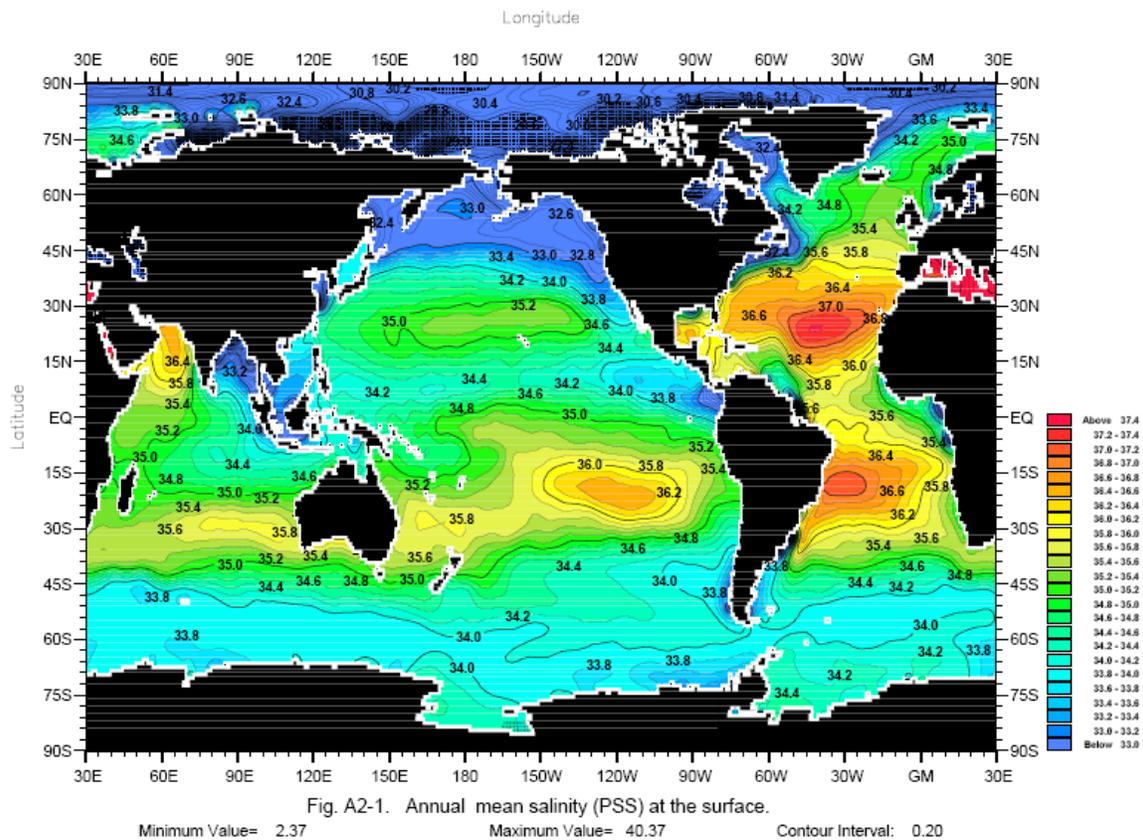


Figure 1: Annual ocean mean salinity with colorbar in practical salinity units (PSS) [4].

In May 2010, the Aquarius/SAC-D satellite will be launched in a sun-synchronous orbit at an altitude of 657 Km, an inclination angle of 98° , and equatorial crossing times of 6 a.m. (descending) and 6 p.m. (ascending), which is a 7-day exact repeat orbit. The Aquarius & MWR instrument swath and the satellite orbit are selected so that complete global sampling is obtained during each 7-day period. The final Aquarius data product will be a monthly map of the salinity field validated against in-situ measurements from ships, buoys, and the Argos system [3]. The Aquarius mission overview is pictorially shown in Figure 2.



Figure 2: Aquarius/SAC-D Satellite Mission [5].

2.1.1 Aquarius Instrument

The Aquarius instrument consists of an L-band radiometer and scatterometer [6], [7], and [8]. The radiometer is the primary instrument for the measurement of sea surface salinity, and it operates in the protected passive-frequency band at 1.413 GHz, which is very sensitive to salinity. The scatterometer, operating in the space-radar band at 1.26 GHz, is an ancillary instrument which provides a critical surface roughness correction to the radiometer's salinity measurements. Both of the instruments are fully polarimetric to provide information to correct for the Faraday rotation from the ionosphere.

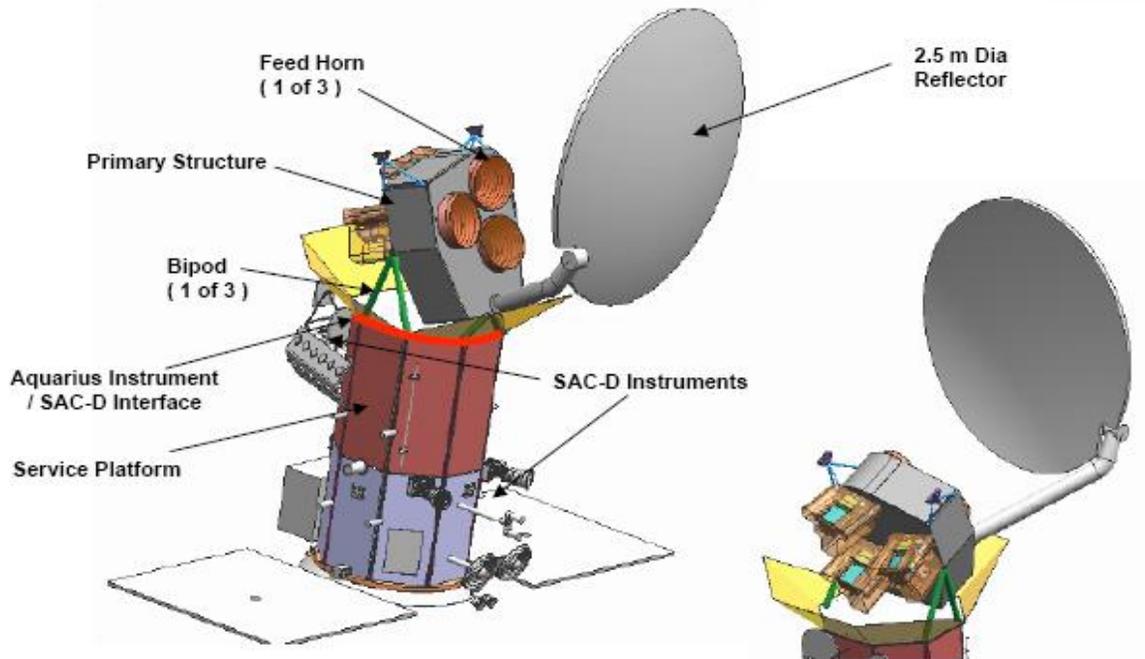


Figure 3: Aquarius Instrument Configuration [9].

The most prominent feature of the instrument is the 2.5 m diameter parabolic reflector with three feed horns. The feed horns are arranged in a triangular geometry and share the reflector antenna as shown in Figure 3. Each feed horn has a separate radiometer, while the scatterometer is sequentially time shared between them.

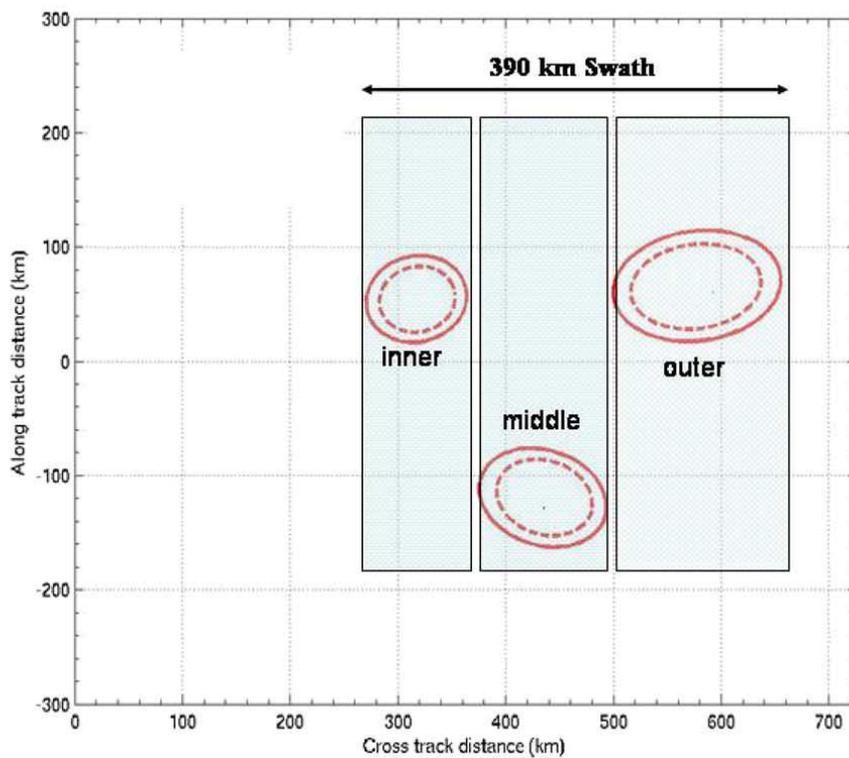
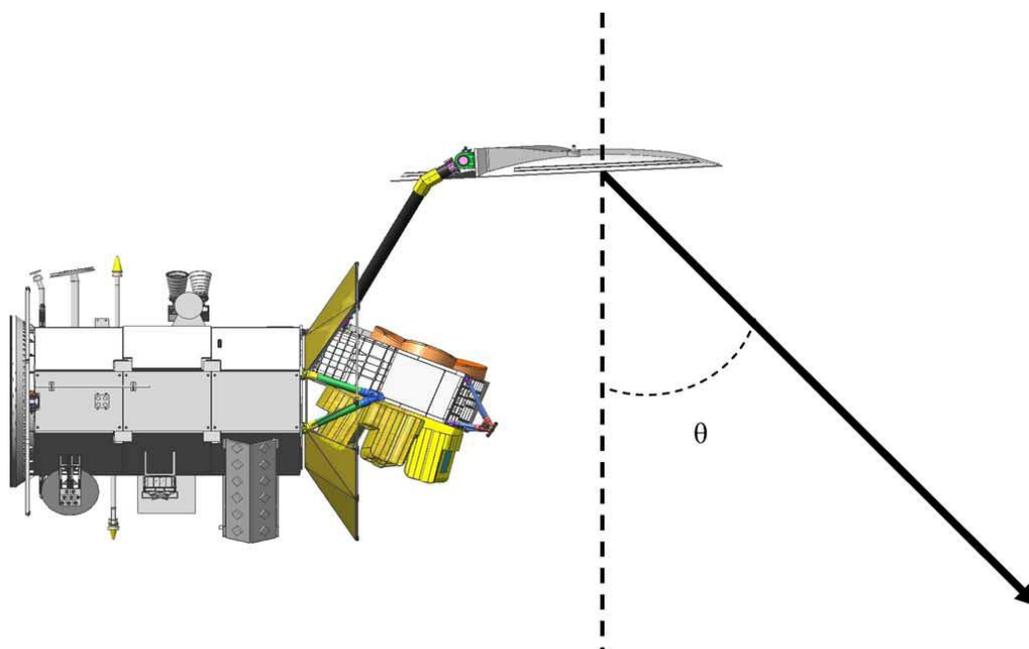


Figure 4: Aquarius Beam Pointing [6].

The feed horns produce three beams which scan in a push broom fashion pointed across-track (roughly 90° with respect to the satellite's flight direction) at nadir angles between 25° and 40° as illustrated in Figure 4. The figure shows the satellite flying into the page while the beams look towards the right side of the flight direction at nadir angles of 25.8° , 33.8° , and 40.3° which correspond to incidence angles of 28.7° , 37.8° , and 45.6° , respectively. The footprints of the three beams are also illustrated at the bottom in Figure 4. The resolution of the beams ranges from 76×94 km for the inner beam to 97×157 km for the outer beam and together they provide a coverage swath of ~ 390 km. The three beams do not point exactly across-track (azimuth = 90°): the inner and outer beams point slightly forward and the middle beam (33.8°) points slightly aft. The dashed ellipses in Figure 4 (bottom) show the 3-dB footprints for the scatterometer. The scatterometer and radiometer beams are aligned at the beam boresight and have approximately the same shape at the 3-dB level. For more information on the Aquarius instrument see [6].

2.1.2 Microwave Radiometer (MWR)

The Microwave Radiometer (MWR), one of CONAE's instruments on the Aquarius satellite, measures the surface brightness temperatures at a frequency range sensitive to geophysical parameters over the ocean and atmosphere e.g. water vapor, wind speed, rain rate, cloud liquid water, and sea ice concentration. MWR geophysical retrievals provide critical corrections to the Aquarius instrument for accurate sea surface salinity estimates.

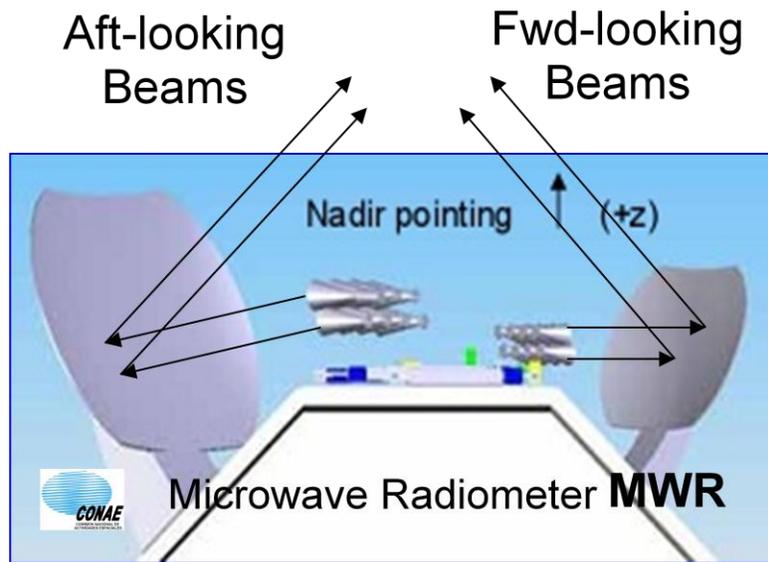


Figure 5: MWR Instrument Configuration [5].

MWR consists of two radiometers operating at K and Ka band, respectively. The aft-looking K band radiometer operates at 23.8 GHz and is only vertically polarized while the forward-looking Ka band radiometer operates at 36.5 GHz and has vertical, horizontal, and $\pm 45^\circ$ polarized outputs. Each radiometer has eight horns arranged as two rows of four feeds that illuminate a parabolic torus reflector as shown in Figure 5. The forward- and aft-looking beams are arranged in a push-broom configuration and operate at two incidence angles 52° and 58° , alternating between the eight beams. For both frequencies, the 3dB beam width is $\sim 1.64^\circ$, which results in an IFOV resolution of ~ 40 Km. The forward and aft looking MWR beams produce a swath which overlaps with the Aquarius instrument's swath to complement SSS measurements and also provides global coverage in a 7-day period.

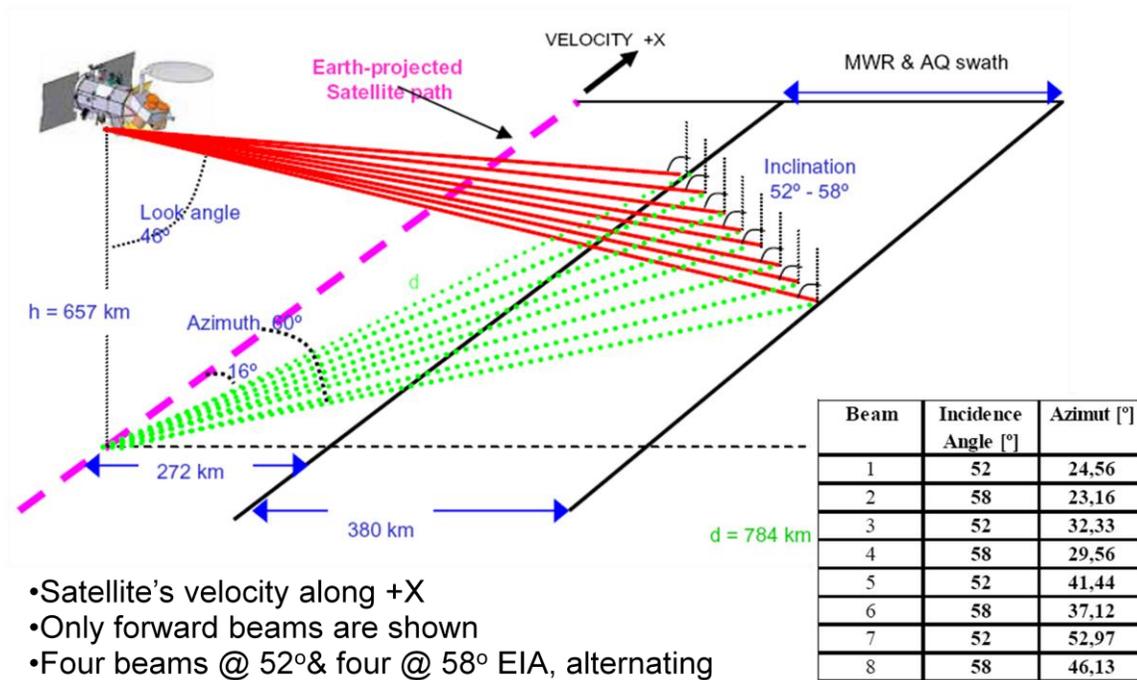


Figure 6: MWR Forward Beam (Ka-Band) Geometry [10].

As the satellite travels in a sun-synchronous orbit at an altitude of 657 Km, the forward- and aft-looking MWR beams always look towards the right side (night side) of the flight direction to avoid the sun glint. Figure 6 shows the sensor geometry of the forward-looking (Ka-band) beams only. The dashed pink line is the satellite's ground track as it travels in the $+X$ direction. The eight forward-looking MWR beams look towards the right side of the ground track at an off-nadir cone angle of $\sim 46^\circ$ and an azimuth range of $\sim 16^\circ$ to 60° . Four of the eight beams (alternating) are at an incidence angle of 52° while the other four are at 58° . Both sets of beams form along-scan conical arcs, which move forward like a push-broom with the satellite's motion. This sensor geometry produces a swath width of ~ 380 Km, displaced 272 Km across-track (towards the right), which exactly overlaps with the Aquarius instrument's swath. Also, the table

in Figure 6 lists the incidence angle and the corresponding azimuth angle of each of the eight forward looking beams.

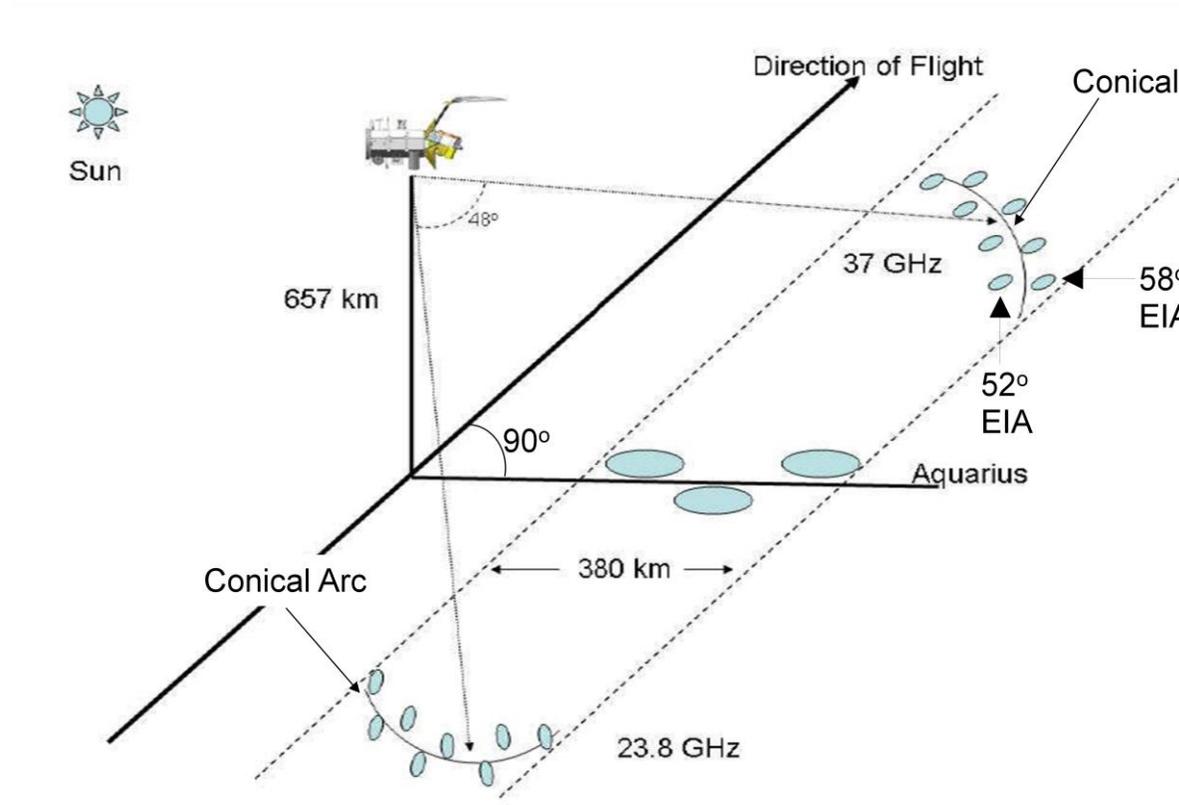


Figure 7: MWR Beam Geometry & Swath Overlap [11].

Figure 7 shows the sensor geometry of MWR and the Aquarius instrument together. Each set of forward- and aft-looking MWR beams form conical arcs at 52° and 58° incidence angles. The MWR swath overlaps with the 380 Km swath produced by the three orthogonal Aquarius beams.

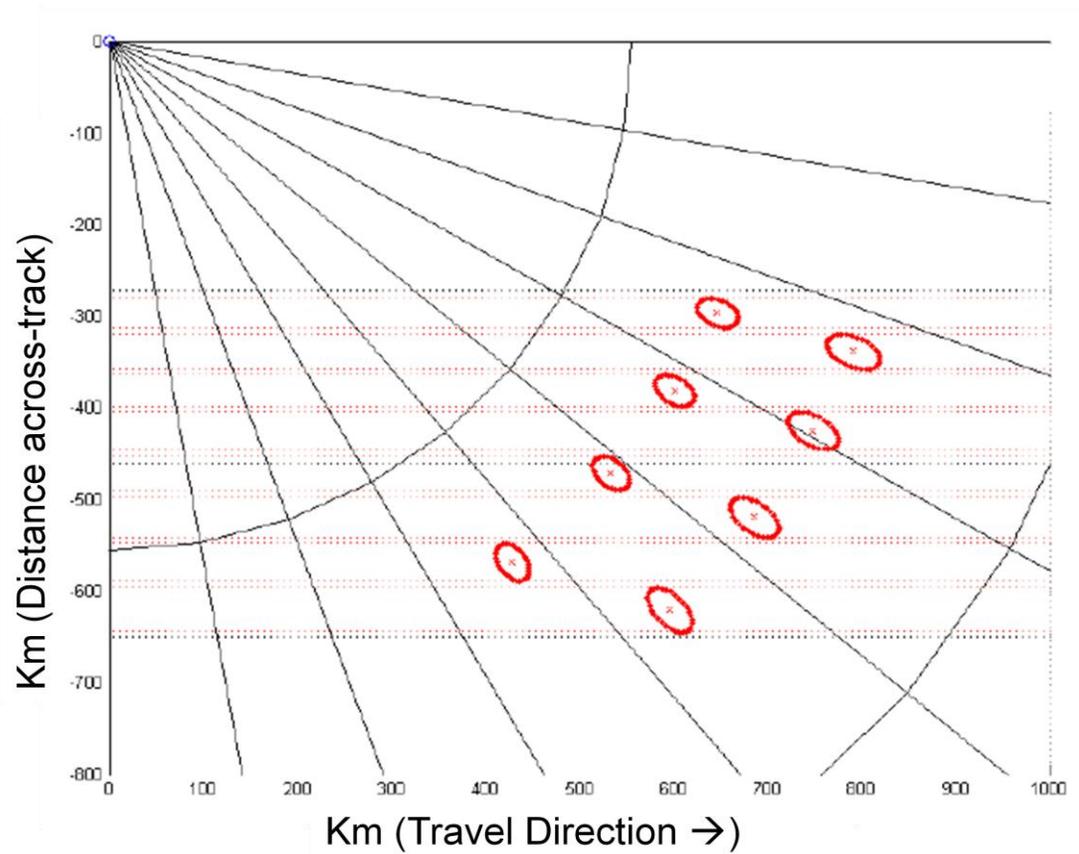


Figure 8: Target Footprints of Forward Looking (Ka-Band) Beams [10].

Figure 8 shows the dimension and orientation of the forward looking MWR footprints. The y-axis shows the across-track distance while the x-axis shows the along-track distance in Km. MWR beams have a 3 dB beam width of $\sim 1.64^\circ$ which forms elliptical IFOVs of two different dimensions at the 52° and 58° incidence angles in a conical arc limited between MWR's azimuth range.

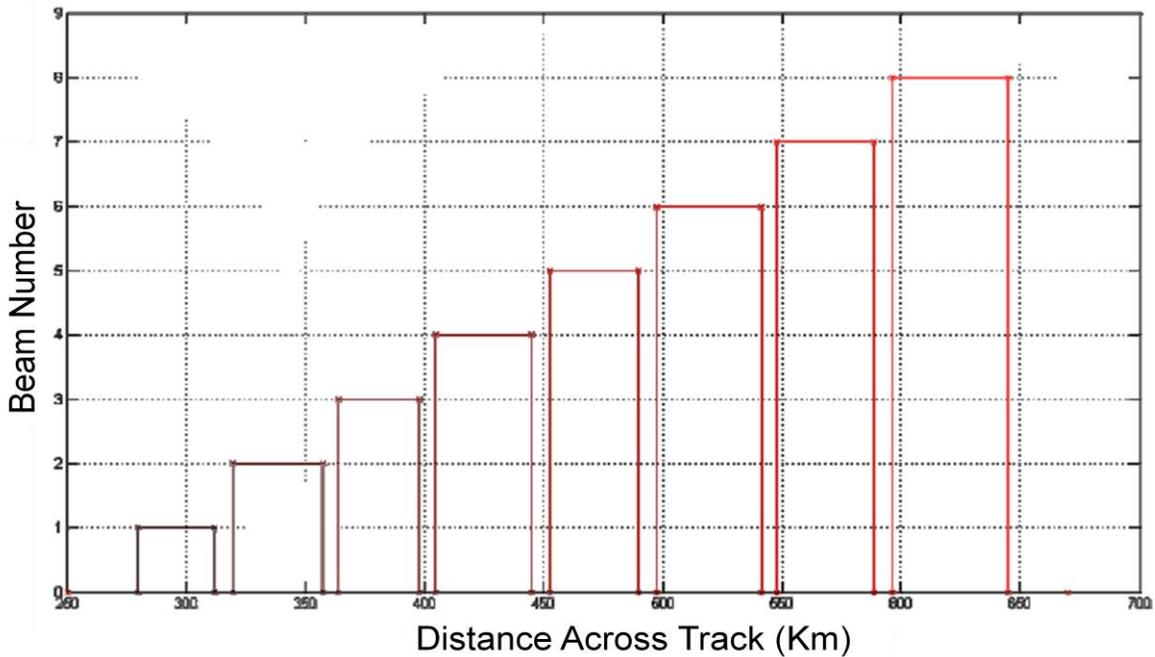


Figure 9: MWR Swath Coverage per Beam [10].

Figure 9 shows the across-track distance covered by each MWR beam. The y-axis shows the beam number while the x-axis shows the across-track distance covered by the beams in Km. The across-track coverage per beam is ~40 Km with a ~6 Km (average) across-track gap between beams.

The MWR beams of a given polarization are input to a single radiometer receiver, which are sampled sequentially with an integration time of 0.24 sec as shown in Figure 10. The beam number indicates the feed horn at any time. The 0.24 integration period is subdivided into eight subintervals of 30 ms in duration and comprises a signal measurement (S), a signal plus noise measurement (S+N), and the Dickie switch to reference load to protect MWR from potential RFI produced by the Aquarius scatterometer transmit pulses. The 0.24 sec integration time of each

beam results in a beam sampling time of 1.92 sec (i.e., each beam is sampled once each 1.92 sec).

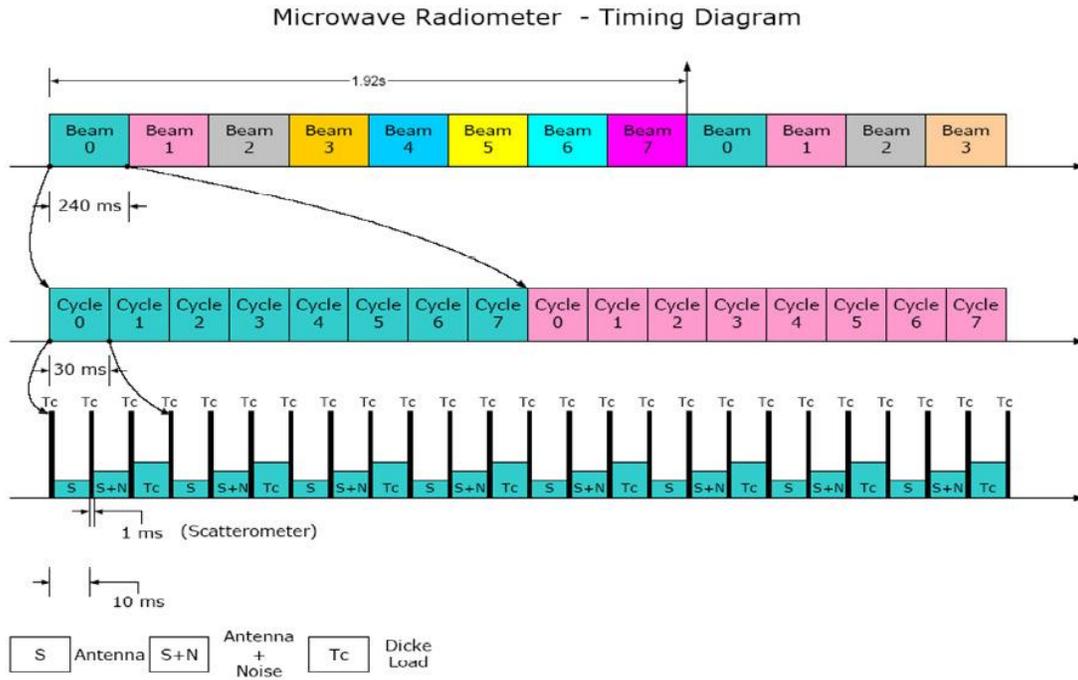


Figure 10: Timing Diagram [10].

At an altitude of 657 Km, the Aquarius satellite travels with a velocity ($V_{s/c}$) of 7.53 Km/sec, which corresponds to a ground velocity (V_g) of 6.82 Km/sec. At this ground velocity, in 1.92 sec beam sampling time, the beams travel an along-track distance of 13.1 Km.

2.1.3 Yaw Steering

The forward and aft looking MWR beams must be collocated for geophysical parameter retrievals; however, it takes ~5 mins for the aft looking beams to reach the forward looking

beams. During this interval, the IFOV's of the forward looking beams have moved towards the east due to the Earth's rotation and desired collocation will not occur. Thus, to compensate for the earth's rotation, the satellite needs to be "yaw steered" to collocate the forward and aft MWR swath as illustrated below.

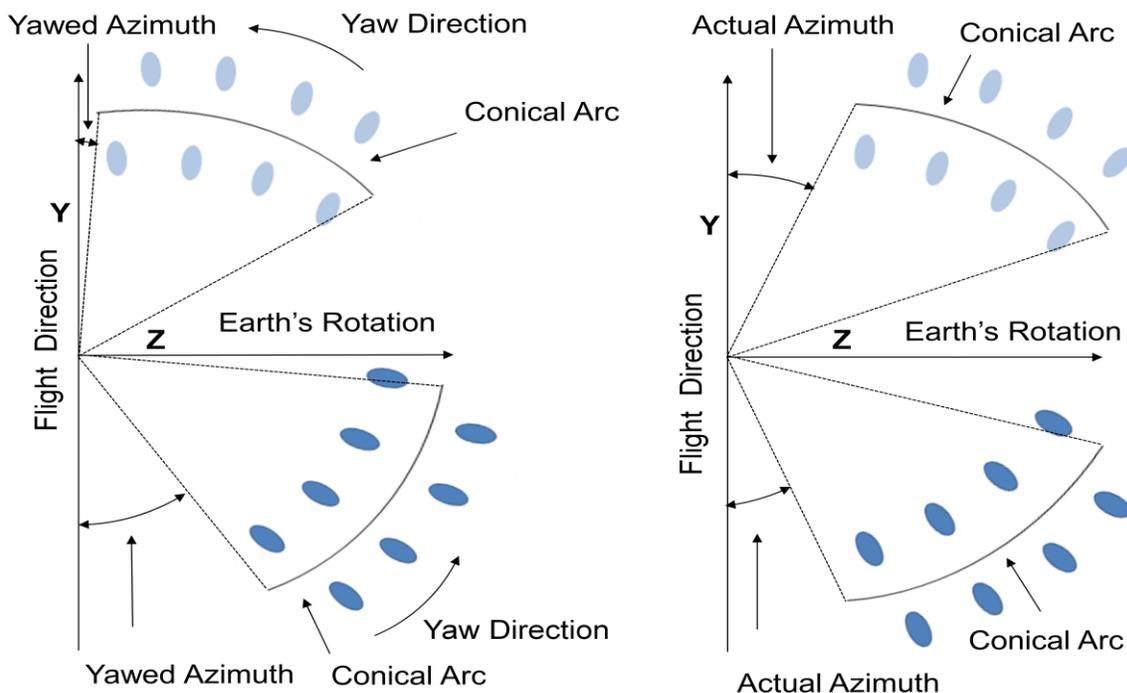


Figure 11: Aquarius/SAC-D yaw steering (left panel) and without yaw steering (right panel).

Figure 11 is a snap-shot in time of the relative IFOV locations for the forward and aft looking MWR beams as the satellite moves along the Y direction and the Earth rotates towards the Z direction (east). The right panel shows the forward and aft swath without yaw steering, and the left panel shows the IFOV's when the spacecraft has positive yaw. Note that these figures are not to scale and the movement of the IFOV's is exaggerated for illustrative purposes. Since the Earth's rotation is maximum at the equator and zero at the poles, the magnitude and direction of

the yaw varies along the orbit as a sinusoid with peak amplitude of approximately 4 degrees. The yaw is anti-clockwise (positive) for ascending flight direction and is clockwise (negative) for the descending flight direction.

2.2 Coriolis/WindSat

WindSat is a satellite-based multi-frequency polarimetric microwave radiometer developed by the Naval Research Laboratory for the U.S. Navy and the National Polar-orbiting Operational Environmental Satellite System Integrated Program Office [2] and [12]. It was designed to demonstrate the capability of polarimetric microwave radiometry to measure the ocean surface wind vector from space. WindSat is the primary payload on the Department of Defense Coriolis satellite, which was launched on January 6, 2003. It is in a 840-km circular sun-synchronous orbit at an inclination angle of 98.7° and equatorial crossing times of 6 a.m. (descending) and 6 p.m. (ascending).

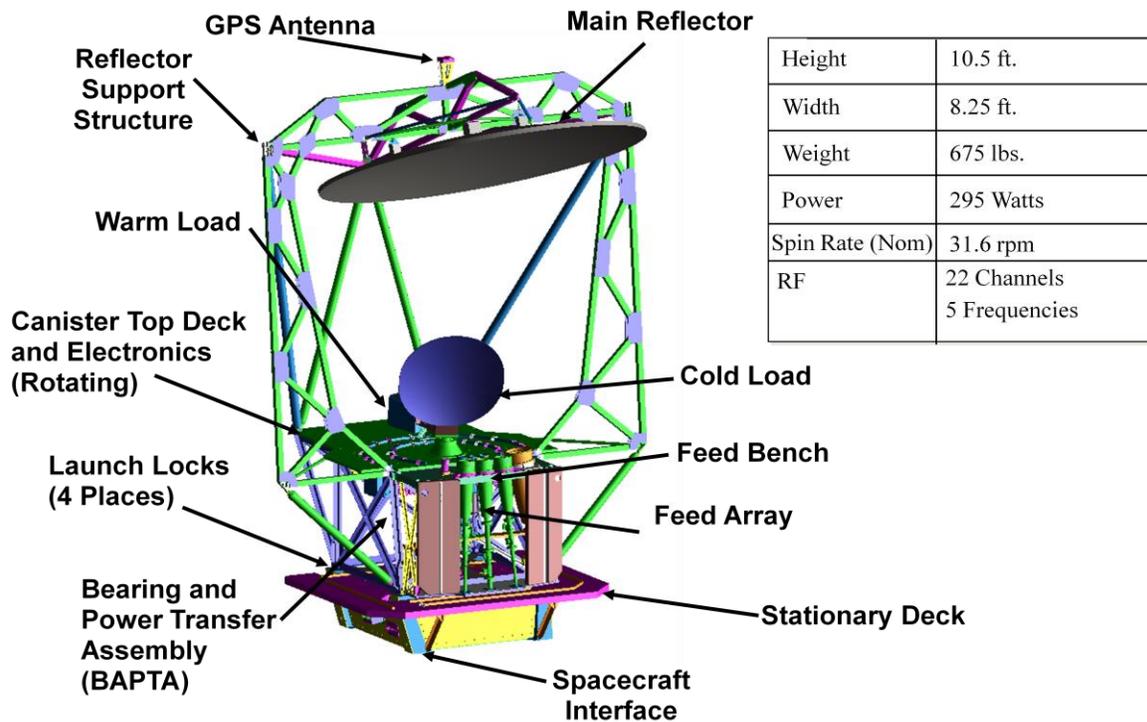


Figure 12: WindSat Mechanical Configuration [12].

Figure 12 shows configuration of the WindSat instrument, which is the first space borne polarimetric microwave radiometer. It operates in discrete bands at 6.8, 10.7, 18.7, 23.8, and 37.0 GHz. The 10.7, 18.7, and 37.0 GHz channels are fully polarimetric, while the 6.8 and 23.8 GHz channels are dual polarized only (vertical and horizontal). Figure 13 shows the WindSat feed-bench with the 6.8 GHz, 10.7 GHz, 18.7 GHz, 23.8 GHz, and 37 GHz horns represented by the color orange, green, blue, magenta, and yellow, respectively. The multiple feeds result in eleven sets of dual-polarized antenna beams as shown in Figure 14. Beams with the same frequency band have the same earth incidence angle. All the frequencies and polarizations are re-sampled and averaged to a common footprint.

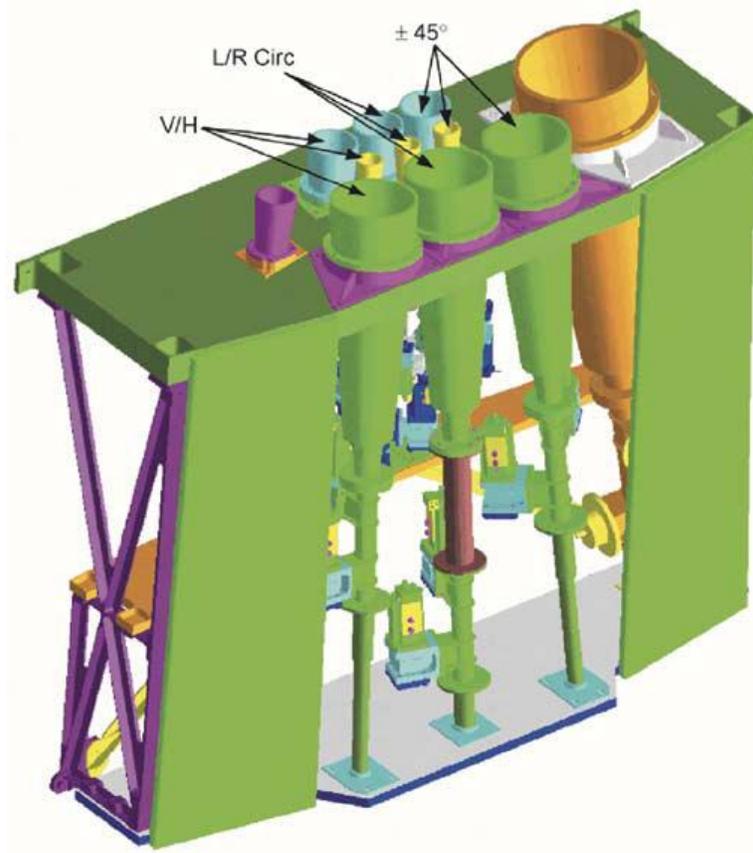


Figure 13: WindSat Feed-bench with multi-frequency/polarization horns [2].

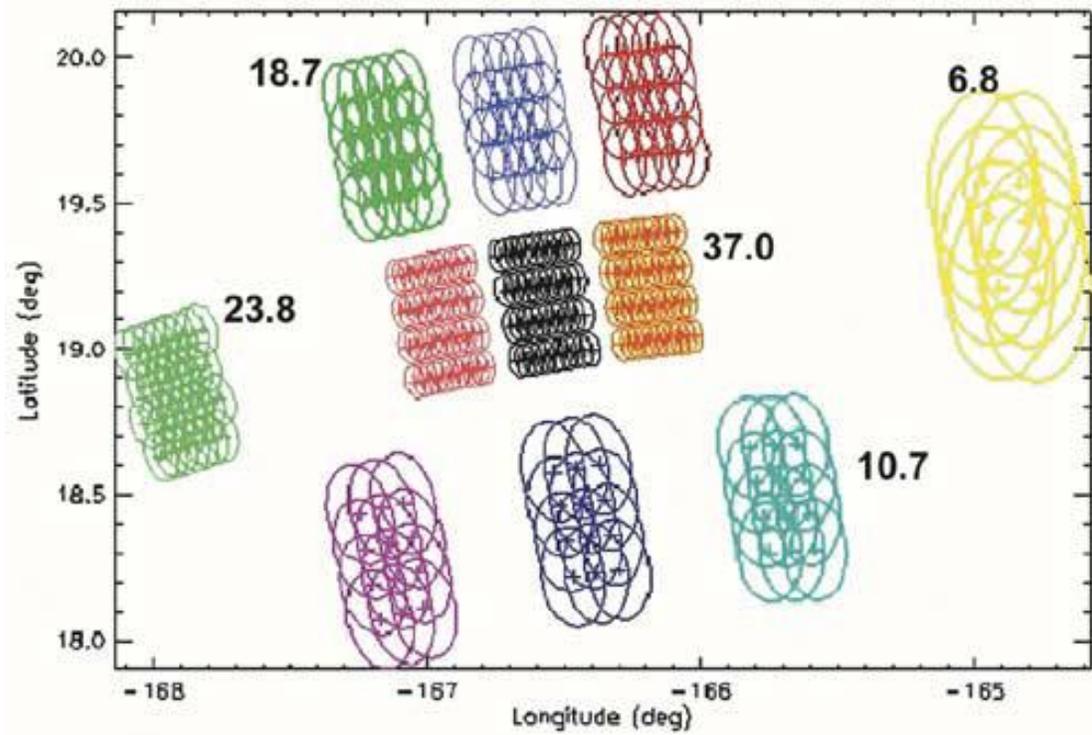


Figure 14: WindSat Earth Projected Beams for multiple antenna scans [2].

Figure 15 shows the WindSat geometry and pointing angles as it flies over the earth. The yellow arrow, curved along the surface of the earth, is the satellite ground-track, and the angle between the beam bore-sight and the normal to the surface is the earth incidence angle (EIA).

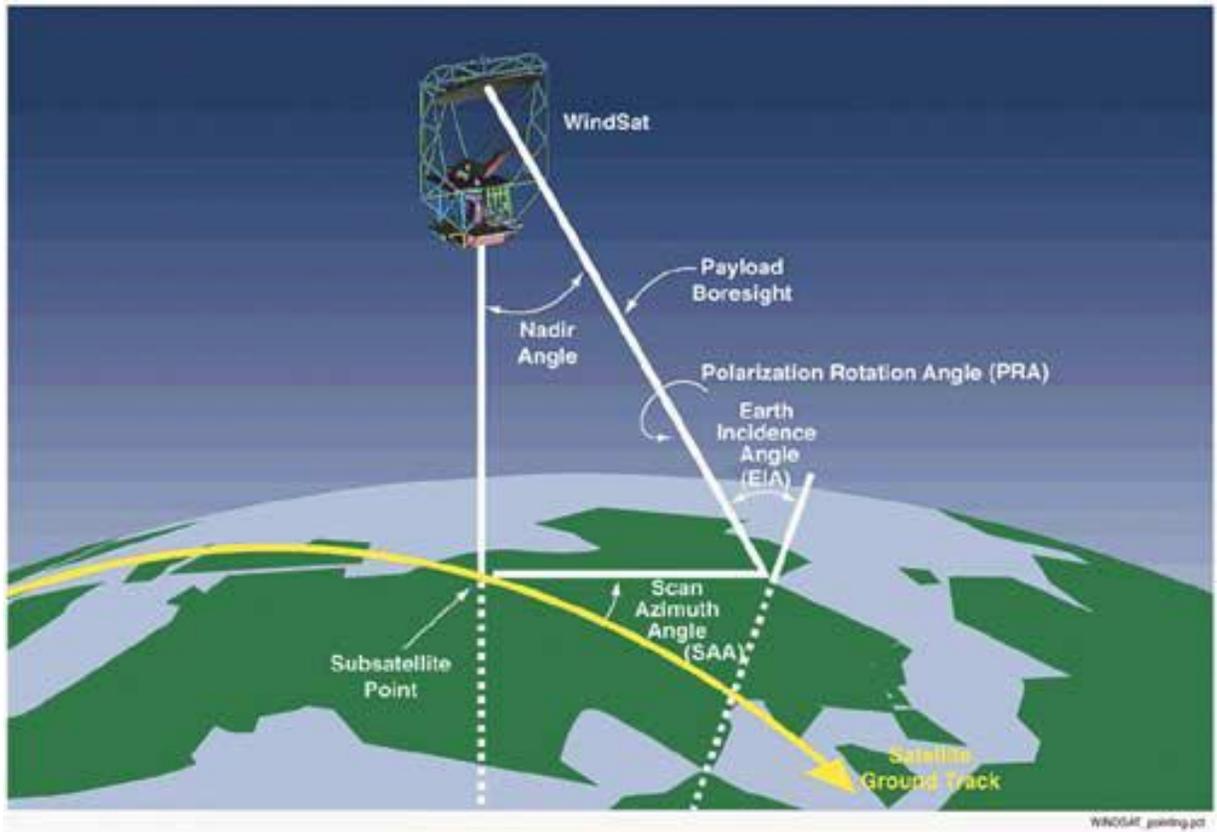


Figure 15: WindSat Geometry & Pointing Angles [2].

WindSat is an anti-clockwise conical scanning radiometer as shown in Figure 16. The satellite's ground-track is shown by the arrow. The scan angles on the right side of the flight direction are negative, while those on the left side are positive. A portion of the forward and aft scan of the WindSat radiometer is also shown in the figure. For further reading on conical scanning sensor geometry see Appendix A.

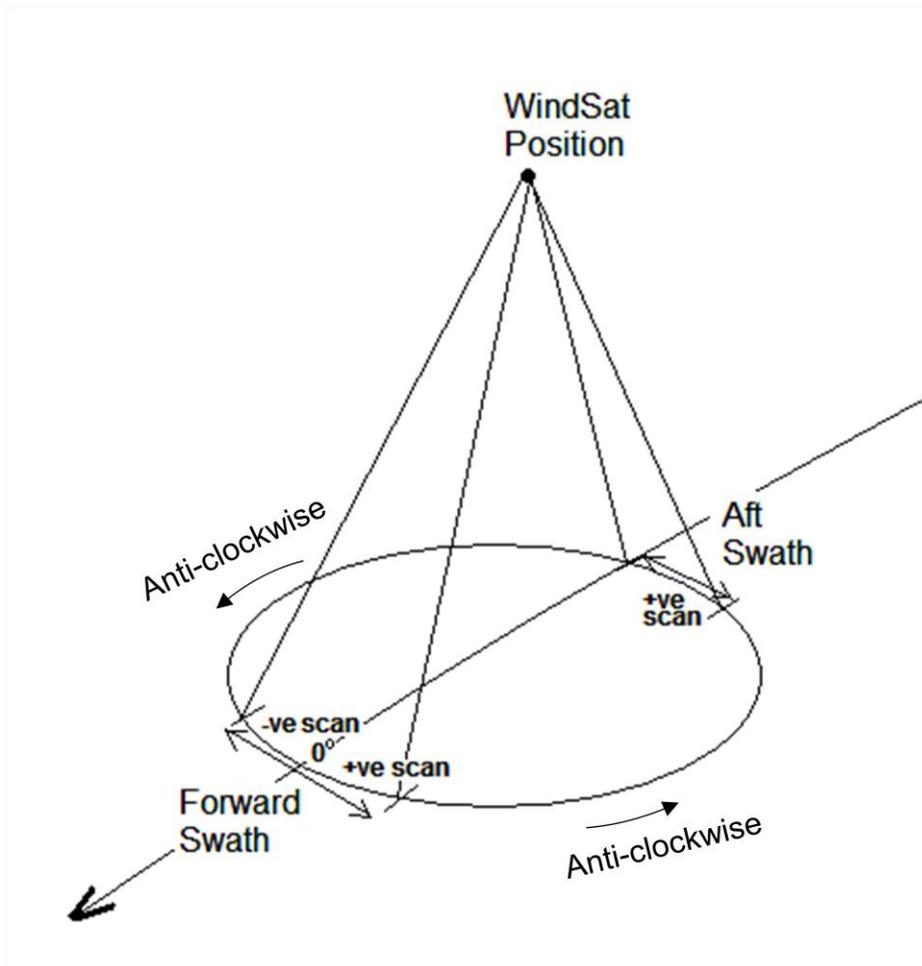


Figure 16: WindSat Conical Scanning geometry.

Figure 17 shows the angular distribution of WindSat swath for each of its frequencies. The 6.8 and 23.8 GHz angular distributions have the same radial positions as 37 GHz, but they have been shifted for clarification. Also, the 23.8 and 37 GHz frequencies have the same earth incidence angle of 53°.

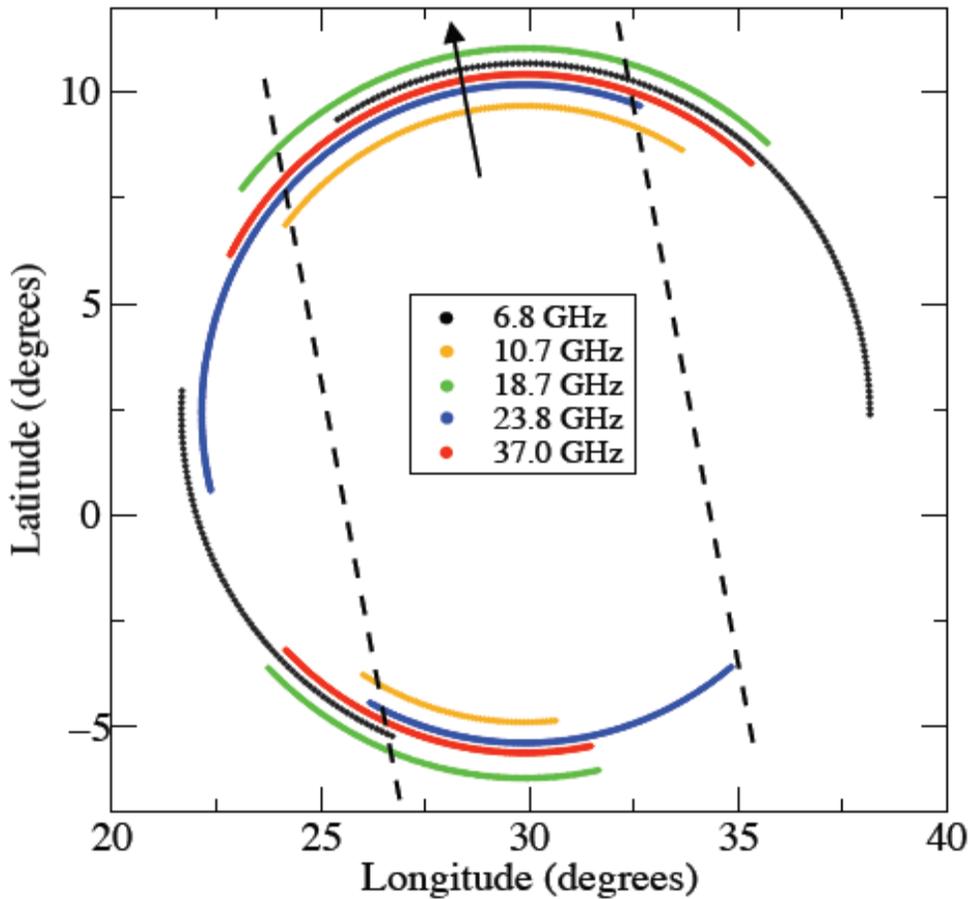


Figure 17: WindSat Swath Angular Distributions for different frequencies [12].

2.3 Inter-Satellite Relative Orbital Periods

WindSat shares similarities in satellite ground-track, radiometric frequencies and swath overlap with MWR. The two sensors are aboard satellites following sun-synchronous orbits with very close inclination angles and eccentricities. They also share the same equatorial crossing times at the ascending (6 p.m.) and descending (6 a.m.) nodes. The 3 channels of MWR are a subset of WindSat's channels, although they differ in earth incidence angles. Moreover, the WindSat swath is much wider (~950 Km) than MWR's swath (~380 Km), which allows considerable

radiometric collocations between the two sensors. Table 2 lists the similarities between MWR and WindSat.

Table 2: MWR & WindSat Similarities

Parameter	WindSat	MWR
Altitude	840 Km	657 Km
Eccentricity	0.00134	0.0012
InclinationAngle	98.7°	98.01°
AscendingNode	6 p.m.	6 p.m.
Frequency	23.8- (V, H) & 37- (V,H) GHz	23.8 (V) & 36.5 (V& H) GHz
SwathWidth	~950 Km	~380 Km
EarthIncidenceAngle	53°	52° & 58°

The orbits of the two satellites were simulated in Satellite Toolkit (STK) software [13]. Aquarius, orbiting at a lower altitude than Coriolis, has a higher velocity and therefore, laps the Coriolis satellite in approximately 45 hours and 36 mins (lap time). When the two satellites are collocated, their ground-tracks are in-phase. As Aquarius begins to lead Coriolis, the ground-tracks gradually phase out to a maximum value when the temporal spacing between the two satellites is exactly half of an orbit period (~ 45 min). Figure 18 shows the scenario when the ground-tracks of the two satellites are in-phase, while Figure 19 shows the ground-tracks out of phase. When the temporal distance between the two satellites is approximately half an orbit, the ground-tracks are maximally out of phase. It was also observed that it takes about 57 days for the satellites to become collocated at approximately the same point on the surface of the earth. This orbit repeat period consists of ~30 lap times i.e., after Aquarius laps Coriolis 30 times in approximately 57 days the ground tracks will repeat.

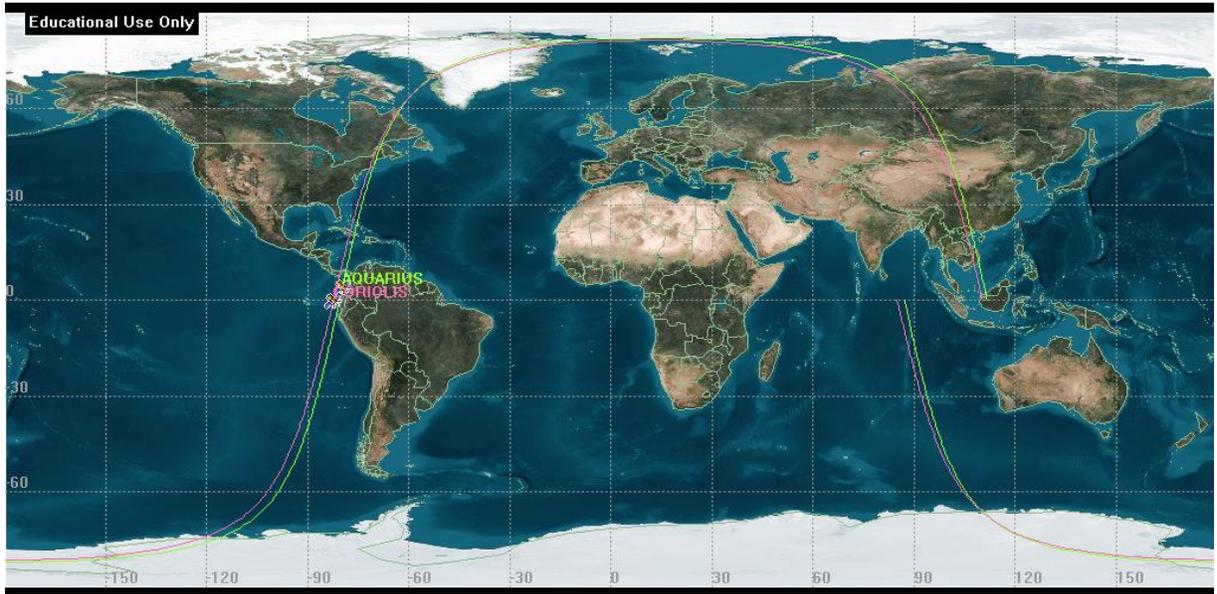


Figure 18: Aquarius & Coriolis Ground-Tracks in-Phase.

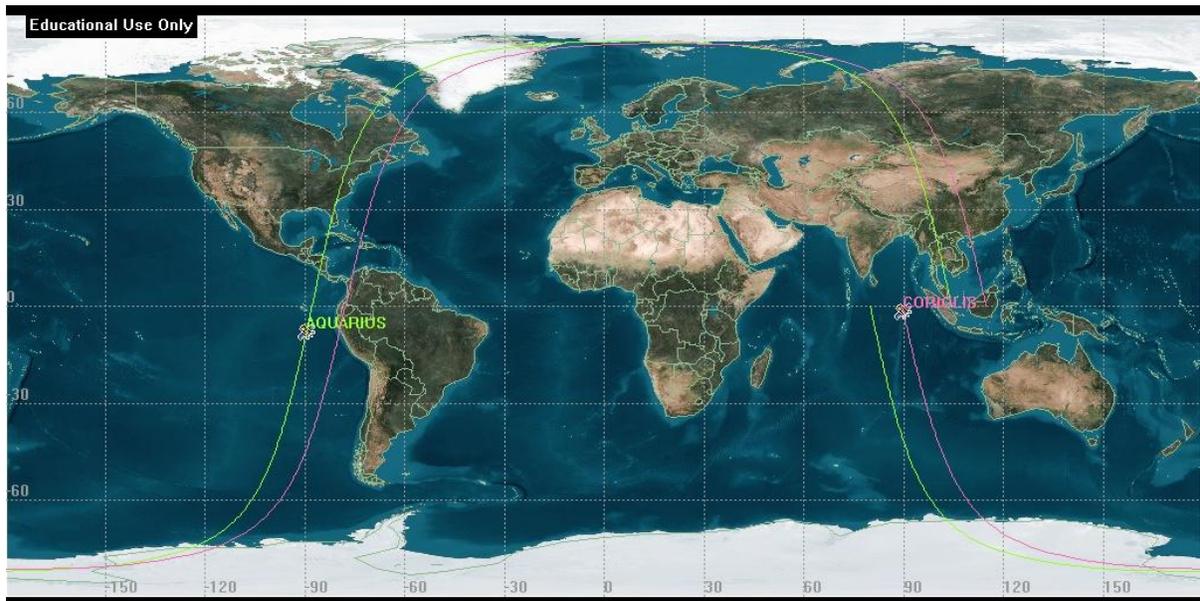


Figure 19: Aquarius & Coriolis Ground-Tracks out of Phase.

2.4 Inter-Satellite Radiometric Spatial/Temporal Collocations

MWR has a significant swath overlap with WindSat. When the Aquarius/SAC-D is in orbit, the swath overlap between the two sensors will provide many spatial/temporal radiometric collocations. This will play a significant role in the post-launch radiometric calibration of MWR using WindSat, a very well calibrated radiometer, as a reference.

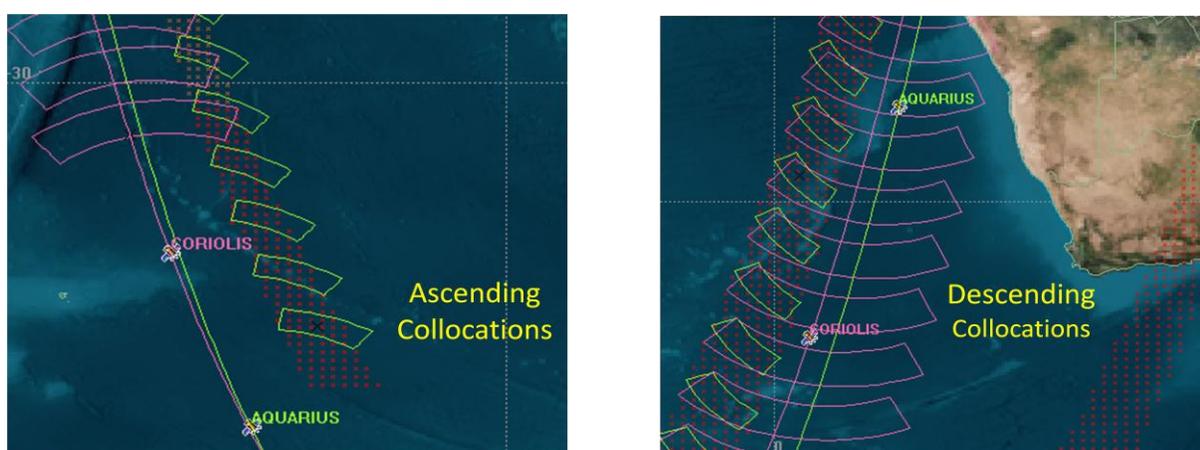


Figure 20: Example of Ascending/Descending Collocations for MWR and WindSat.

The inter-satellite spatial/temporal radiometric collocations were evaluated by simulations using Satellite Toolkit (STK) software. The collocations were observed in a 0.5° resolution earth grid, restricted between $\pm 50^\circ$ latitudes, with a ± 45 min separation window. Figure 20 shows the ascending and descending collocations between MWR (green) and WindSat (magenta) swaths. The red colored points show corners of collocated 0.5° resolution boxes.

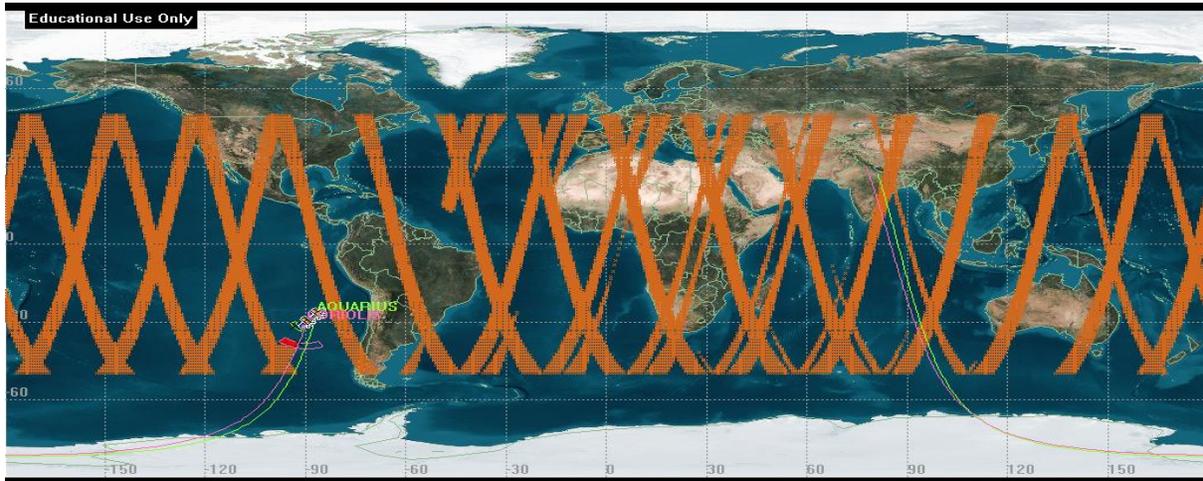


Figure 21: Collocation swath in 45 hours.

Figure 21 shows the collocation swath coverage in ~45 hours. There are about 19,000 (0.5° lat/lng boxes within $\pm 50^\circ$ lat., ± 45 min window) collocations in a 45 hour period. The swaths have maximum collocation when the ground-tracks of the two satellites are in-phase and they have minimum collocation when the ground-tracks are out of phase. The average 19,000 collocations in a 45 hour period result in ~1 million ocean collocations in about 5 months, which is the rough order of magnitude required for inter-satellite radiometric calibration [14].

CHAPTER 3. SIMULATED MICROWAVE RADIOMETER DATASET

This chapter explains the procedure of simulating realistic brightness temperatures for the 3-channel Microwave Radiometer on the Aquarius/SAC-D mission, using the WindSat radiometer brightness temperature observations as the source. These T_b data will be required by CONAE in the development of a pre-launch, set of geophysical retrieval algorithms. The numerous similarities of WindSat with MWR substantiate its significance as a source to simulate MWR T_b dataset. However, the azimuthal distribution of the WindSat swath imposes considerable constraints on the simulation.

WindSat Intermediary Data Record (IDR), a high spatial resolution data product, available for both K/Ka band frequencies, is used as the input to the simulation. The elliptical MWR footprints at 52° & 58° earth incidence angles are approximated as rectangular footprints for simplification and the number of IDR pixels falling into rectangular MWR footprints is calculated. The average T_b of the IDR pixels falling into the rectangular MWR footprints results in averaged WindSat T_b at the MWR spatial resolution. After averaging, these T_b s are corrected for the incidence angle difference between WindSat and MWR beams.

The following section describes the procedure for using WindSat as a source to simulate MWR brightness temperature dataset and highlights the requirements & constraints imposed by it.

3.1 MWR Tb Simulation Requirements

MWR channels are a subset of WindSat channels. The T_b s of MWR channels at 23.8 GHz (V-pol) and 36.5 GHz (V & H pol) can be simulated using corresponding WindSat channels after normalizing the WindSat T_b s due to the frequency and incidence angle differences between the two sensors.

The forward- and aft-looking swaths of MWR must be collocated for geophysical parameter retrievals. In the actual MWR measurements, the aft-looking 23.8 GHz swath (beam IFOVs) is collocated with the forward-looking 36.5 GHz swath (after several minutes) through the SAC-D satellite's yaw steering. However, to achieve the MWR 3-channel collocation, it is sufficient to use the two forward-looking WindSat swaths at K and Ka band to simulate the collocated MWR swath.

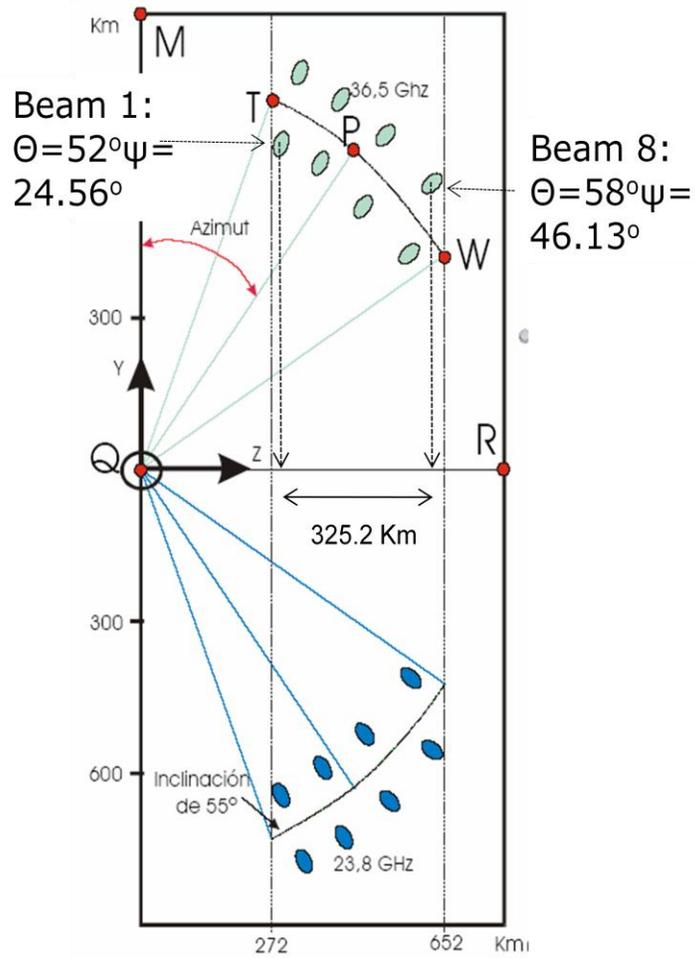


Figure 23: MWR Swath [10].

Figure 23 shows that the MWR beams, which define the edges of its swath, range from 24.56° to 46.13° in azimuth angle (Beam 1 and Beam 8, respectively). The swath on the surface of the earth depends on the range of cross-track central angles (γ_{ct}). The cross-track central angles, which define the MWR swath on the Earth's surface, are the same for MWR and WindSat; however, the earth central angles (γ), in the incident plane, are different ($\gamma_{MWR \text{ at } 58 \text{ deg}} = 7.75^\circ$, and $\gamma_{MWR \text{ at } 52 \text{ deg}} = 6.40^\circ$, and $\gamma_{WindSat \text{ at } 53 \text{ deg}} = 8.04^\circ$). The equation,

$$\sin(\gamma_{ct}) = \sin(\gamma) \times \sin(\psi)$$

from Conical Scanning Sensor Geometry (Appendix A), is used to calculate the corresponding translated WindSat azimuth angles, as shown in Table 3.

Table 3: Equivalent WindSat Azimuth Angles

	Beam-1 (Az, Deg)	Beam-8 (Az, Deg)
MWR	24.56°	46.13°
WindSat	-19.37°	-44.07°

To produce collocated 23.8 and 36.5 GHz MWR beams, only the common azimuth range of WindSat K and Ka band forward swaths can be utilized. Figure 24 shows the angular distribution of WindSat’s 23.8 GHz (blue) and 37 GHz (red) swaths. The azimuth angles of 37 GHz WindSat swath range from -50° to 53° (forward scan) and 125° to 183° (aft scan), while those of 23.8 GHz WindSat swath range from -28° to 94° (forward scan) and 142° to 211° (aft scan). In this simulation, the 23.8 V channel of MWR is simulated as a forward swath collocated with 36.5 (V & H) channels. The common swath between WindSat’s 23.8 and 37 GHz frequencies, ranging from -28.8° to 53° , is not wide enough on the right side of the flight direction to simulate the actual right-sided MWR swath. Therefore, a mirrored left-sided swath with the same azimuth range, but opposite polarity, is a viable solution ($+19.37^{\circ}$ to $+44.07^{\circ}$).

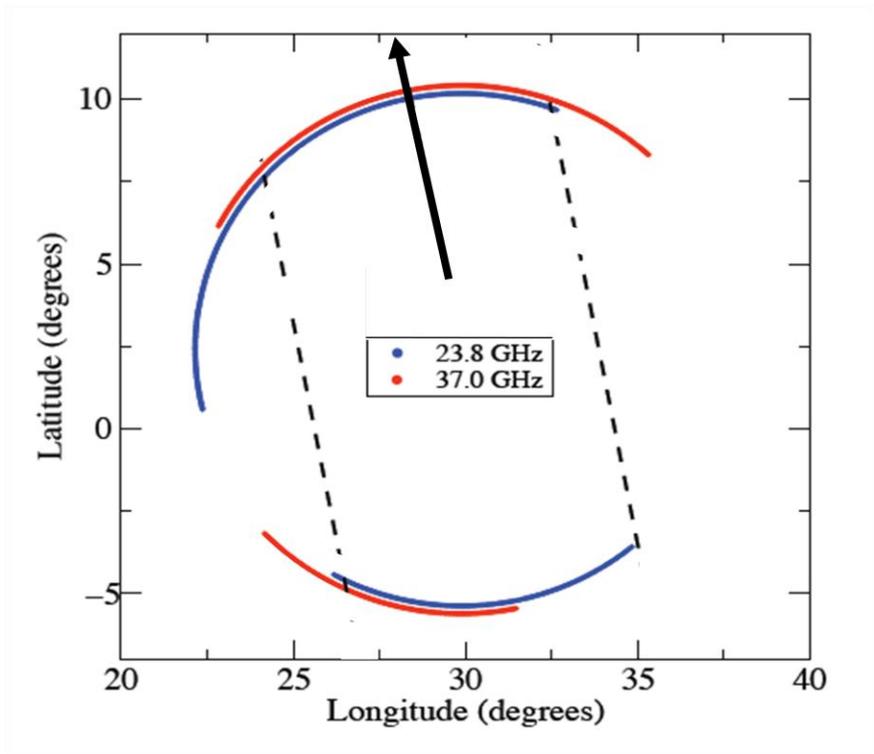


Figure 24: WindSat K/Ka Band Scan Geometry [10].

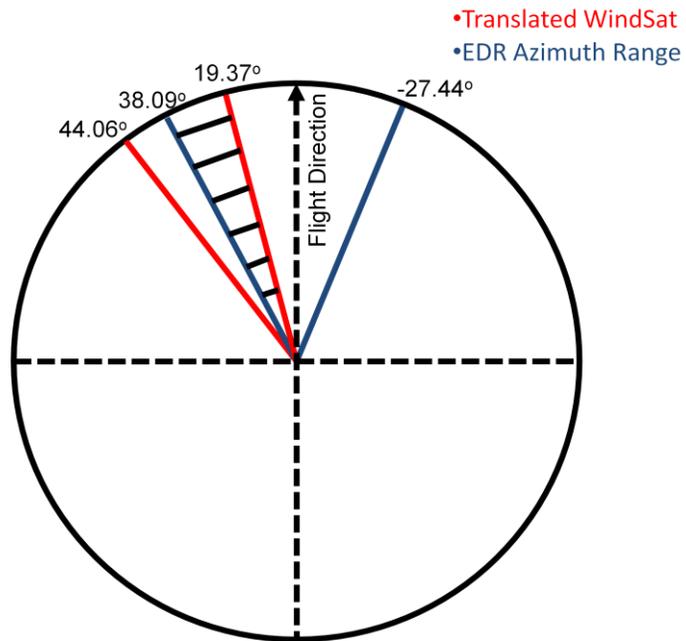


Figure 25: EDR Azimuth Limits.

The MWR geophysical retrievals will be validated through near simultaneous, collocated inter-comparison with WindSat environmental data records (EDRs). Although, the WindSat azimuth range, required to simulate the left-sided mirrored MWR swath, varies from 19.37° to 44.06° , it is further restricted by the azimuth limits of WindSat EDR. The EDRs vary in azimuth angle from -27.44° (right side of flight direction) to 38.09° (left side of flight direction). This implies that the left most MWR beam cannot have an azimuth angle greater than 38.09° as shown in Figure 25. The figure shows the azimuth limits of EDR (blue) and the translated WindSat azimuth range (red), mirrored on the left side. The shaded sector shows the range of azimuth angles that can be used to simulate MWR brightness temperatures. The $\sim 6^{\circ}$ loss in the left most MWR azimuth range, accounts to ~ 2 MWR beams; therefore, only 6 MWR beams can be simulated in the shaded azimuth range.

Since the MWR swath, including cross-track distance, could not be simulated using WindSat IDR/EDR data, it was decided to abandon this approach and to simulate the maximum possible number of MWR pixels for geophysical algorithm development. This resulted in 19 equivalent MWR beams with proper IFOV size, orientation, and relative beam to beam spatial sampling. Further, the grouping of any 8 of 11 adjacent beams will simulate the MWR measurement swath but not the cross-track distance, which has no significance to the geophysical retrievals.

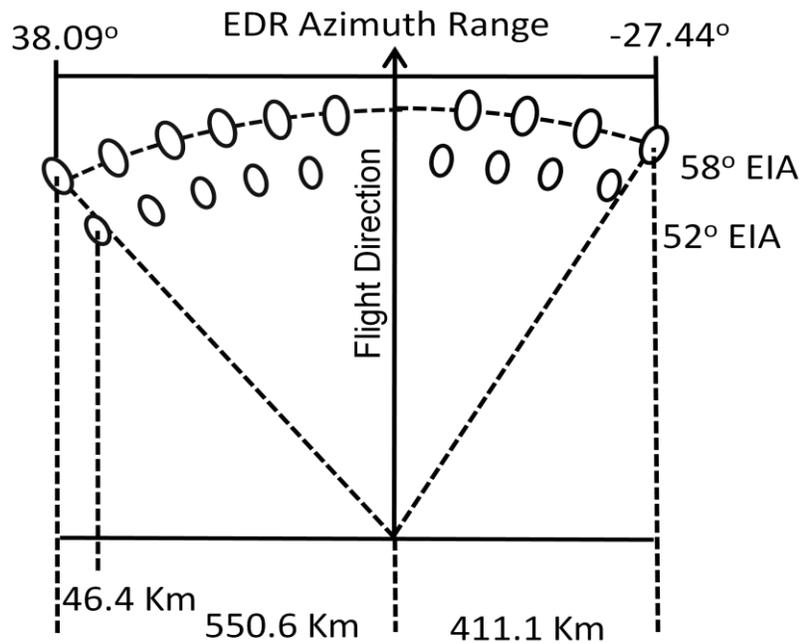


Figure 26: Nineteen Simulated MWR Beams.

Figure 26 shows the nineteen simulated MWR beams (eleven on the left side and eight on the right side of the flight direction). The left most and the right most beam centers correspond to a swath width of 550.6 Km and 411.1 Km, respectively. The beam boresight spacing, projected across-track, corresponds exactly to MWR spacing of 46.4 Km ($325.2/7$ Km), see Figure 23. The beams have accurate IFOV orientation by azimuth and the first eight beams on either side of the flight direction are symmetric. The center azimuth angle of each of the nineteen beams is given in Table 4.

Table 4: Simulated MWR Beams

Beam Number	Swath Width (Km)	$\gamma_{ct} = \text{Swath Width}/\rho$ (Deg)	$\Psi_{beam} = \sin^{-1}(\sin(\gamma_{ct})/\sin(\gamma))$ (Deg)	EIA
1	550.6	4.95°	38.09°	58°
2	504.2	4.53°	34.39°	52°
3	457.8	4.11°	30.86°	58°
4	411.4	3.69°	27.45°	52°
5	365	3.28°	24.14°	58°
6	318.6	2.86°	20.91°	52°
7	272.2	2.45°	17.75°	58°
8	225.8	2.03°	14.64°	52°
9	179.4	1.61°	11.57°	58°
10	133	1.19°	8.54°	52°
11	86.6	0.77°	5.53°	58°
12	-86.3	-0.77°	-5.52°	52°
13	-132.7	-1.19°	-8.53°	58°
14	-179.1	-1.61°	-11.56°	52°
15	-225.5	-2.03°	-14.63°	58°
16	-271.9	-2.44°	-17.74°	52°
17	-318.3	-2.86°	-20.90°	58°
18	-364.7	-3.28°	-24.12°	52°
19	-411.1	-3.69°	-27.44°	58°

MWR T_b is simulated as the average brightness temperature of a group of WindSat's Intermediary Data Record (IDR) pixels, falling into the IFOV of the MWR beam. The following section gives a detailed description of WindSat's IDR.

3.2 WindSat IDR

Measurements collected by the WindSat instrument are processed and archived at different levels for a variety of calibration, validation, and general analysis [15]. The Raw Data Record (RDR) is the fundamental dataset which represents the voltage measurements of the instrument hardware. The next level of processing produces the Temperature Data Record (TDR), which is a collection of T_b measurements, encompassing all the independent frequencies and polarizations. WindSat

processing also produces a rather unique data set called the Intermediary Data Record (IDR). The IDRs provide the four Stokes' parameters associated with the same surface footprints as the TDRs for the polarimetric channels (10.7, 18.7, and 37.0 GHz), and the first two Stokes' parameters for the non-polarimetric channels (6.8 and 23.8 GHz).

The simulated MWR T_b dataset has been produced using WindSat IDR's polarimetric channel at 37 GHz and the non-polarimetric V-pol channel at 23.8 GHz. The IDRs provide a high spatial resolution sampling, along a conical scanning arc. Figure 27 shows the along-scan and inter-scan overlap of conical scanning IDR pixels (IFOVs) at 37 GHz for two consecutive scans.

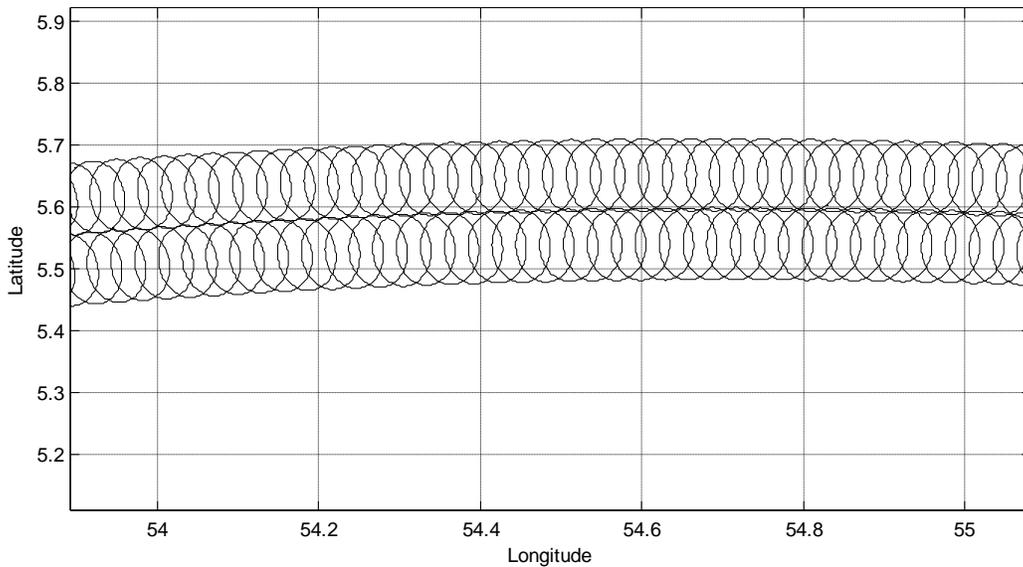


Figure 27: Conical Scanning IDR Pixels at 37 GHz for two antenna scans.

Figure 28 shows the along-scan and inter-scan overlap of conical scanning IDR pixels at 23.8 GHz (two consecutive scans).

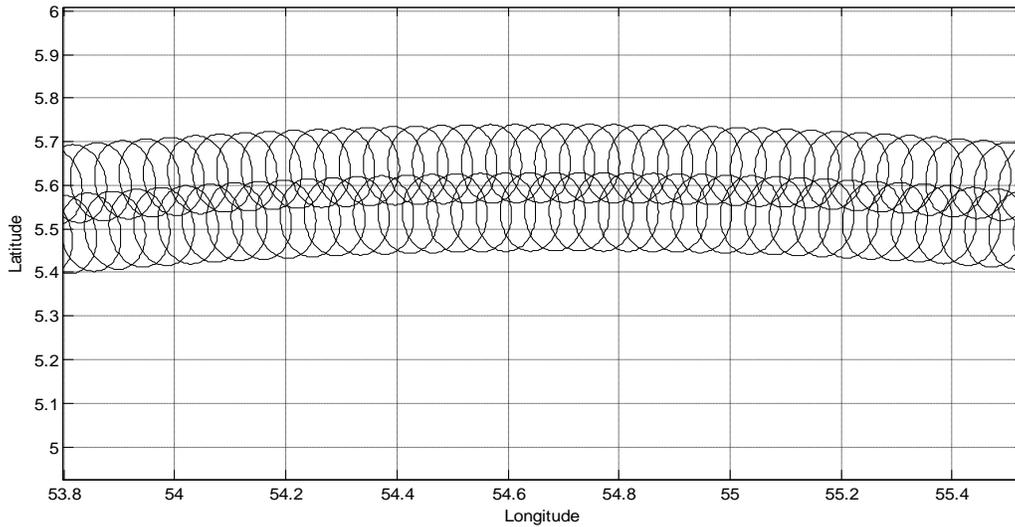


Figure 28: Conical Scanning IDR Pixels at 23.8 GHz for two antenna scans.

Figure 29 shows the spatial resolution and overlap of IDR pixels at the two frequencies. The spatial resolution of IDR pixels at 37 GHz is 8 x 13 Km and is 12 x 20 Km at 23.8 GHz, respectively. There is a 58.8% across-track overlap between IDR pixels at 37 GHz and a 59.2 % across-track overlap at 23.8 GHz, which corresponds to a 3.3 Km along-scan sampling at 37 GHz and a 4.9 Km sampling at 23.8 GHz. The inter-scan pixel boresight separation at both the frequencies is 12.6 Km, which accounts to an inter-scan overlap of 3.1% at 37 GHz and 37.0% at 23.8 GHz.

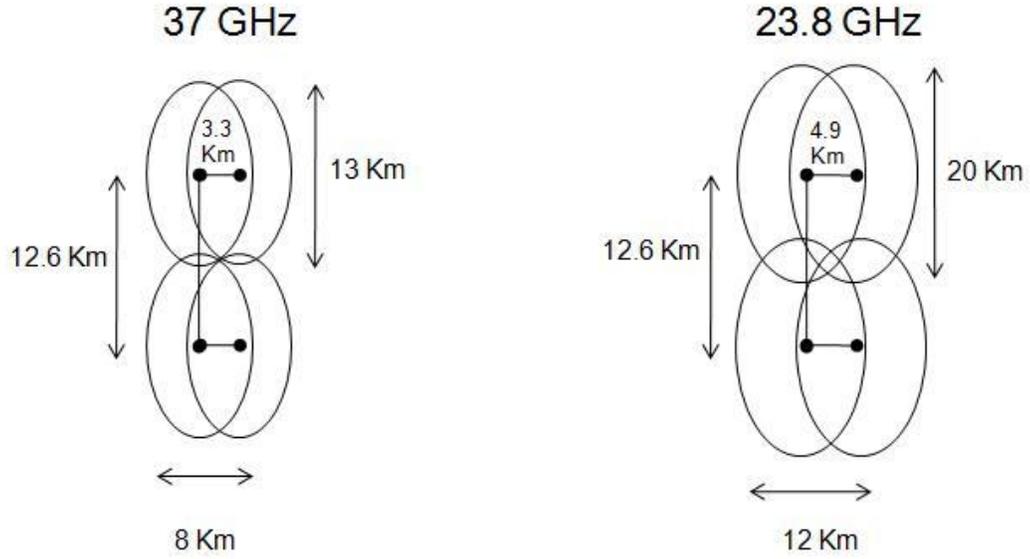


Figure 29: IDR Pixels at 37 GHz & 23.8 GHz.

Table 5 shows the fields present in the IDR data product. A detailed description of each field can be found in [16] and is also included below:

Table 5: IDR Fields

Column Number	Field Name
1	JD2000
2	$T_b V$
3	$T_b H$
4	3 rd stoke
5	4 th stoke
6	Latitude
7	Longitude
8	EIA
9	PRA
10	CAA
11-13	RSATECF
14	ERRFlag
15	Scan
16	DCNum
17	Surface Type
18	Spare

- “*JD2000* is the time of the measurement in Julian Day 2000 format, the number of seconds since January 1, 2000 noon GMT. *JD2000* can be converted to seconds past midnight by $SPM = (JD2000 + 43200) \text{ modulus } 86400$.
- $T_b V$ is the brightness temperature at vertical polarization.
- $T_b H$ is the brightness temperature at horizontal polarization.
- Column 4 & 5 are the 3rd and 4th stoke.
- *Latitude and Longitude* is the location of the pierce point projected onto the earth. Range of the longitude is $\pm 180^\circ$.
- *EIA* contains the earth incidence angles of the 5-feedhorn frequencies (radians). For the bands with three feeds, and therefore three EIAs, only the V/H EIA is written. EIA differences among the three horns at any one band are less than 0.02° .
- *PRA* is the polar rotation angle of the 5-feedhorn frequencies (radians). For the bands with three feeds, and therefore three EIAs, only the V/H PRA is written. PRA differences among the three horns at any one band are negligible.
- *CAA* is the compass azimuth angle, or the radiometer viewing angle, at the pierce point (radian) relative to north for the 37 GHz V/H horn. North equals 0° and the CAA increases clockwise.
- *RSAT_ECF* is the location of the satellite at the 37-GHz VH measurement time in the Earth-Centered-Fixed coordinate system (meters).
- *ERRFlag* is a bit pattern indicating errors in the processing or quality control.
- *Scan* is the number of BAPTA spins beginning at the start of the TDR file. This number is incremented when a sync mark is crossed.

- *DCNum* is the down count number associated with this measurement and pixel location.
- *Surface type* is an integer value where:
 - 0 = LAND
 - 1 = NOT USED
 - 2 = NEAR COAST
 - 3 = ICE
 - 4 = POSSIBLE ICE
 - 5 = OCEAN
 - 6 = COAST
 - 7 = SPARE”

The azimuth angle (scan angle) of IDR pixels is a function of the down count number. Since the spatial resolution of IDR pixels is different for 37 GHz and 23.8 GHz, their azimuth angles are calculated as two separate functions of the down count number. The down count number associated with the flight direction (0° azimuth angle) is 985 for 37 GHz and 680 for 23.8 GHz. The change in azimuth angle per IDR pixel is 0.21° for 37 GHz and 0.313° for 23.8 GHz. The azimuth angle as a function of down count number, at 37 GHz and 23.8 GHz, is given by Eqn. (3-1) & Eqn. (3-2), respectively:

$$\psi_{37GHz} = 3.71528 - DCNumber \times 0.0054636 \quad (3-1)$$

$$\psi_{23.8GHz} = 3.61236 - DCNumber \times 0.0036657 \quad (3-2)$$

3.3 MWR Footprint Analysis

The dimensions of the IFOV of a conical antenna beam depend upon the beam width (β radians), the slant range (R), and the earth incidence angle (θ_i), as shown in Figure 30.

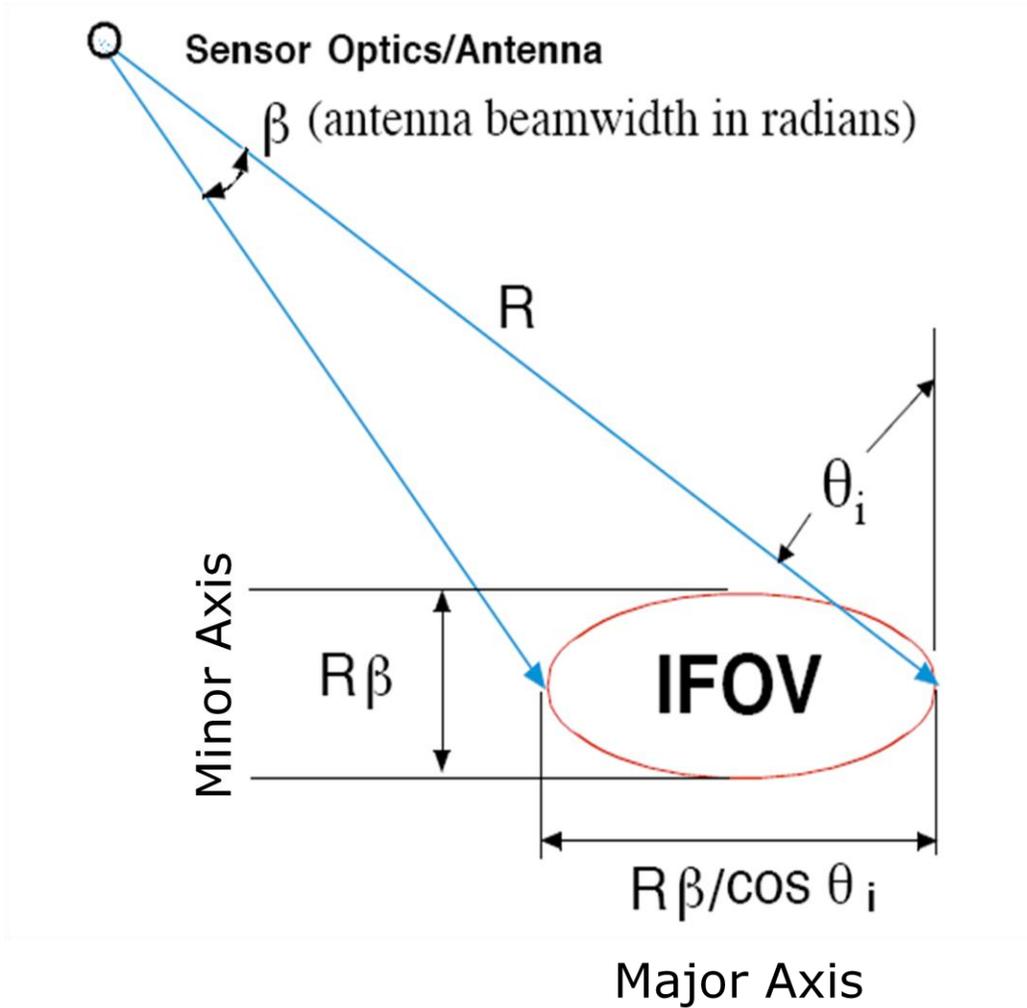


Figure 30: Sensor Instantaneous Field of View (IFOV) [17].

$$IFOV_{MinorAxis} = R \times \beta \quad (3-3)$$

$$IFOV_{MajorAxis} = R \times \beta / \cos \theta_i \quad (3-4)$$

The 3 dB beam width of MWR is 1.64° [10]. The slant range at 58° and 52° incidence angle is 1118.57 Km and 995.83 Km, respectively. The dimensions of elliptical IFOVs, at 52° and 58° incidence angles, are calculated using Eqn. (3-3) & Eqn. (3-4), and are listed in Table 6.

Table 6: Elliptical IFOV Dimensions

Incidence Angle	Minor axis (Km)	Major axis (Km)
58°	32.02	60.42
52°	28.50	46.29

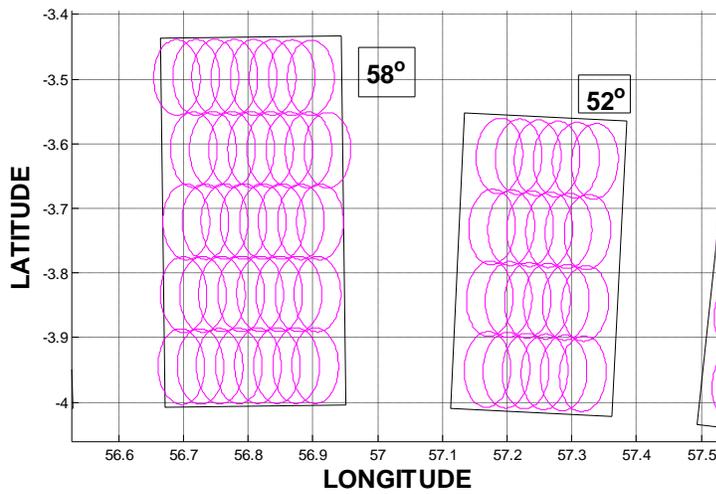
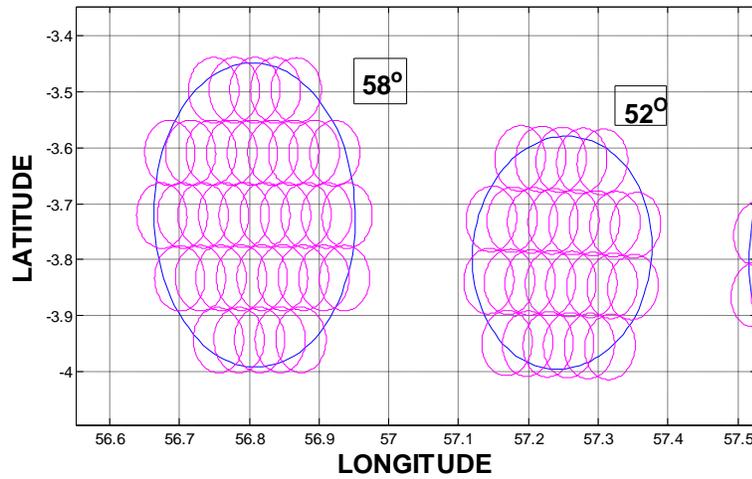


Figure 31: Elliptical & Rectangular MWR IFOVs.

The MWR IFOVs are assumed to have a rectangular shape instead of an ellipse. Figure 31 shows the elliptical and rectangular IFOVs of MWR, filled by 37 GHz IDR pixels, at both incidence angles. The number of IDR pixels in a scan varies along the major axis of the elliptical MWR IFOV. In contrast to elliptical footprints, rectangular IFOVs have an equal number of IDR pixels along the length of the footprint. The accuracy of the MWR T_b spatial averaging does not decrease significantly by using rectangular footprints in place of ellipses; however, the complexity of implementation is reduced. The number of IDR pixels falling into a rectangular MWR footprint, along scan and along track, is calculated using the following equations:

$$n = ((MinorAxis - w) / x) + 1 \quad (3-5)$$

$$m = ((MajorAxis - l) / y) + 1 \quad (3-6)$$

Where ‘n’ is the integer number of IDR pixels along scan, ‘m’ is the integer number of IDR scans, ‘w’ is the minor axis of the IDR pixel, ‘l’ is the major axis of the IDR pixel, ‘x’ is the along scan boresight distance between adjacent IDR pixels and, ‘y’ is inter-scan boresight distance, as shown in Figure 32 and Figure 33.

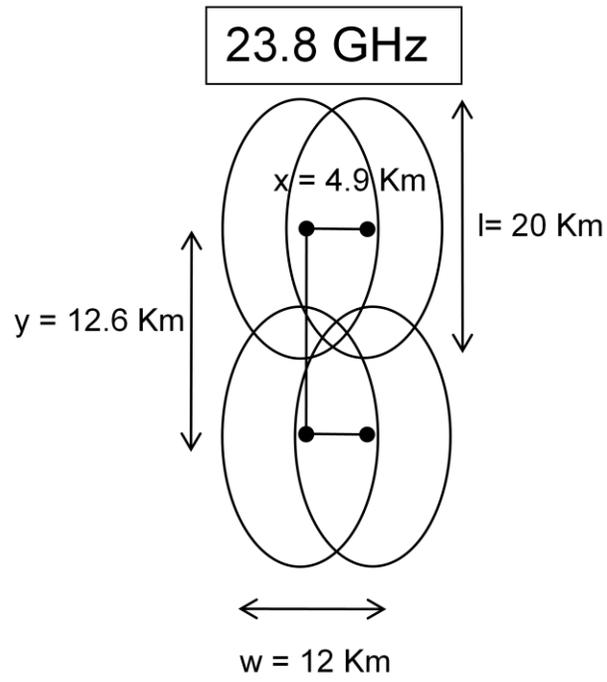


Figure 32: IDR Spatial Resolution at 23.8 GHz.

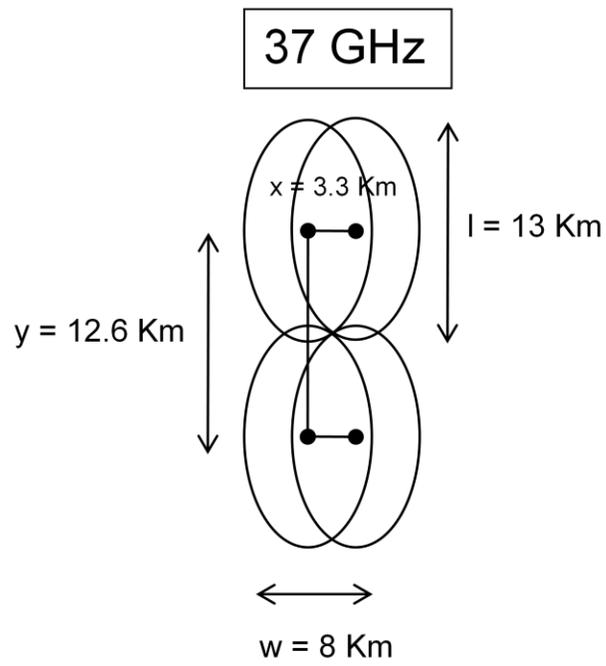


Figure 33: IDR Spatial Resolution at 37 GHz.

For each frequency and incidence angle, the number of IDR pixels falling into a rectangular IFOV, along-scan and along track, have been calculated using Eqn. (3-5) & Eqn. (3-6), and are shown in Table 7 & Figure 34:

Table 7: Rectangular IFOV Dimensions

Frequency	Incidence Angle	Width (Km)	Length (Km)	IDR Pixels (Across Track)	IDR Pixels (Along Track)
23.8 GHz	58°	31.6	57.8	5	4
	52°	26.7	45.2	4	3
37 GHz	58°	31.1	63.4	8	5
	52°	27.8	50.8	7	4

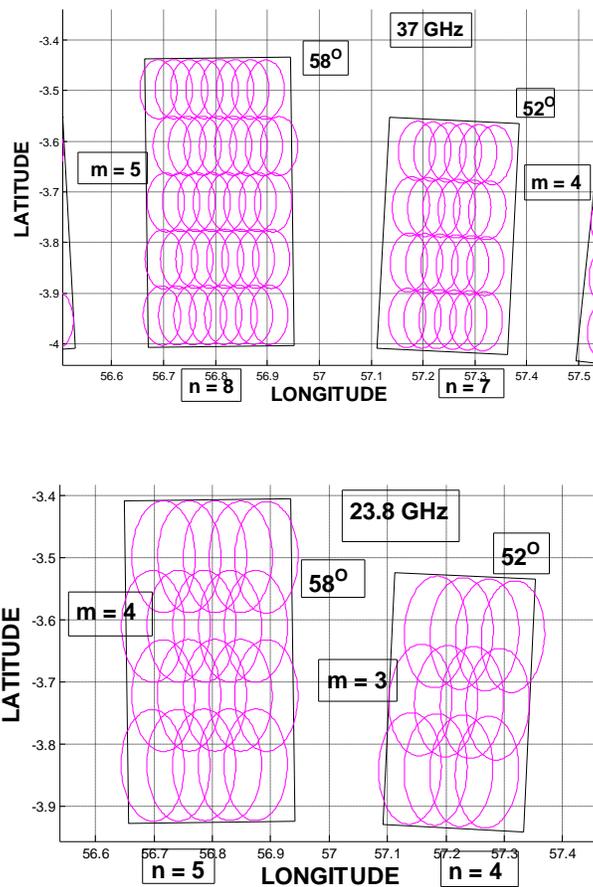


Figure 34: IDR beam-fills in Rectangular MWR IFOVs.

As mentioned in Chapter 2, the ground distance travelled by the Aquarius satellite, in a sampling time of 1.92 sec, is 13.1 Km(MWR inter-scan distance). The MWR scans will periodically skip an IDR scan because the IDR inter-scan distance (12.6 Km) is 0.5 Km less than the MWR inter-scan distance (13.1 Km). This skip will occur after ~ 25.2 MWR scans ($12.6 \text{ Km}/0.5 \text{ Km} = 25.2$), as shown in Figure 35.

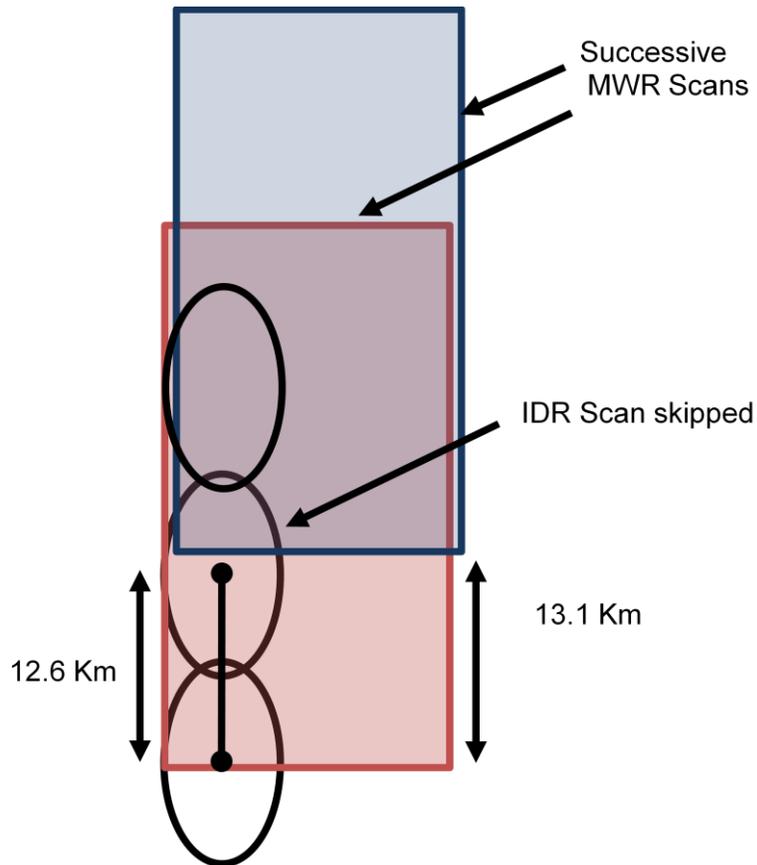


Figure 35: IDR Scan Skip.

The flow chart of MWR T_b simulation algorithm is shown in Figure 36.

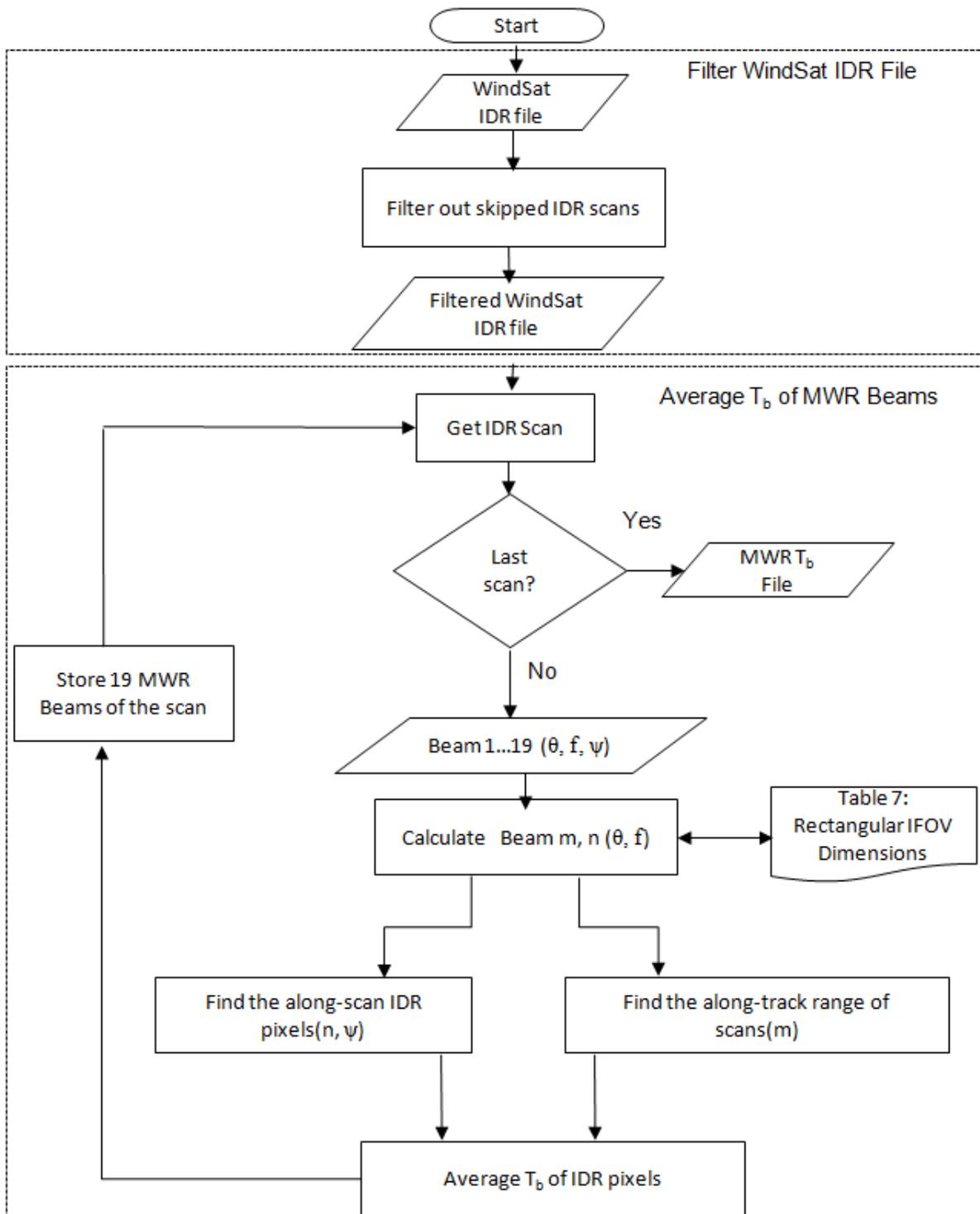


Figure 36: Flowchart of MWR Brightness Temperature Simulation Algorithm.

3.4 Normalization of WindSat Brightness Temperatures to MWR Incidence Angles

The ocean T_b measured by a microwave radiometer depends upon earth incidence angle and frequency among numerous other variables. This is because the ocean emissivity (Fresnel power reflection coefficient) is a function of the incidence angle and frequency (see Appendix B). WindSat and MWR have different earth incidence angles and at Ka band their frequencies differ by 0.5 GHz. Therefore, the T_b measurements between the two sensors will be different. To normalize WindSat's T_b s (averaged at MWR spatial resolution) to MWR T_b s, radiometric corrections must be applied.

A radiative transfer model (RTM) has been used to normalize WindSat T_b measurements to equivalent MWR frequencies and incidence angles. The RTM generated theoretical (rain-free) MWR and WindSat T_b s at their respective frequencies and incidence angles, using one week's (15-22 February, 2007) GDAS environmental parameter match-ups of wind speed, water vapor, sea surface temperature, and cloud liquid water at 1° spatial resolution. The differential "T_b bias" was calculated as the theoretical T_b difference between MWR and WindSat:

$$\Delta T_b = \text{MWR}_{\text{theoretical}} - \text{WindSat}_{\text{theoretical}} \quad (3-7)$$

The simulated MWR T_b was calculated by adding the T_b bias to the measured WindSat T_b :

$$T_{b \text{ MWR}} = \text{WindSat}_{\text{measured}} + \Delta T_b \quad (3-8)$$

3.4.1 Adjustments for Incidence Angle

Regression analyses were performed to relate theoretical T_b measurements of WindSat channels at 53° incidence angle against the corresponding theoretical T_b measurements of MWR channels at both 52° & 58° incidence angles. Bin averages were computed using the method of Hong [14], which finds the best-fit Gaussian curve and removes the outliers when calculating the mean and variance. Next linear regression analysis was applied to the bin averages, and finally, the difference between the linear fit and a 45° line was used to compute the T_b bias as a function of the WindSat theoretical T_b .

Figure 37 shows a T_b scatter diagram (density plot) with the result of the linear regression analysis using the binned averages of the independent variable, WindSat theoretical T_b , and the dependent variable, MWR theoretical T_b , at 23.8 GHz (V polarization) and 52° incidence angle.

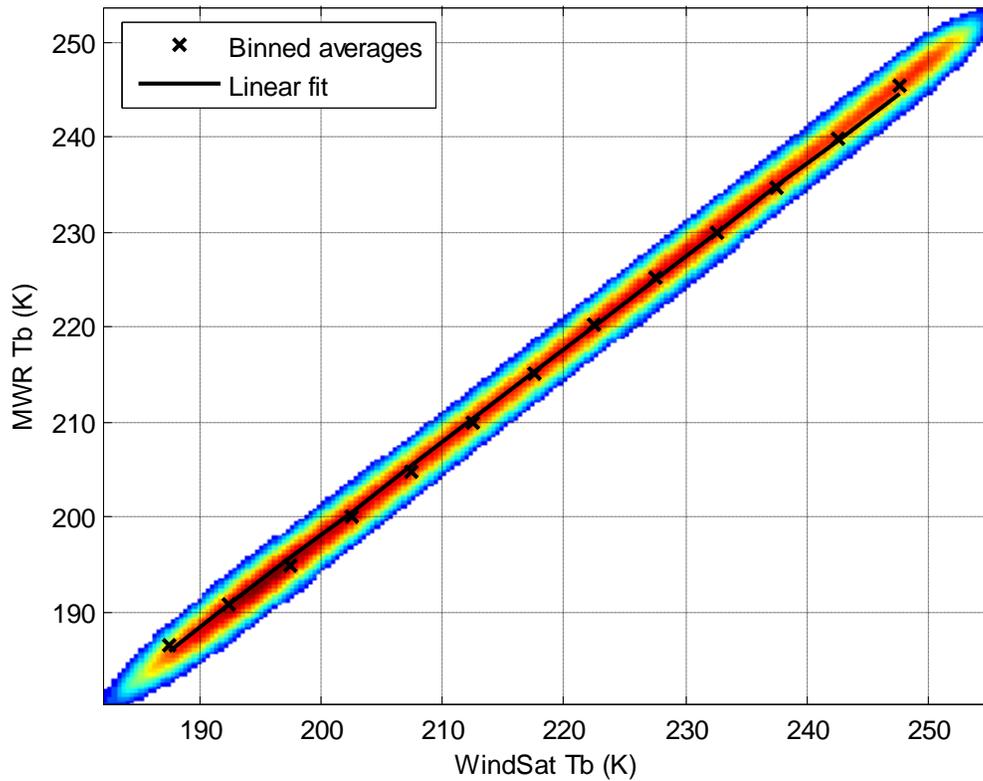


Figure 37: Density plot, binned averages & linear fit at 23.8 GHz V pol (52°).

Figure 38 shows that the linear fit has an offset and a small slope difference compared to the 45° line at 23.8 GHz (V polarization) and 52° incidence angle. To compute the radiometric bias, the equation of the 45° line is subtracted from that of the linear fit, and the result is a function of theoretical WindSat T_b :

$$\Delta y = -0.013 \times x + 0.5378 \quad (3-9)$$

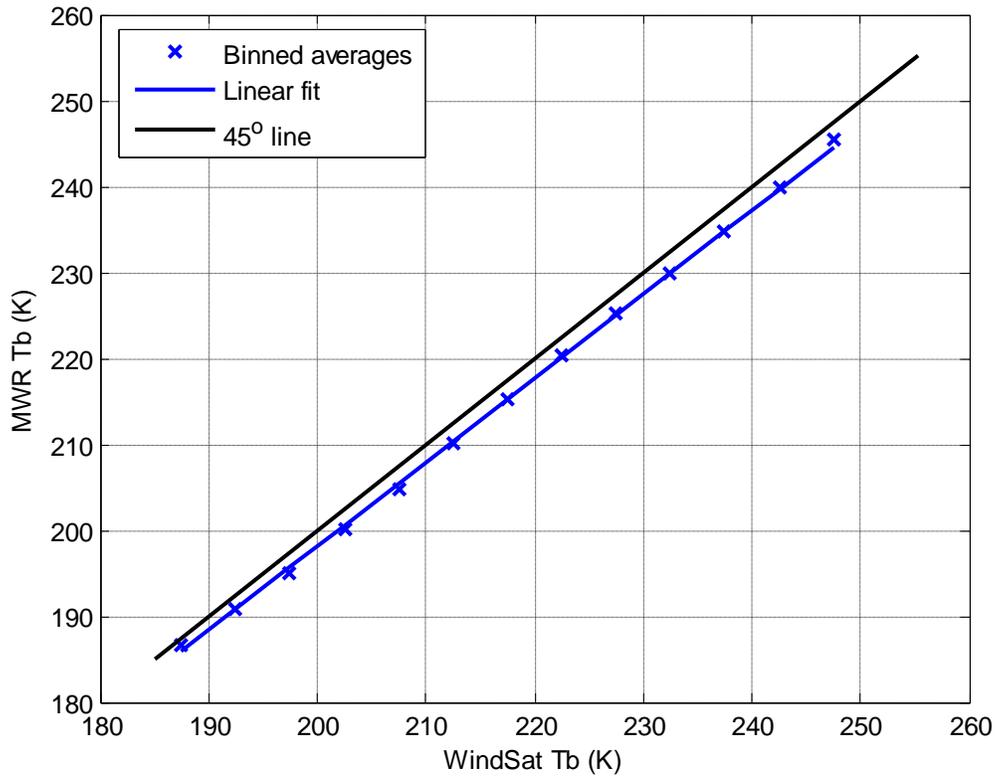


Figure 38: Comparison of Linear Fit to the 45° Line at 23.8 GHz V pol (52°).

Figure 39 and 40 illustrate similar results for the MWR theoretical T_b at 23.8 GHz (V polarization) and 58° incidence angle.

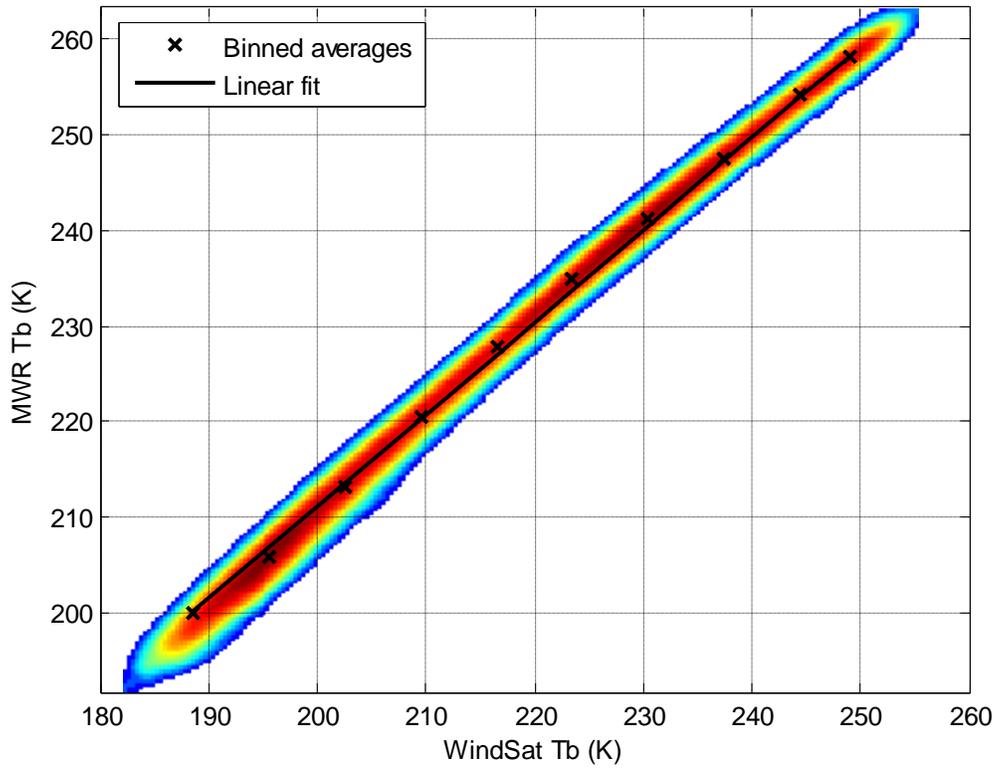


Figure 39: Density plot, binned averages & linear fit at 23.8 GHz V pol (58°).

Also, equation (3-10) shows that the resulting bias is a function of WindSat T_b :

$$\Delta y = -0.0379 \times x + 18.62 \quad (3-10)$$

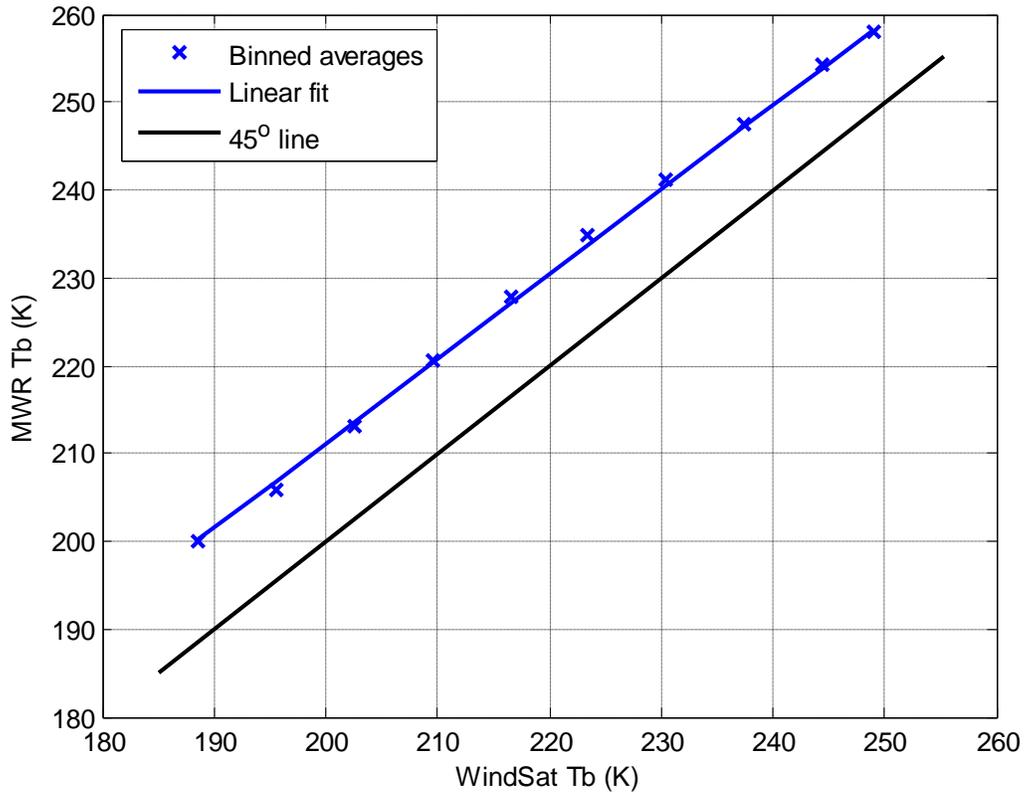


Figure 40: Comparison of Linear Fit to the 45° Line at 23.8 GHz V pol (58°).

In a similar manner, Figure 41 and 42 presents results for theoretical MWR T_b , at 36.5 GHz (V polarization) and 52° incidence angle.

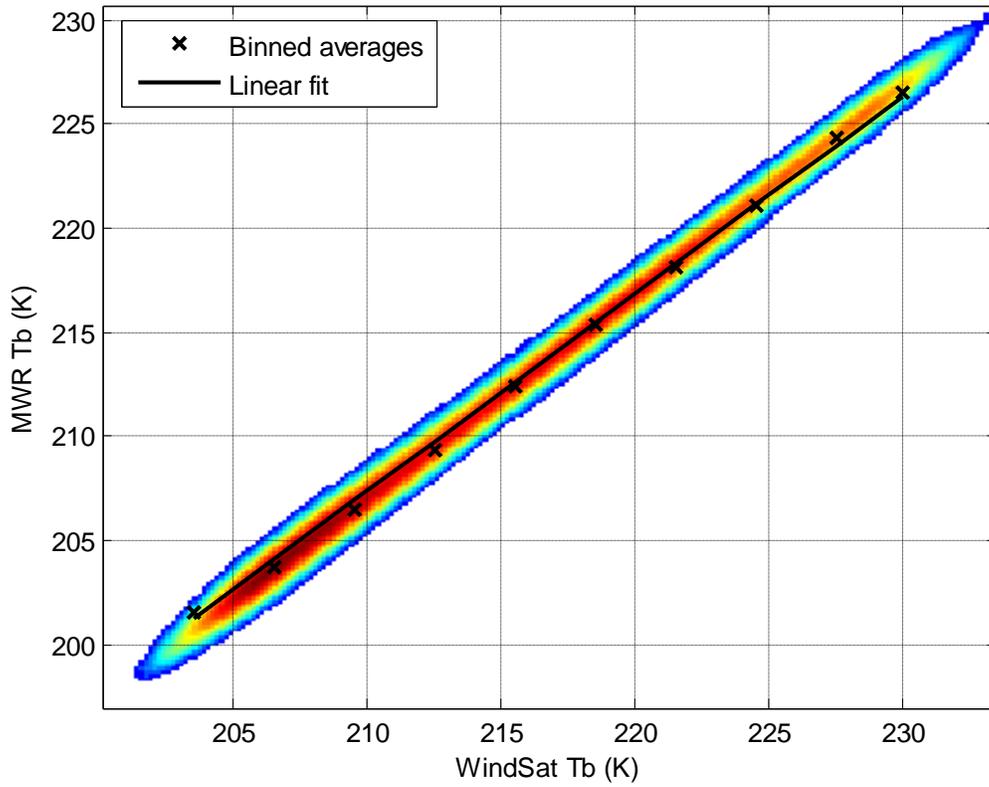


Figure 41: Density plot, binned averages & linear fit at 37 GHz V pol (52°).

Equation (3-11) shows the bias as a function of theoretical WindSat T_b :

$$\Delta y = -0.0425 \times x + 6.212 \quad (3-11)$$

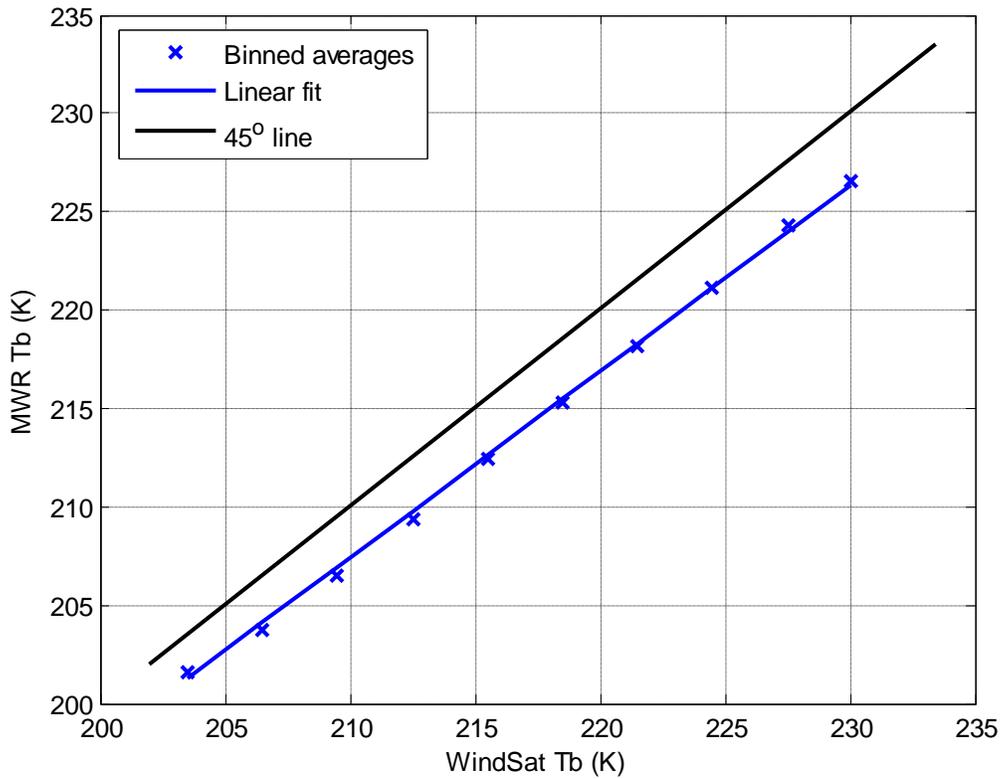


Figure 42: Comparison of Linear Fit to the 45° Line at 37 GHz V pol (52°).

In a similar manner, Figure 43 and 44 presents results for theoretical MWR T_b at 36.5 GHz (V polarization) and 58° incidence angle.

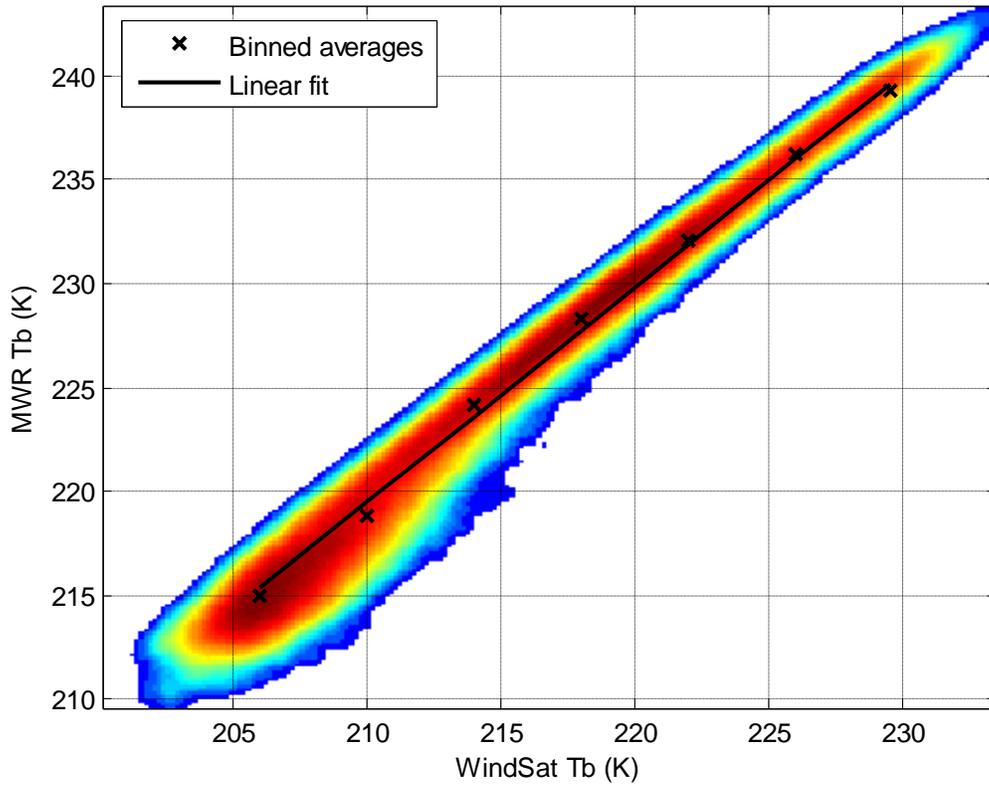


Figure 43: Density plot, binned averages & linear fit at 37 GHz V pol (58°).

Equation (3-12) shows the bias as a function of WindSat T_b :

$$\Delta y = 0.032 \times x + 2.825 \quad (3-12)$$

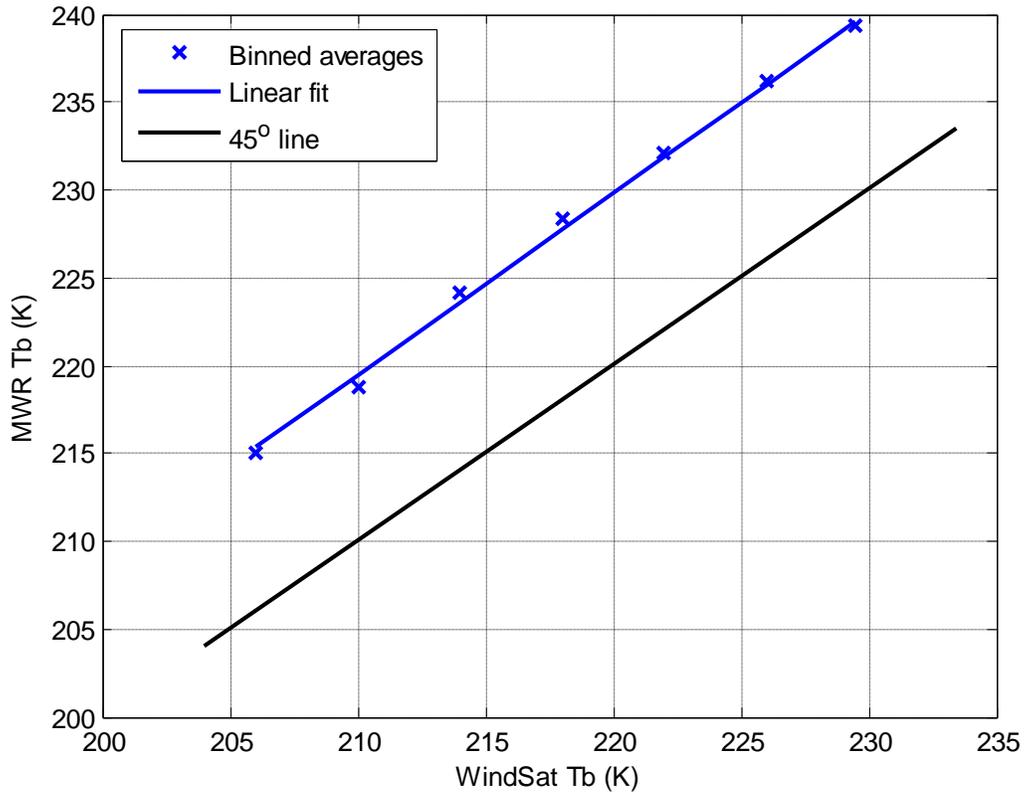


Figure 44: Comparison of Linear Fit to the 45° Line at 37 GHz V pol (58°).

In a similar manner, Figure 45 and 46 presents results for theoretical MWR T_b at 36.5 GHz (H polarization) and 52° incidence angle.

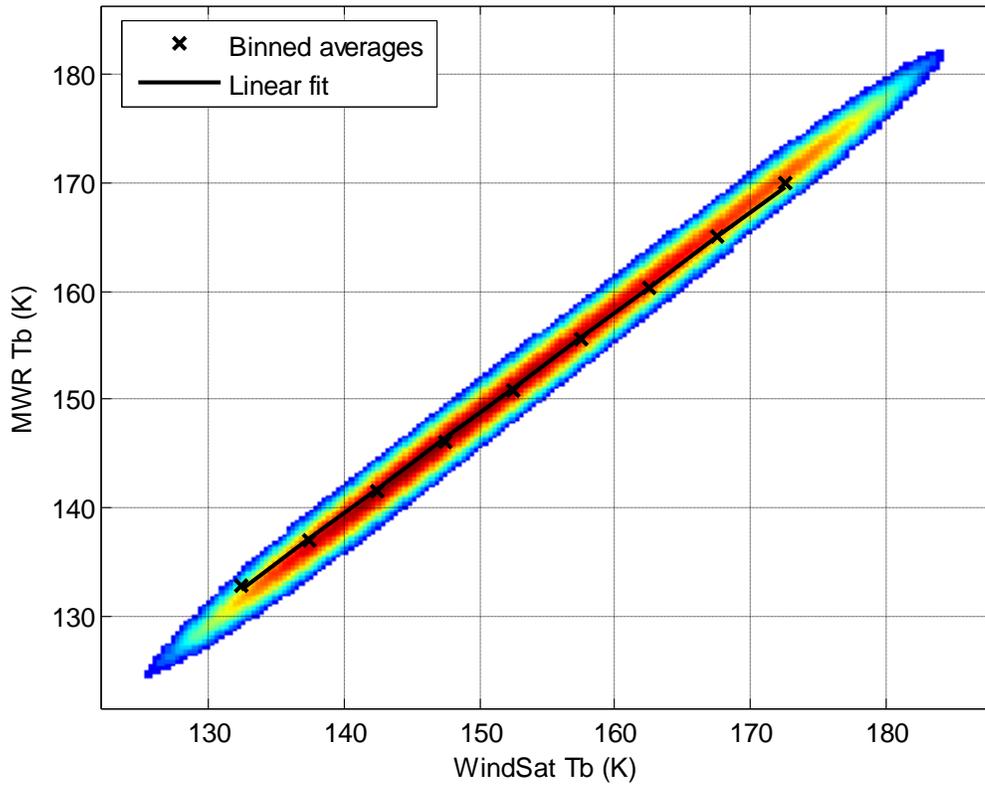


Figure 45: Density plot, binned averages & linear fit at 37 GHz H pol (52°)

Equation (3-13) shows the bias as a function of WindSat T_b :

$$\Delta y = -0.0738 \times x + 9.796 \quad (3-13)$$

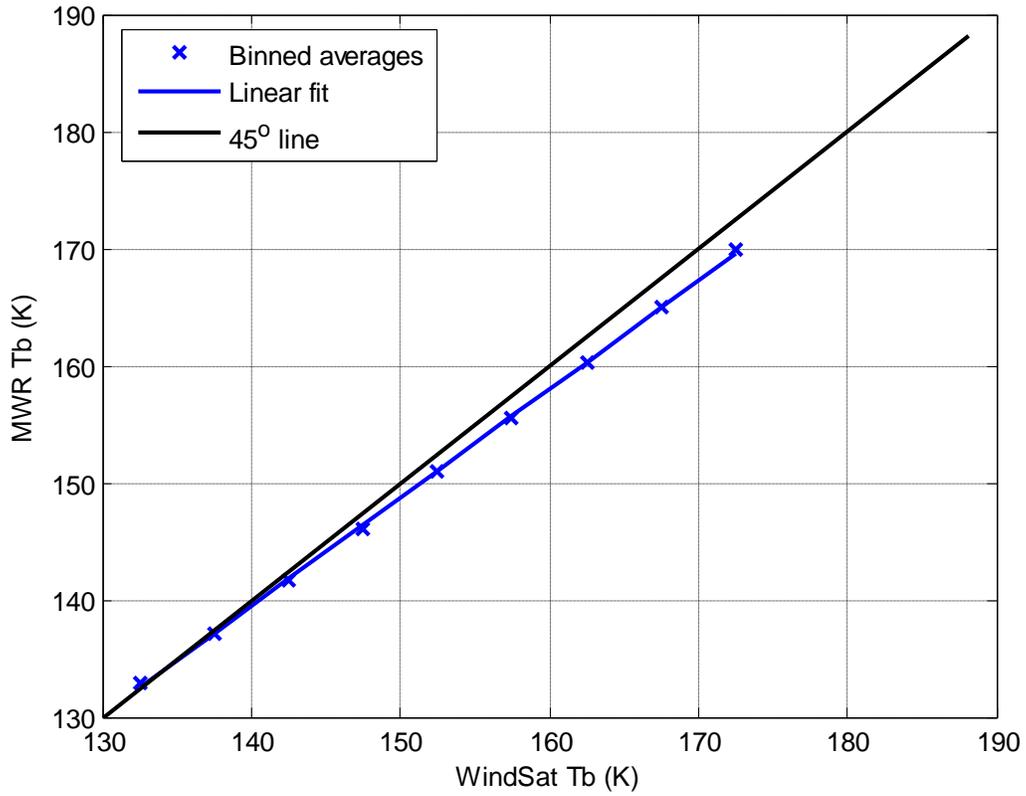


Figure 46: Comparison of Linear Fit to the 45° Line at 37 GHz H pol (52°).

In a similar manner, Figure 47 and 48 presents results for theoretical MWR T_b at 36.5 GHz (H polarization) and 58° incidence angle.

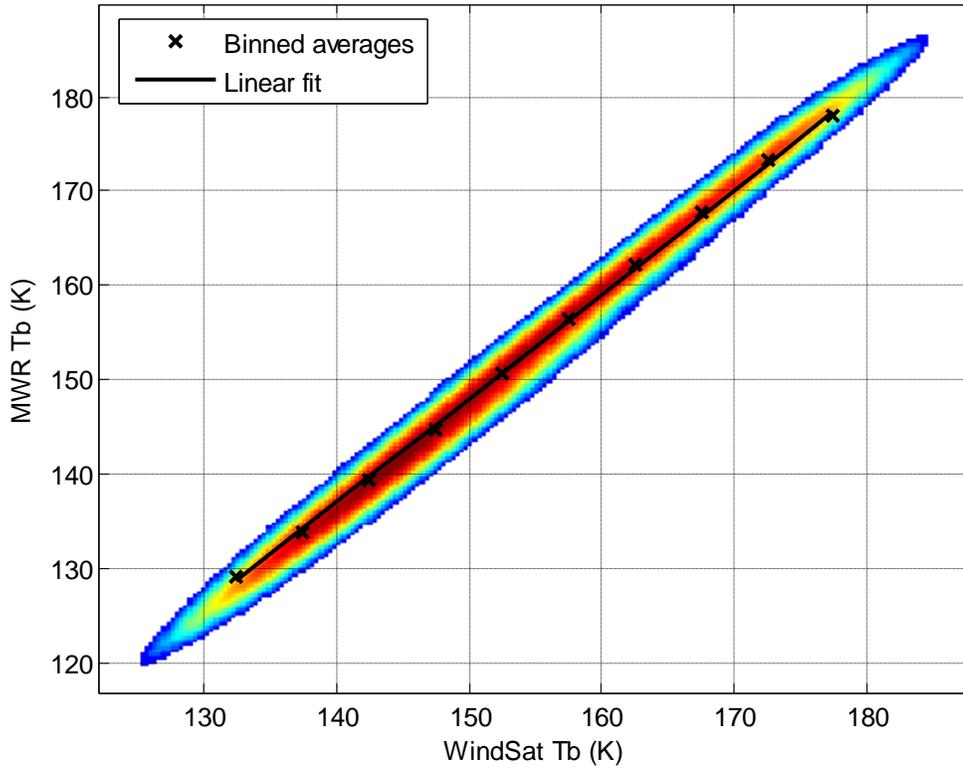


Figure 47: Density plot, binned averages & linear fit at 37 GHz H pol (58°).

Equation (3-14) shows the bias as a function of WindSat T_b :

$$\Delta y = 0.108 \times x - 18.25 \quad (3-14)$$

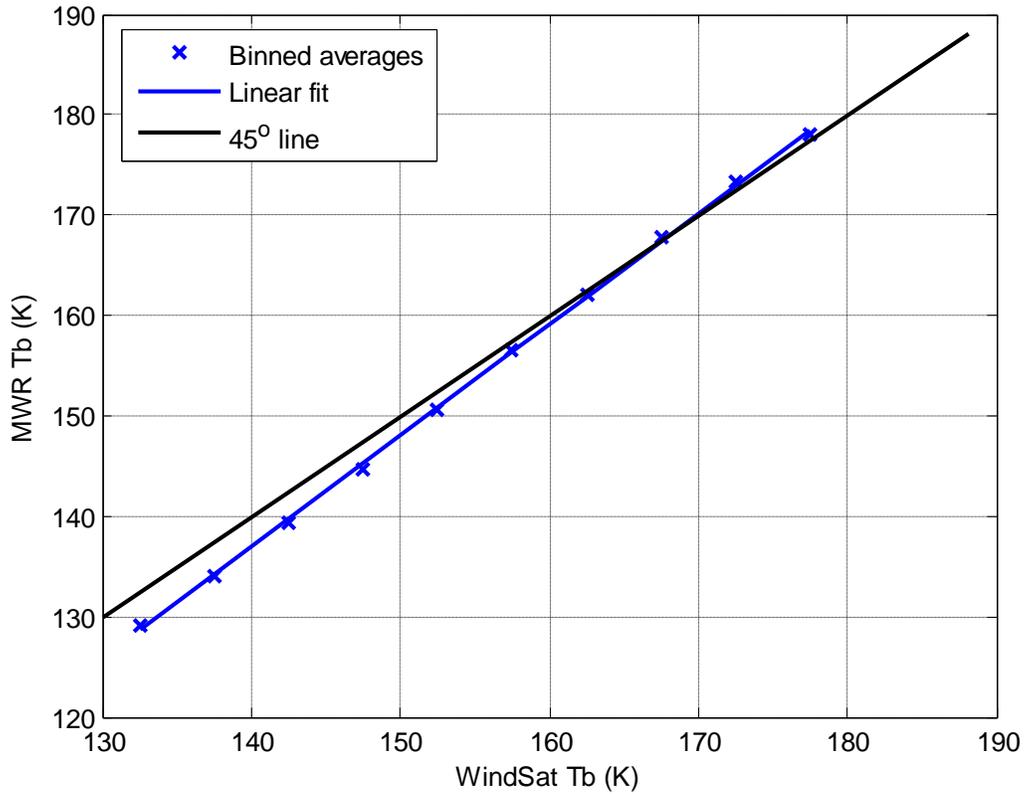


Figure 48: Comparison of Linear Fit to the 45° Line at 37 GHz H pol (58°).

3.4.2 Dependence on Environmental Parameters

The bias between the theoretical MWR and WindSat T_b s was plotted against collocated (1° resolution), geophysical parameter match-ups of GDAS (wind speed, water vapor, cloud liquid water and sea surface temperature) to examine the dependency of the T_b bias on environmental variables. Wind speeds greater than 15 m/s and water vapor greater than 60 mm were eliminated because of the difficulty in the RTM to accurately calculate brightness temperatures in this range. Also only SST's between 270 K and 305 K were considered. For all the channels, the

variation in the T_b bias due to environmental parameters was observed to be < 1 Kelvin, which is not significant for the MWR simulation.

Figure 49 shows the T_b bias plotted against wind speeds between 0 and 15 m/s. Also binned averages are shown as symbols, where each bin is a 2.5 m/s range of wind speed. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of wind speed.

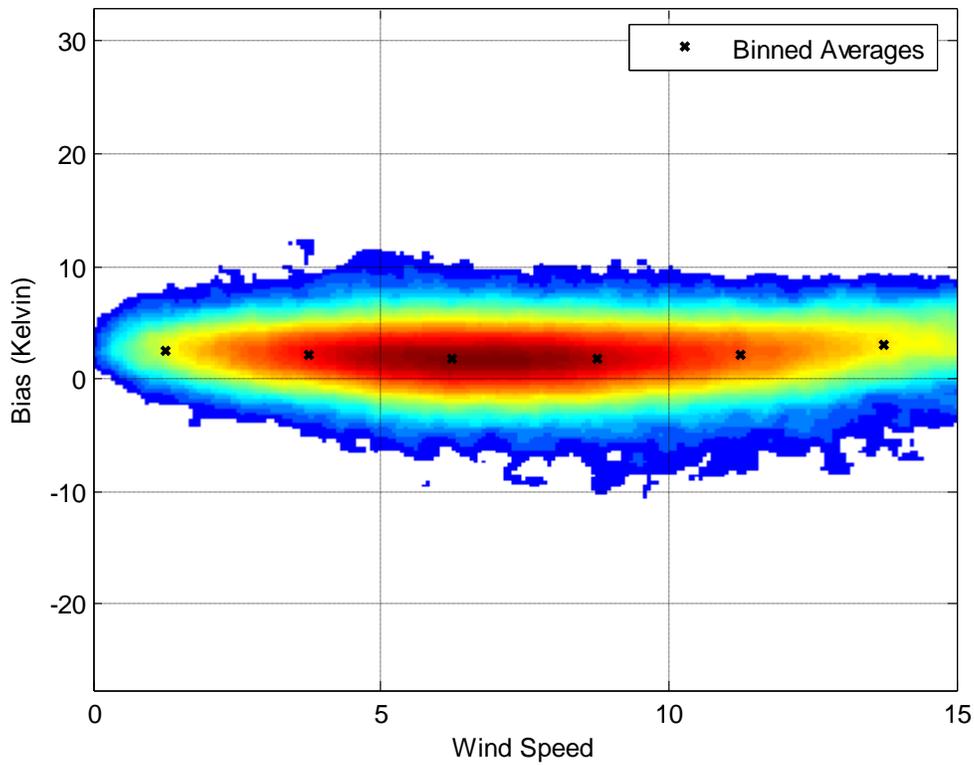


Figure 49: Dependence on wind speed, bias density plot & binned averages at 23.8 GHz V.

In a similar manner, Figure 50 shows the T_b bias plotted against water vapor between 0 and 60 mm. The binned averages are also shown in the figure as symbols where each bin is a 5 mm range of water vapor. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of water vapor.

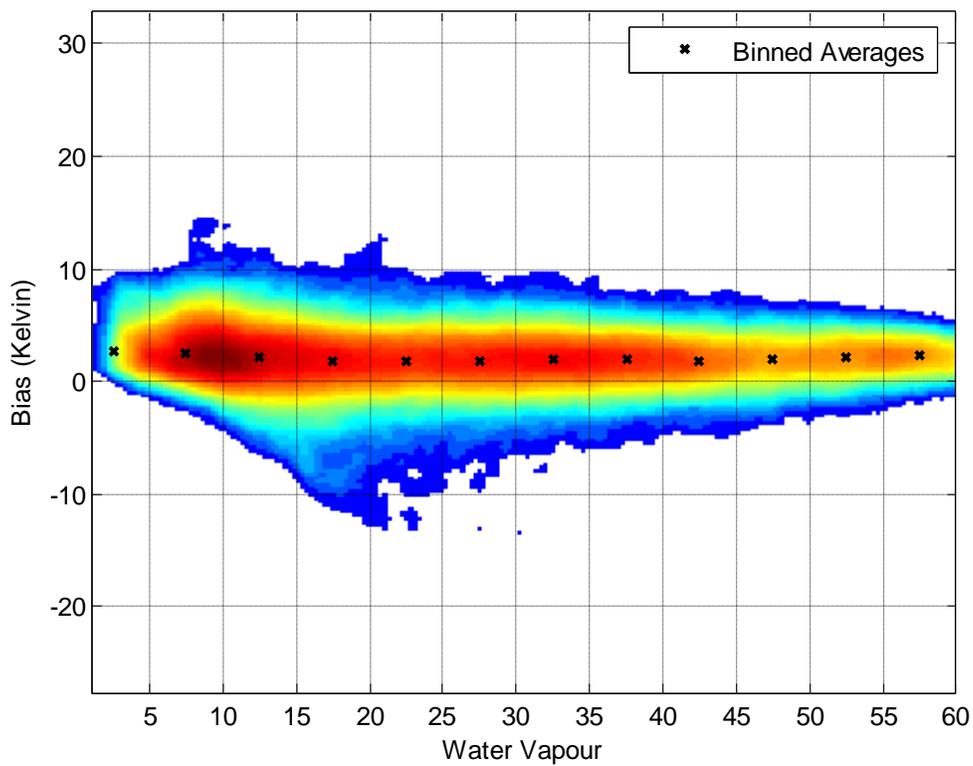


Figure 50: Dependence on water vapor, bias density plot & binned averages at 23.8 GHz V.

In a similar manner, Figure 51 shows the T_b bias plotted against SST between 270 and 305 Kelvin. The binned averages are also shown in the figure as symbols where each bin is a 5 K range of SST. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of SST.

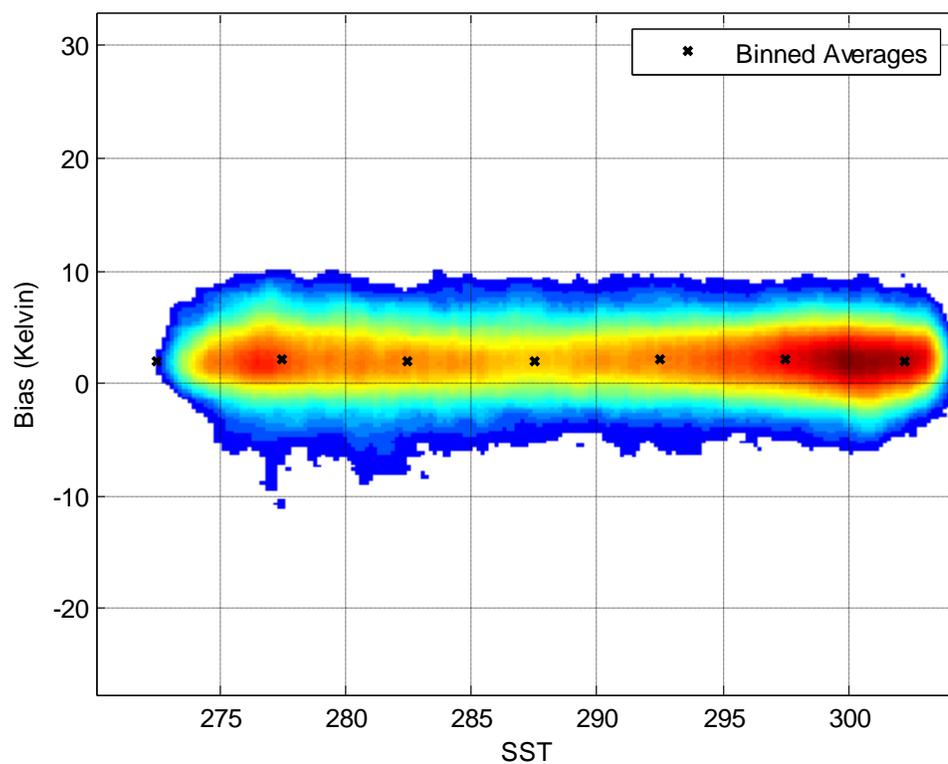


Figure 51: Dependence on SST, bias density plot & binned averages at 23.8 GHz V.

In a similar manner, Figure 52 shows the T_b bias plotted against CLW between 0 and 0.1 mm. The binned averages are also shown in the figure as symbols where each bin is a 0.01 mm range of CLW. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of CLW.

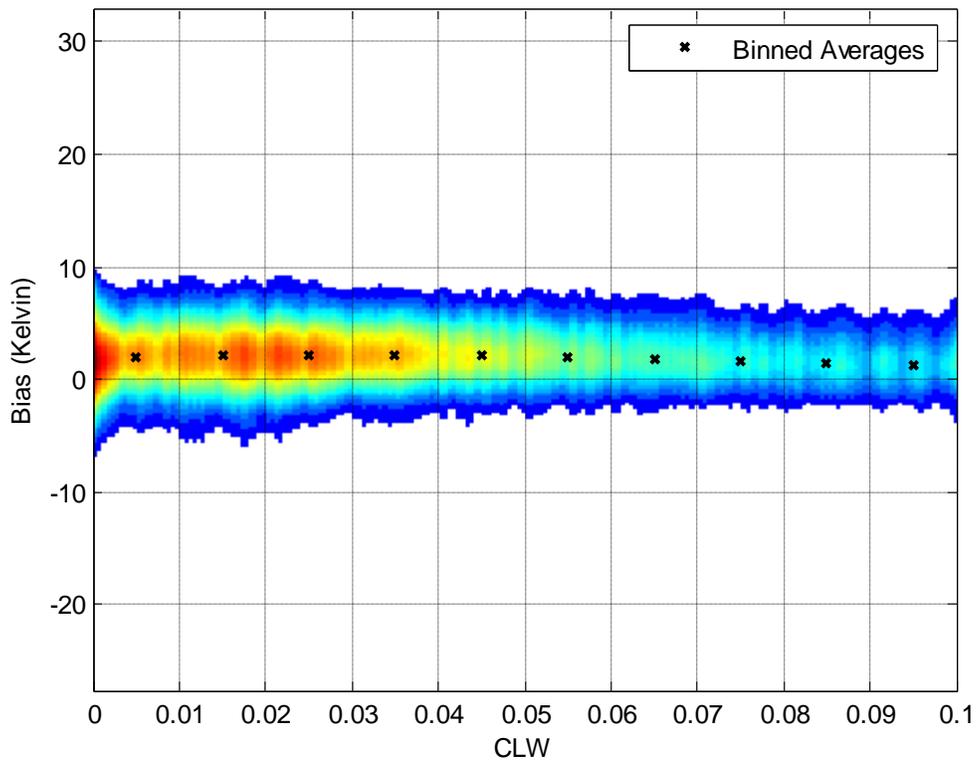


Figure 52: Dependence on CLW, bias density plot & binned averages at 23.8 GHz V.

In a similar manner, Figure 53 shows the T_b bias plotted against wind speeds between 0 and 15 m/s. Also binned averages are shown as symbols, where each bin is a 2.5 m/s range of wind speed. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of wind speed.

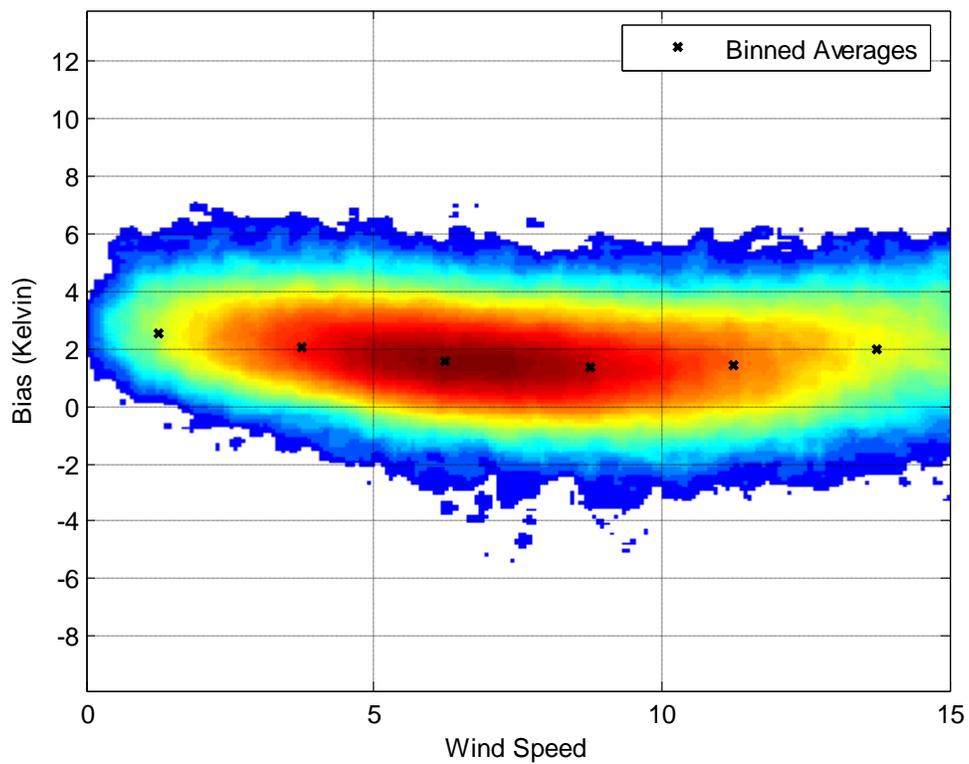


Figure 53: Dependence on wind speed, bias density plot & binned averages at 37 GHz V

In a similar manner, Figure 54 shows the T_b bias plotted against water vapor between 0 and 60 mm. The binned averages are also shown in the figure as symbols where each bin is a 5 mm range of water vapor. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of water vapor.

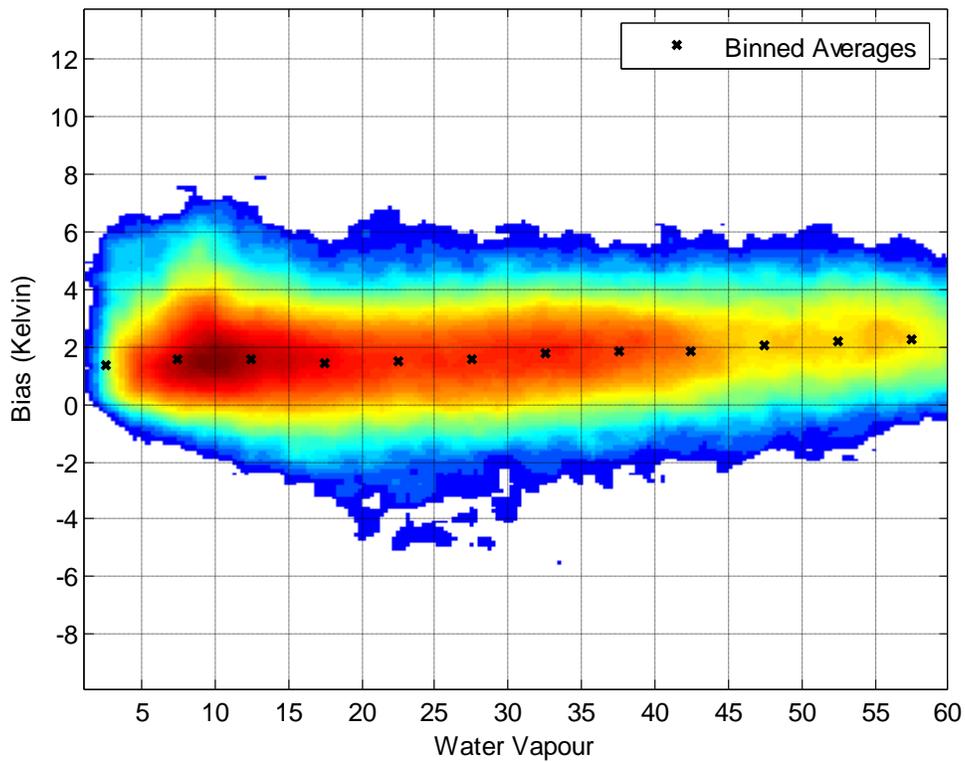


Figure 54: Dependence on water vapor, bias density plot & binned averages at 37 GHz V.

In a similar manner, Figure 55 shows the T_b bias plotted against SST between 270 and 305 Kelvin. The binned averages are also shown in the figure as symbols where each bin is a 5 K range of SST. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of SST.

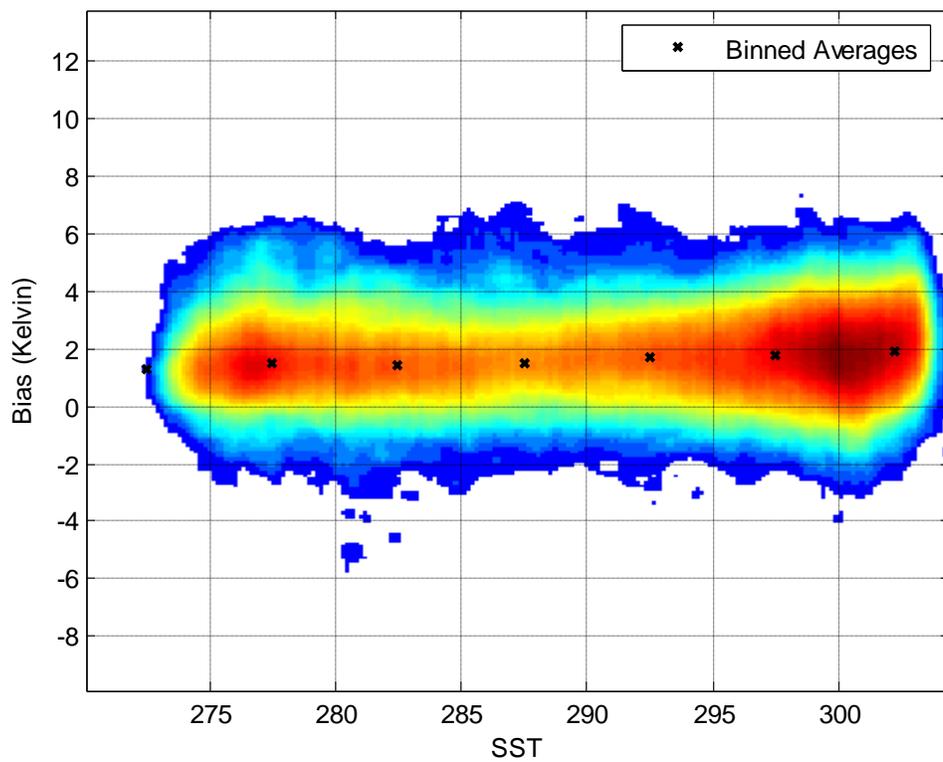


Figure 55: Dependence on SST, bias density plot & binned averages at 37 GHz V.

In a similar manner, Figure 56 shows the T_b bias plotted against CLW between 0 and 0.1 mm. The binned averages are also shown in the figure as symbols where each bin is a 0.01 mm range of CLW. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of CLW.

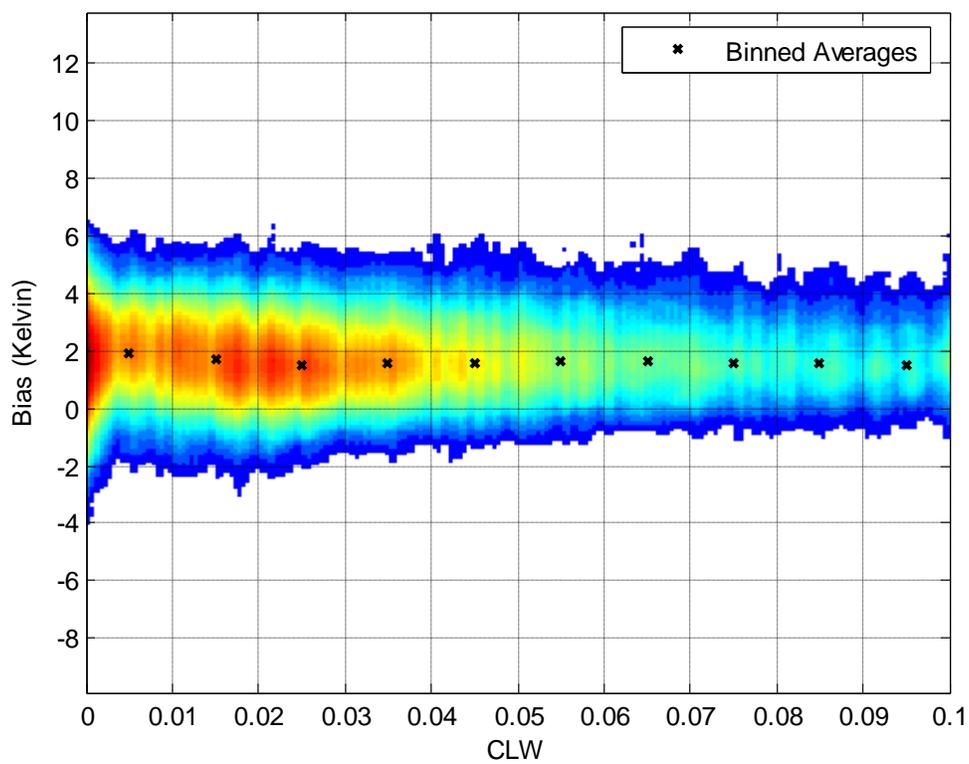


Figure 56: Dependence on CLW, bias density plot & binned averages at 37 GHz V.

In a similar manner, Figure 57 shows the T_b bias plotted against wind speeds between 0 and 15 m/s. Also binned averages are shown as symbols, where each bin is a 2.5 m/s range of wind speed. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of wind speed.

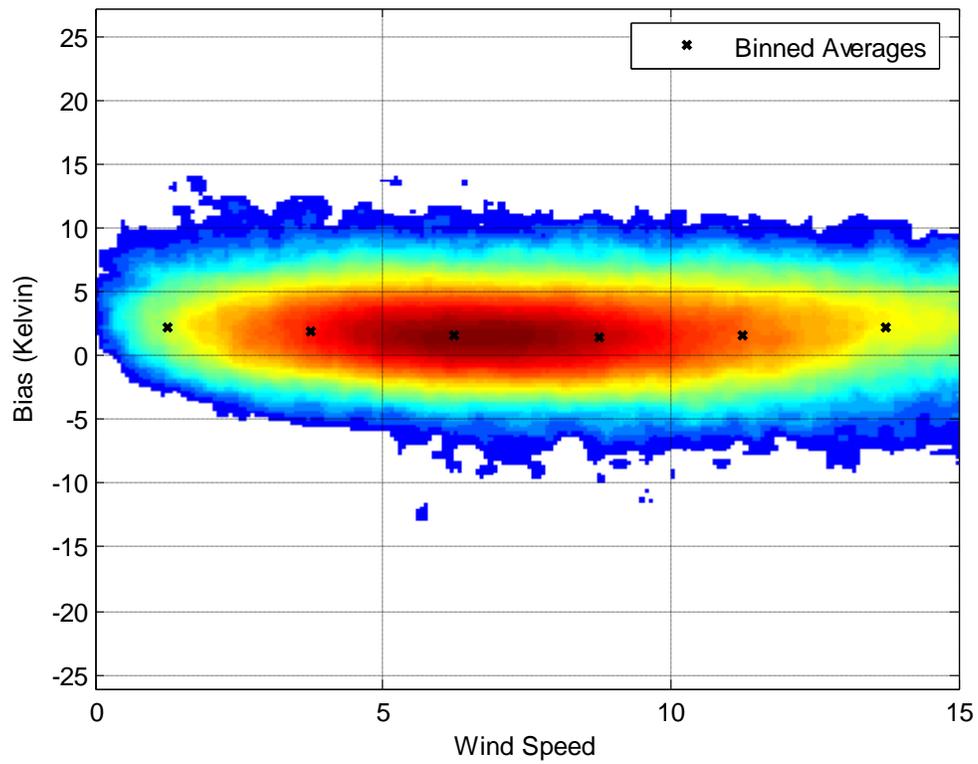


Figure 57: Dependence on wind speed, bias density plot & binned averages at 37 GHz H.

In a similar manner, Figure 58 shows the T_b bias plotted against water vapor between 0 and 60 mm. The binned averages are also shown in the figure as symbols where each bin is a 5 mm range of water vapor. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of water vapor.

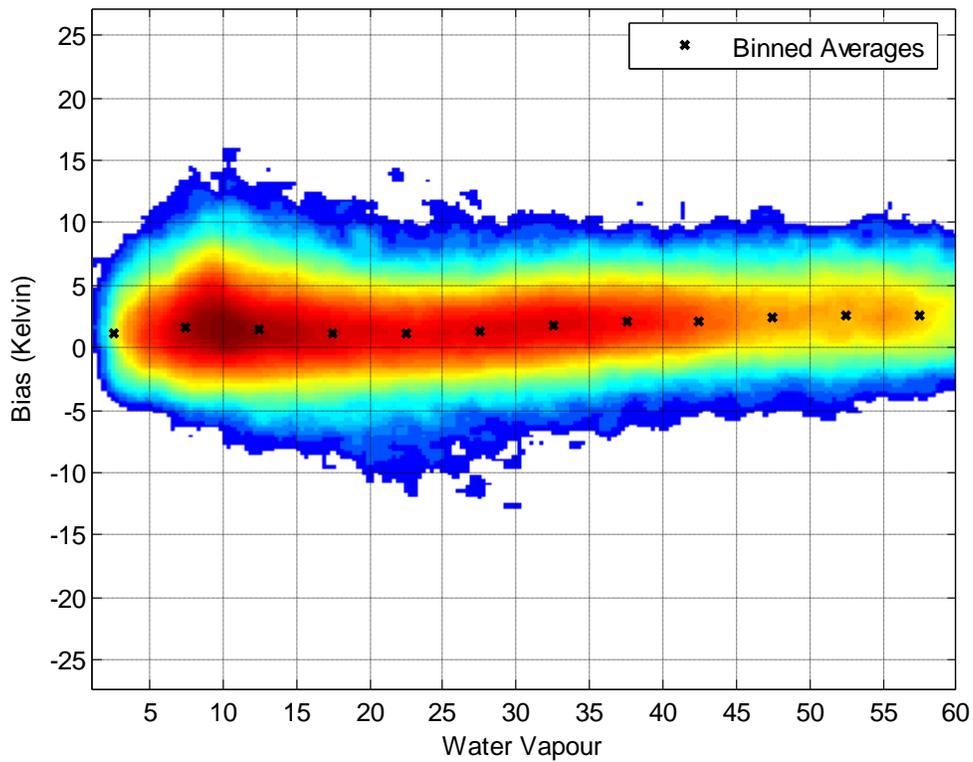


Figure 58: Dependence on water vapor, bias density plot & binned averages at 37 GHz H.

In a similar manner, Figure 59 shows the T_b bias plotted against SST between 270 and 305 Kelvin. The binned averages are also shown in the figure as symbols where each bin is a 5 K range of SST. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of SST.

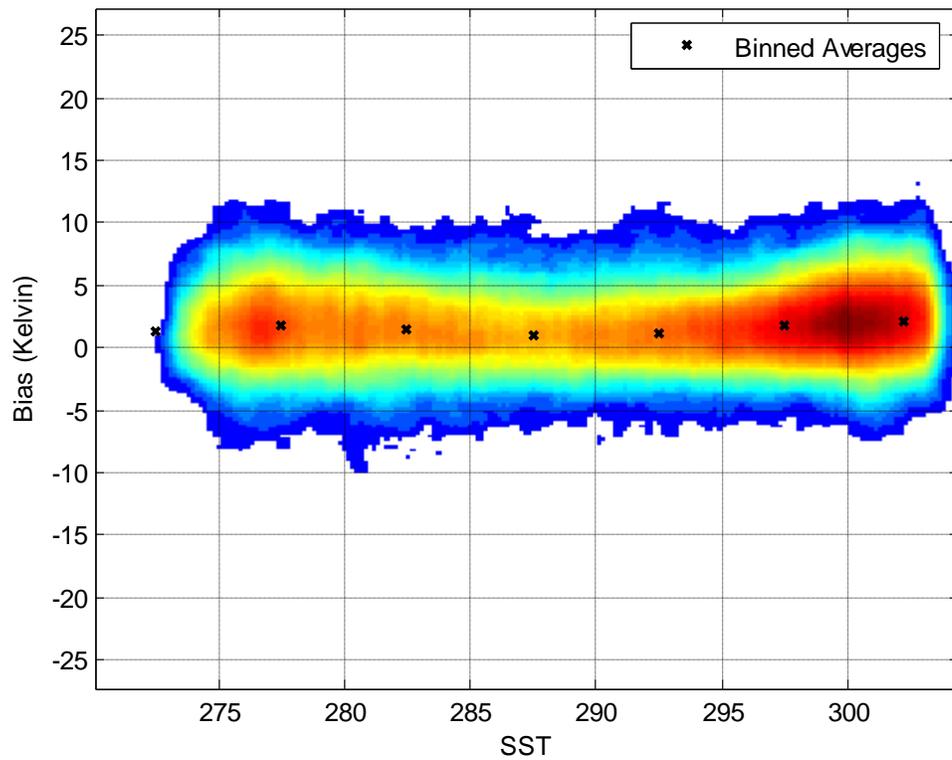


Figure 59: Dependence on SST, bias density plot & binned averages at 37 GHz H

In a similar manner, Figure 60 shows the T_b bias plotted against CLW between 0 and 0.1 mm. The binned averages are also shown in the figure as symbols where each bin is a 0.01 mm range of CLW. Because the plot is “flat” with near zero slope, this infers that the bias is approximately independent of CLW.

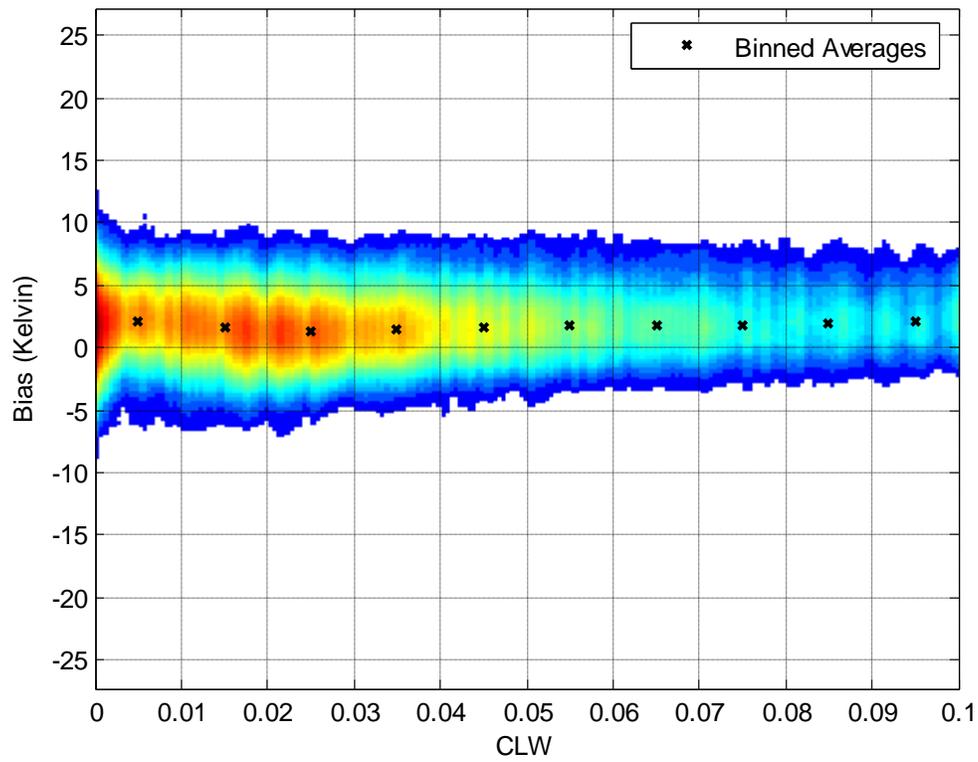


Figure 60: Dependence on CLW, bias density plot& binned averages at 37 GHz H.

CHAPTER 4. RESULTS AND VALIDATION

The simulated MWR dataset has been validated spatially/temporally and radiometrically. Spatial accurate images of the simulated MWR dataset have been plotted to show the IDR pixels (along scan and along track) averaging into an MWR pixel at each frequency and incidence angle. The along scan orientation of MWR pixels has been validated by presenting images of selective MWR scans for all nineteen beams depicting accurate along scan orientation by azimuth angle. A snapshot of MWR pixel collocation for the same scans between the two frequencies and the periodic skipping of IDR scans is also shown. The MWR dataset is radiometrically validated by comparing earth gridded T_b images of simulated MWR T_b s and WindSat IDR T_b s, without applying incidence angle corrections.

As explained in Chapter 2, the forward and aft beams of MWR need to be collocated through yaw steering. The methodology for the verification of CONAE's yaw steering and the corresponding results have also been included in this Chapter.

4.1 Spatial/Temporal Validation

The number of 37 GHz IDR pixels that average into MWR pixels at 52° & 58° incidence angles are 7×4 (along-scan x along-track) and 8×5 (along-scan x along-track), respectively as shown in Figure 61.

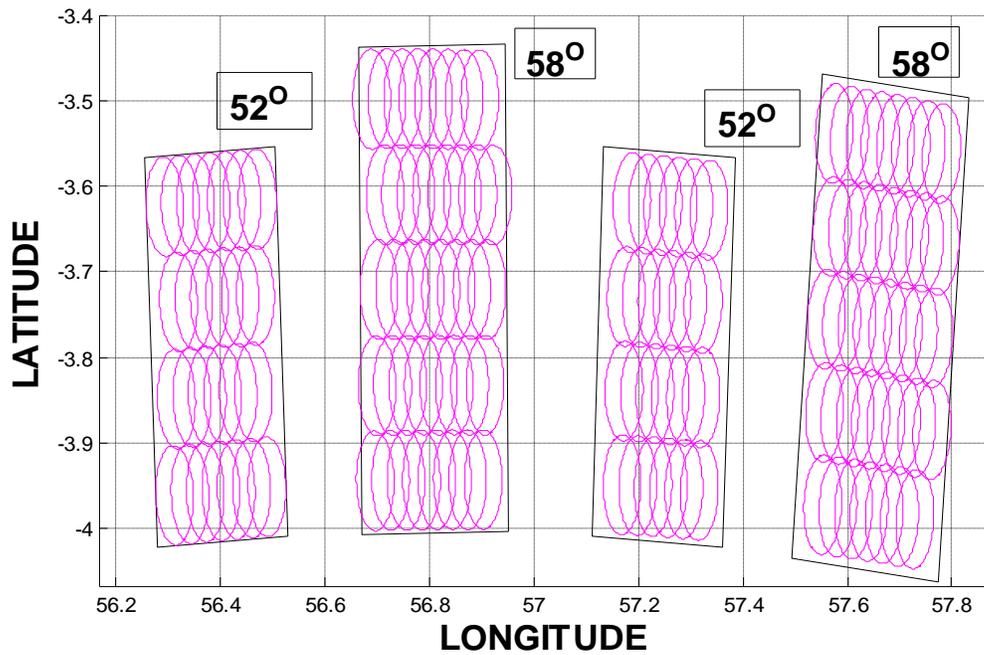


Figure 61: IDR Beam Fills in MWR Pixels at 36.5 GHz.

The number of 23.8 GHz IDR pixels that average into MWR pixels at 52° & 58° incidence angles are 4×3 (along-scan x along-track) and 5×4 (along-scan x along-track), respectively as shown in Figure 62.

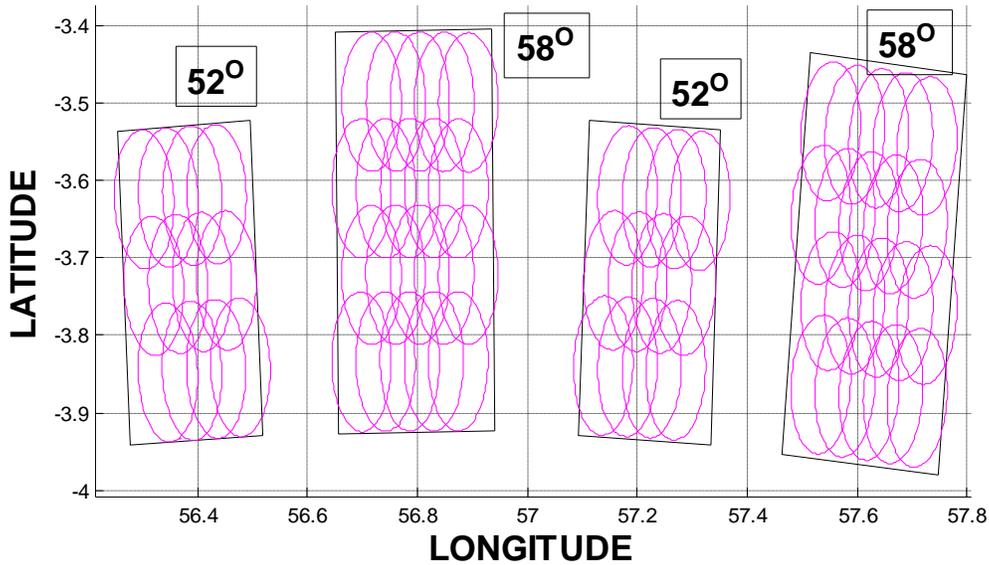


Figure 62: IDR Beam Fills in MWR Pixels at 23.8 GHz.

The K/Ka band simulated MWR scans are highly collocated and their pixel orientation changes gracefully with azimuth angle as shown in Figure 63. This figure shows three non-consecutive 23.8 (Green) and 36.5 (Red) GHz scans, which are inter-separated by six scans. The corresponding K/Ka band pixels collocate well at 52° and 58° incidence angles, with a smooth footprint orientation by azimuth angle.

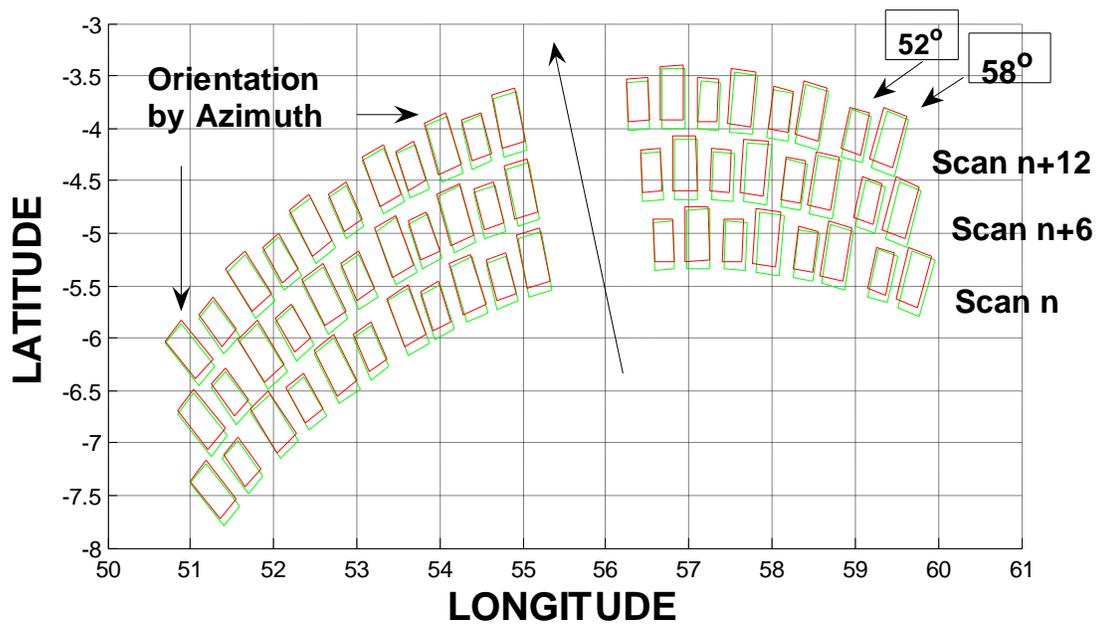


Figure 63: Collocated MWR 36.5 (Red) and 23.8 GHz (Green) Scans.

Figure 64 shows the sampling of simulated MWR beams in time. The red pixels are at 58° , while the green are at 52° incidence angle. There is a significant overlap between MWR scans because the beams travel only 13.1 Km during the sampling interval of 1.92 sec. Figure 64 also shows the skipping of IDR scans as periodic voids in the MWR scan progression. This skip occurs after ~ 25 MWR scans.

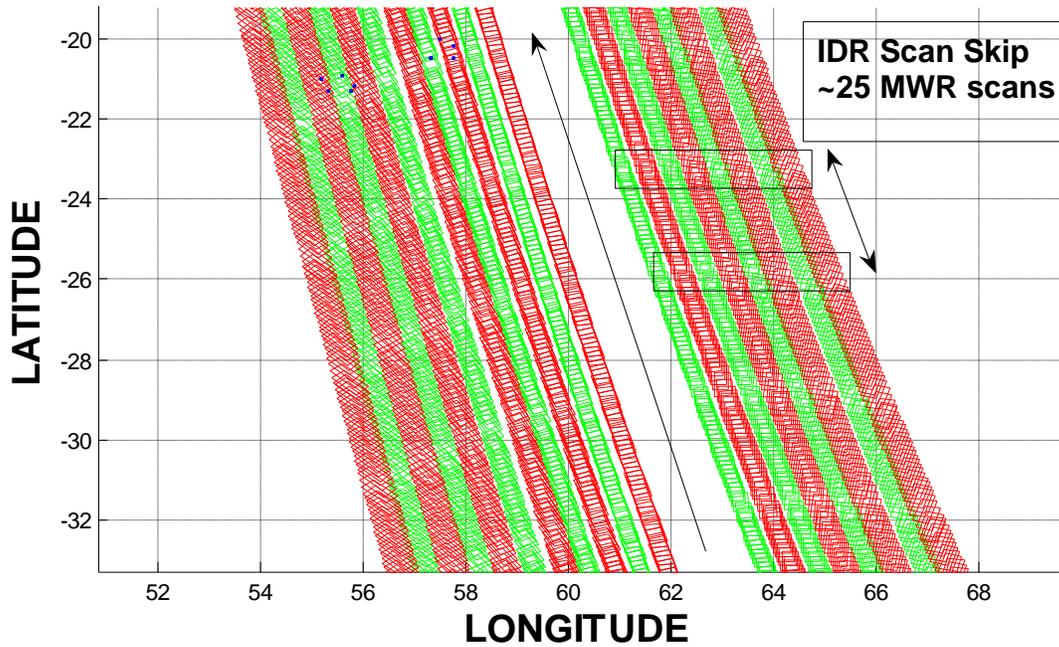


Figure 64: MWR 36.5 GHz Swath at 58° (Red) & 52° (Green).

4.2 Radiometric Validation

Figure 65 shows the comparison between earth gridded T_b s of IDR and MWR at 37 GHz (V pol) for 68 orbits in February 2007, without applying incidence angle corrections. The earth gridded T_b images of WindSat IDR and simulated MWR are shown at the top and bottom, respectively. The T_b s have been averaged in 0.5° resolution boxes. The color scale of both the images is the same. Notice the radiometric accuracy of the T_b simulation as the two T_b images show high similarity.

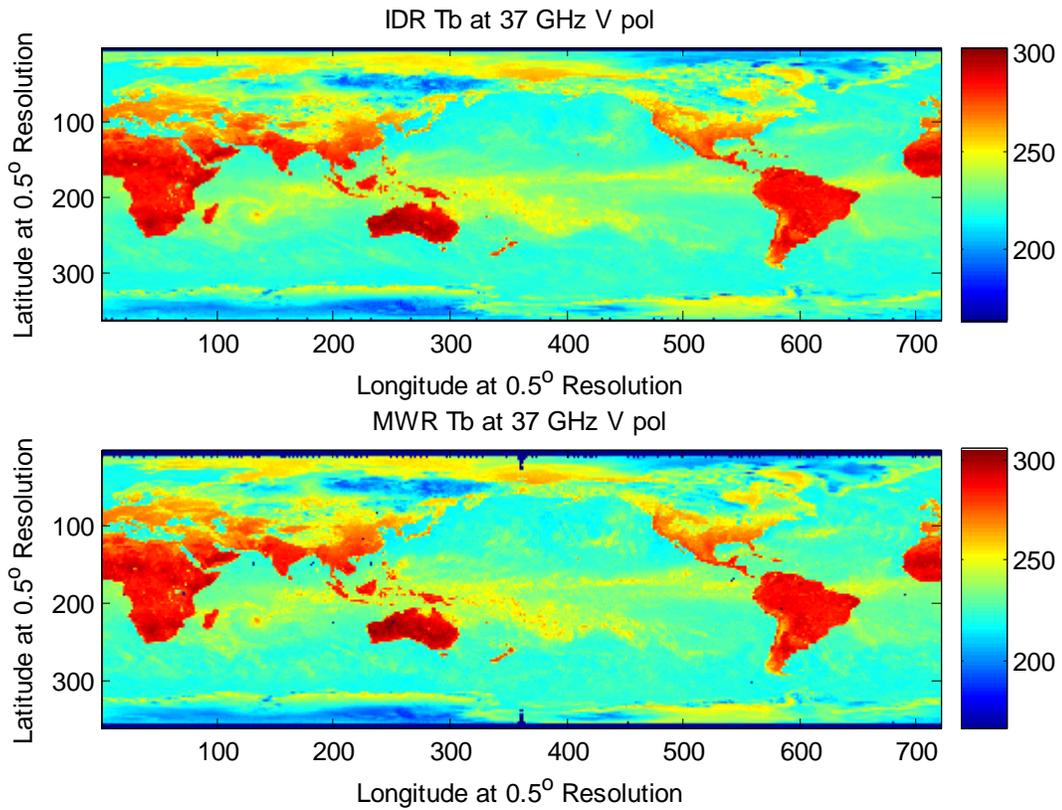


Figure 65: Earth Gridded T_b s at 37 GHz V pol (53°).

Figure 66 shows the T_b difference between the two earth gridded T_b images of Figure 65. The mean T_b difference between the two images is ~ 0 Kelvin with a standard deviation of ~ 2 Kelvin. The areas of higher T_b differences, though very sparse, are due to the difference in the spatial resolution of MWR and IDR pixels.

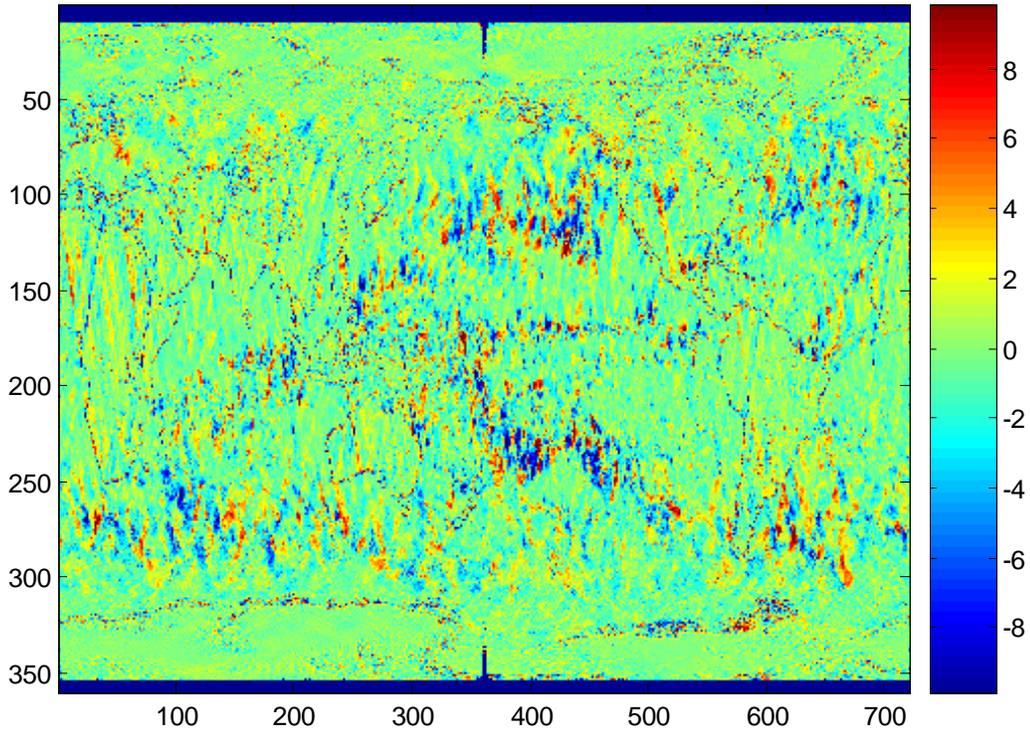


Figure 66: Earth Gridded T_b Difference at 37 GHz V pol (53°).

4.3 Yaw Steering Results

The yaw steering required for MWR forward and aft beam collocation has been provided by CONAE as a function of the sub-satellite point's latitude. Figure 67 shows the absolute yaw in degrees as a function of latitude. The absolute yaw is maximum at the equator and minimum at the poles since the angular rotation of the Earth is maximum at the equator and minimum at the poles. During the ascending and descending halves of the orbit, the yaw remains the same in magnitude but its polarity changes.

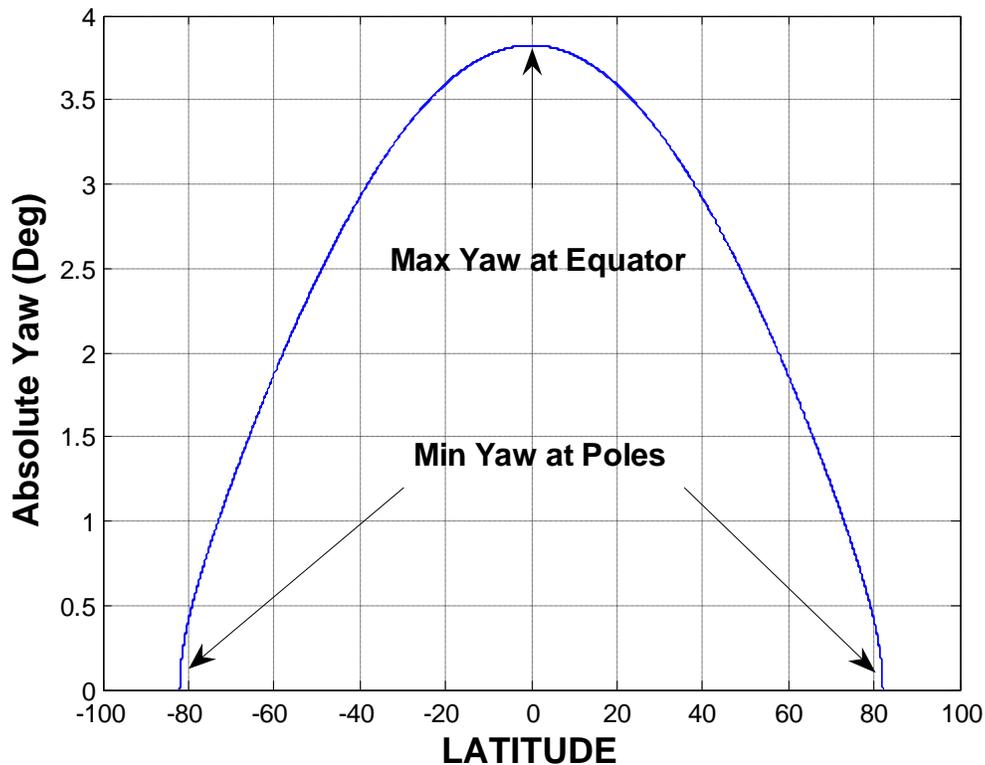


Figure 67: Absolute Yaw vs Latitude.

In order to verify MWR's yaw steering, sub-satellite points and velocity azimuth angles were generated for one orbit with an interval of 0.24 sec using Satellite Toolkit (STK) software. Beam bore-sight intersection points were generated at the same interval (0.24 sec), sequentially for the 8 forward/aft beam pairs. As explained in Chapter 2, the integration time for each MWR forward and aft beam pair is 0.24 sec resulting in a sampling time of $0.24 \times 8 = 1.92$ sec. Therefore, the first forward/aft beam pair is sampled at 0 sec, the second pair at 0.24 sec, the third pair at 0.48 sec and so on till the first pair is re-sampled at 1.92 sec.

Beam boresight intersections were generated using Matlab's 'Reckon' method. The inputs to this method are sub-satellite point's latitude and longitude, the arc length in degrees from sub-satellite point to the beam, and the compass azimuth angle of the beam. The outputs are the latitude and longitude of beam boresight. As mentioned earlier, the sub-satellite points were generated using STK. The arc-length from sub-satellite point to the beam was easily calculated using incidence plane geometry described in Appendix A. The compass azimuth angle of each beam was calculated using the satellite's velocity azimuth and beam azimuth relative to the flight direction. Figure 68 shows satellite velocity azimuth (anti-clockwise relative to North) and satellite velocity compass azimuth angle (clockwise relative to North).

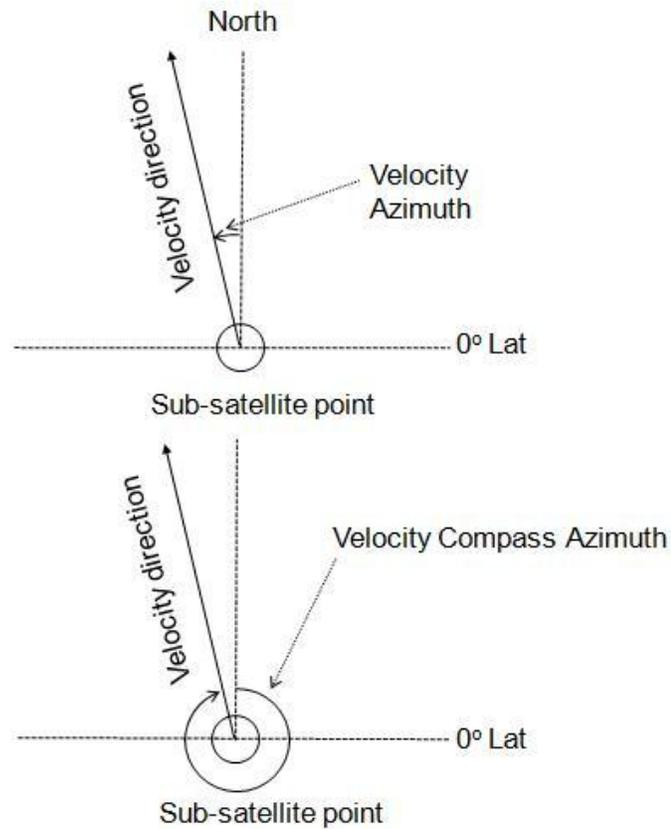
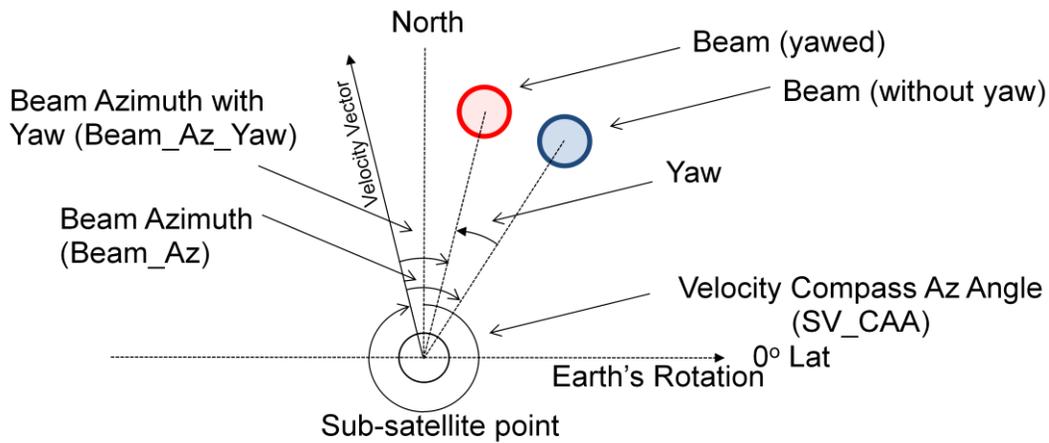


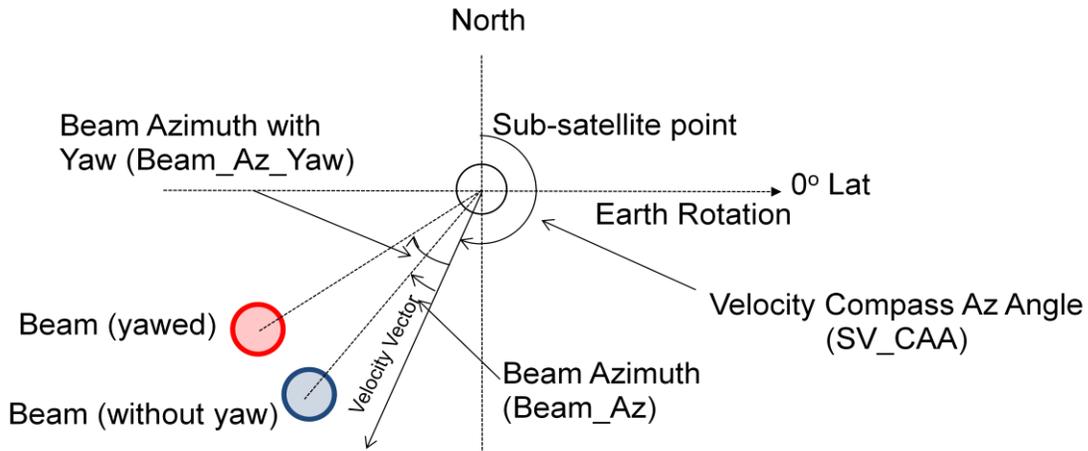
Figure 68: Satellite Velocity Azimuth.

The yawed beam's compass azimuth angle can be calculated by the addition of the satellite's velocity compass azimuth angle, the fixed antenna beam azimuth angle relative to flight direction, the yaw, and computing the modulo of the addition with 360° . Figure 69 and Figure 70 show the geometry for the calculation of yawed beam's compass azimuth angle for ascending and descending flight direction, respectively.



$$\text{Beam_CAA_Yaw} = \text{modulo} ((\text{SV_CAA} + \text{Beam_Az} + \text{Yaw}), 360)$$

Figure 69: Beam Compass Azimuth Angle for Ascending Flight.



$$\text{Beam_CAA_Yaw} = \text{modulo} ((\text{SV_CAA} + \text{Beam_Az} + \text{Yaw}), 360)$$

Figure 70: Beam Compass Azimuth Angle for Descending Flight.

The first and the eighth beam boresight points were generated for one orbit, taking into account the yaw steering provided by CONAE. The distance (Km) between the corresponding collocated

forward/aft beam pairs were calculated. Figure 71 shows the scenario at an ascending and descending node. The collocation separation of the forward/aft beam 1 is ~6 Km for both the ascending and descending node while for beam 8 it is ~2.2 Km for the ascending node and ~6.1 Km for the descending node.

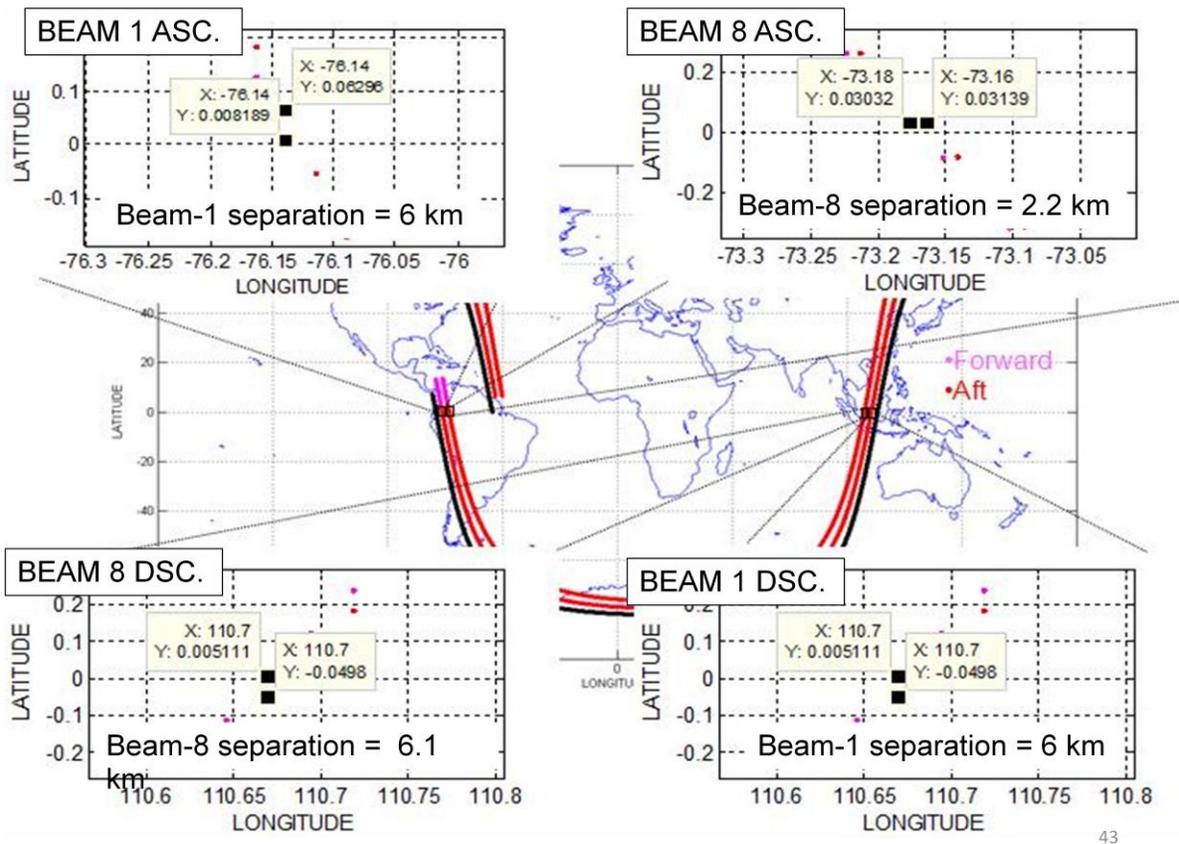


Figure 71: Beam 1 & 8 Collocation Separation at Ascending & Descending Node.

Figure 72 shows the collocation separation of the forward/aft beam 1 as a function of latitude, for one orbit, with ascending and descending collocation distinguishable by color. A mean collocation separation of ~4 Km has been observed.

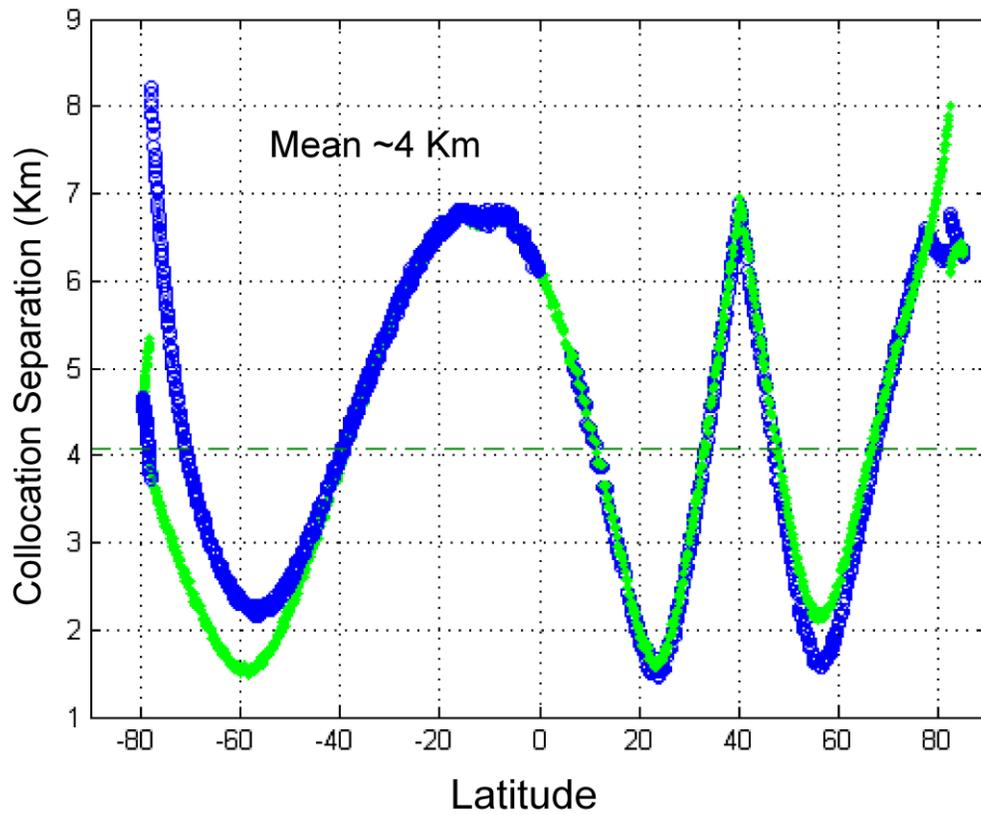


Figure 72: Forward/Aft Beam 1 Collocation Separation.

In a similar manner, Figure 73 shows the collocation separation of the forward/aft beam 2.

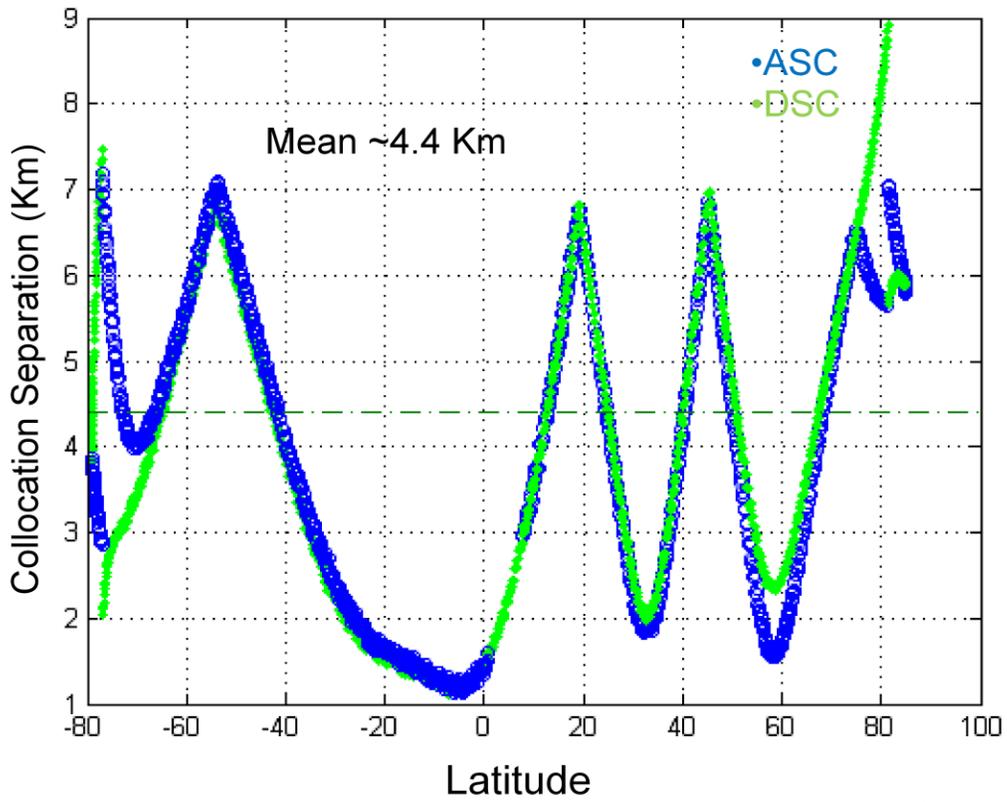


Figure 73: Forward/Aft Beam 2 Collocation Separation.

In a similar manner, Figure 74 shows the collocation separation of the forward/aft beam 3.

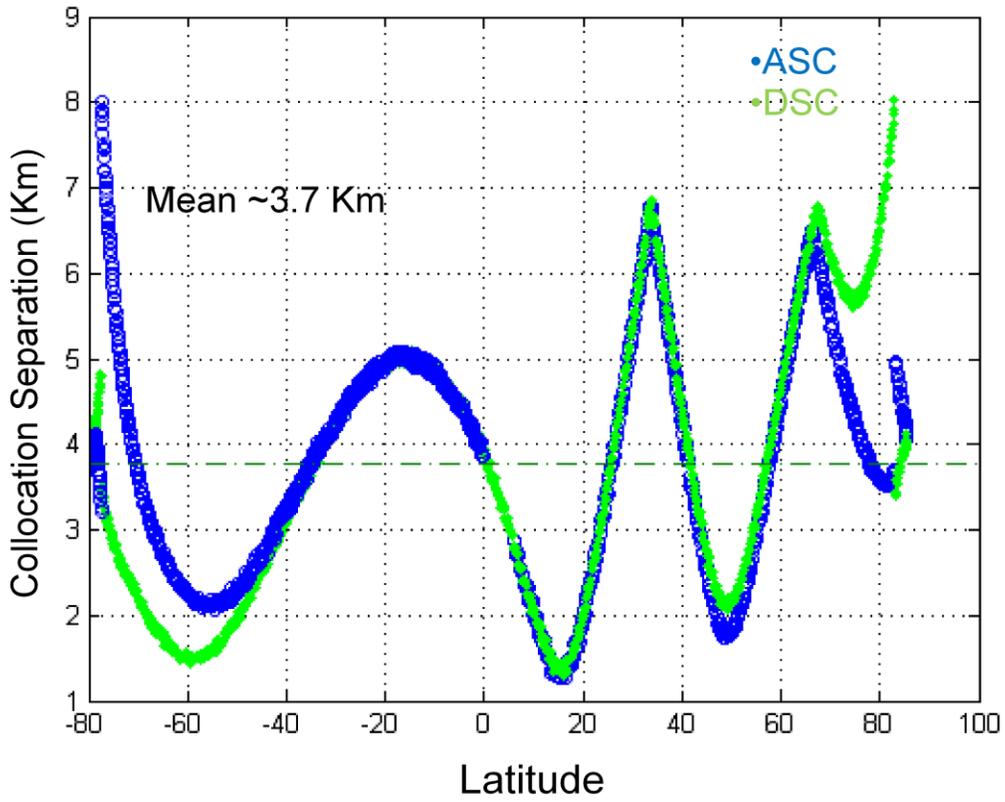


Figure 74: Forward/Aft Beam 3 Collocation Separation.

In a similar manner, Figure 75 shows the collocation separation of the forward/aft beam 4.

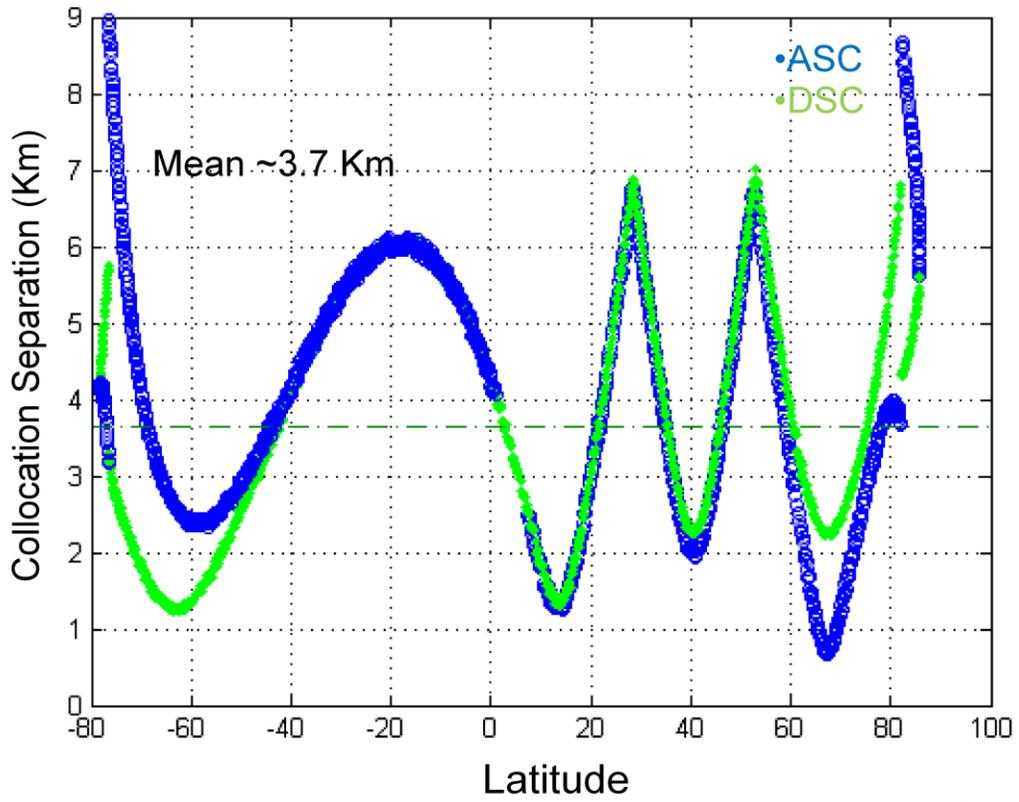


Figure 75: Forward/Aft Beam 4 Collocation Separation.

In a similar manner, Figure 76 shows the collocation separation of the forward/aft beam 5.

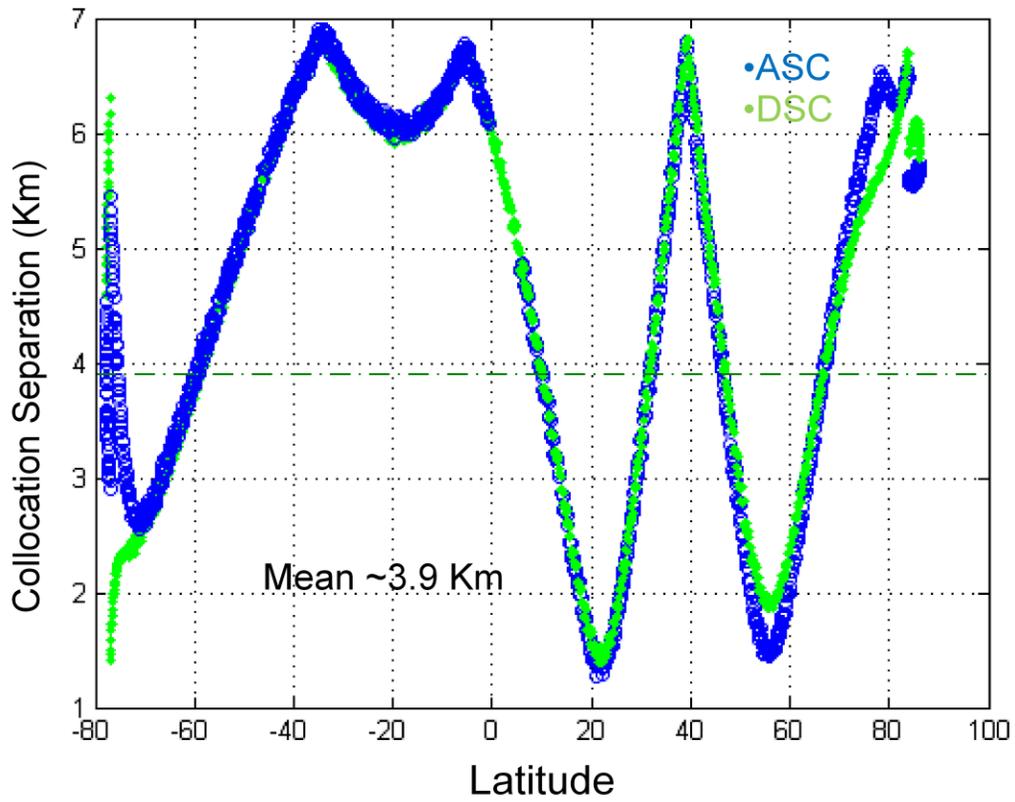


Figure 76: Forward/Aft Beam 5 Collocation Separation.

In a similar manner, Figure 77 shows the collocation separation of the forward/aft beam 6.

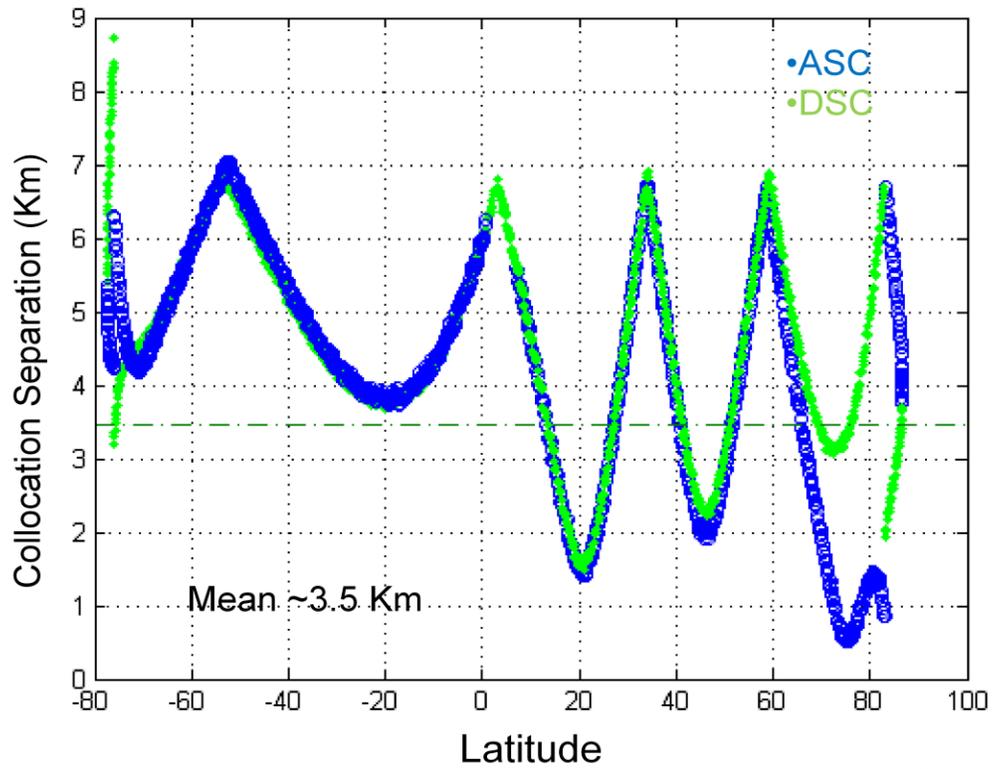


Figure 77: Forward/Aft Beam 6 Collocation Separation.

In a similar manner, Figure 78 shows the collocation separation of the forward/aft beam 7.

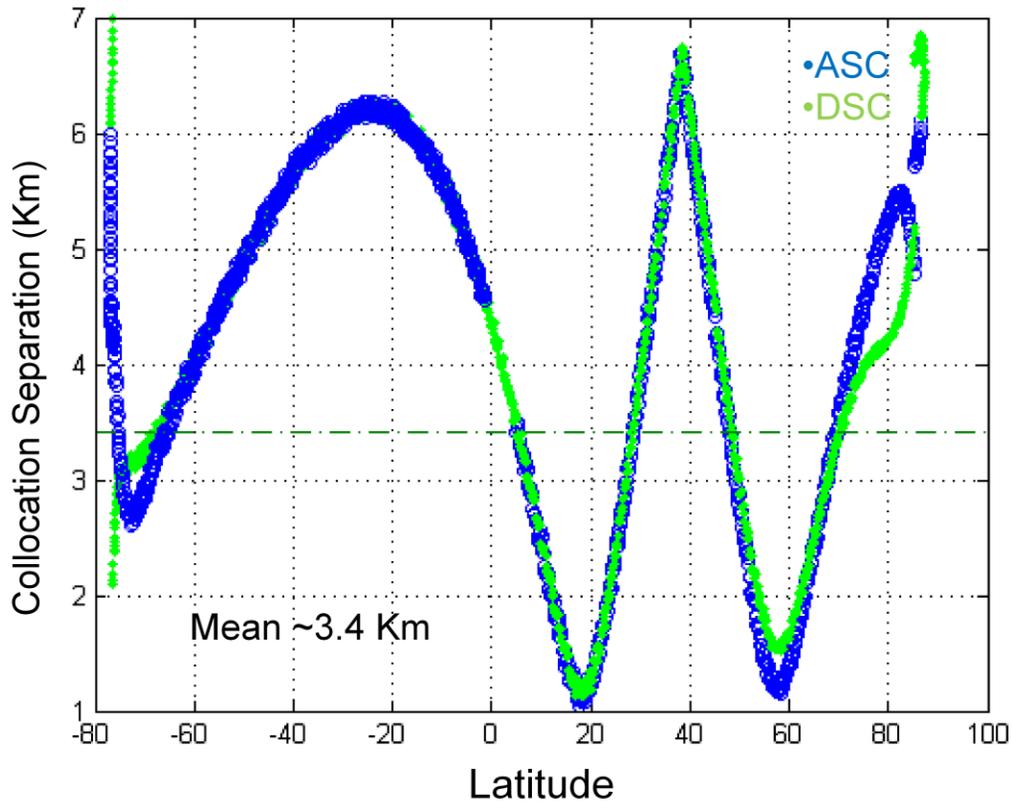


Figure 78: Forward/Aft Beam 7 Collocation Separation.

In a similar manner, Figure 79 shows the collocation separation of the forward/aft beam 8. The mean collocation separation for each of the eight forward/aft beams pairs is approximately 4 Km.

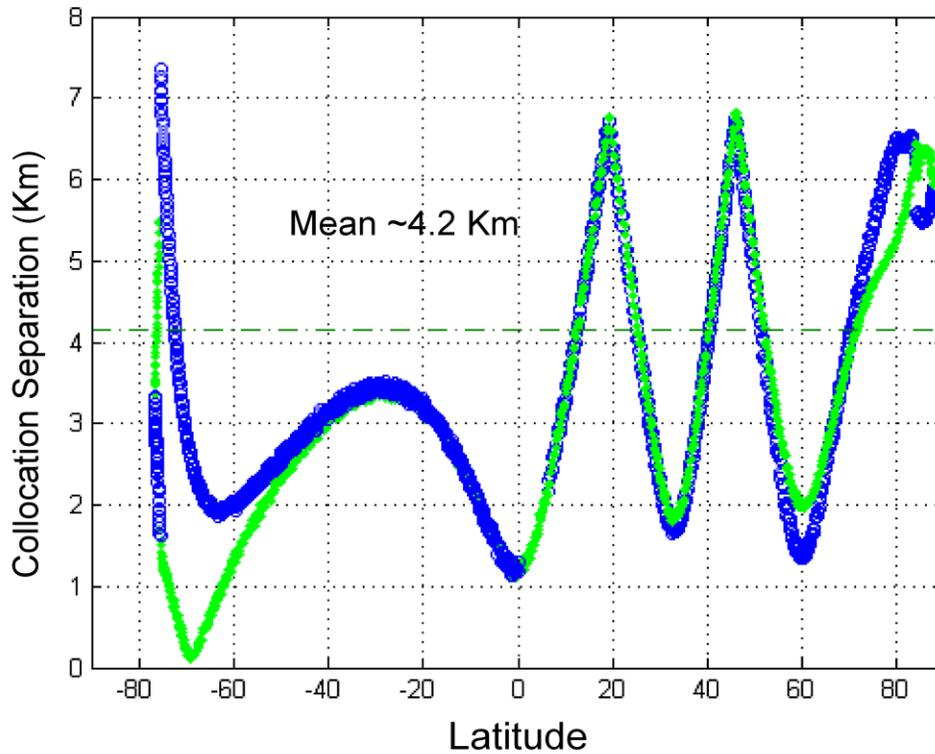


Figure 79: Forward/Aft Beam 8 Collocation Separation.

Figure 80 shows a snapshot of the forward (green) and aft (red) beams and Figure 81 shows the progression of the aft beams as they collocate with three of the forward beams. Notice that beam 1, 2, and 7 collocate in the snapshot as indicated. Figure 82 shows all the forward beams of the 437th scan collocating with aft beams. The scan number of the aft beams which collocate with the forward beams, as the satellite propagates in time, is also indicated in Figure 82.

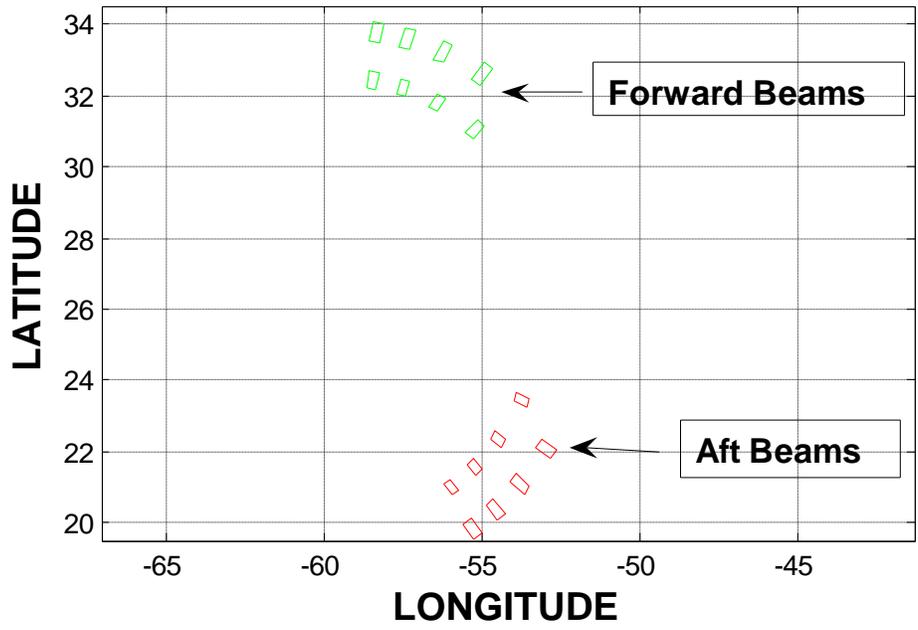


Figure 80: Forward Aft Beams.

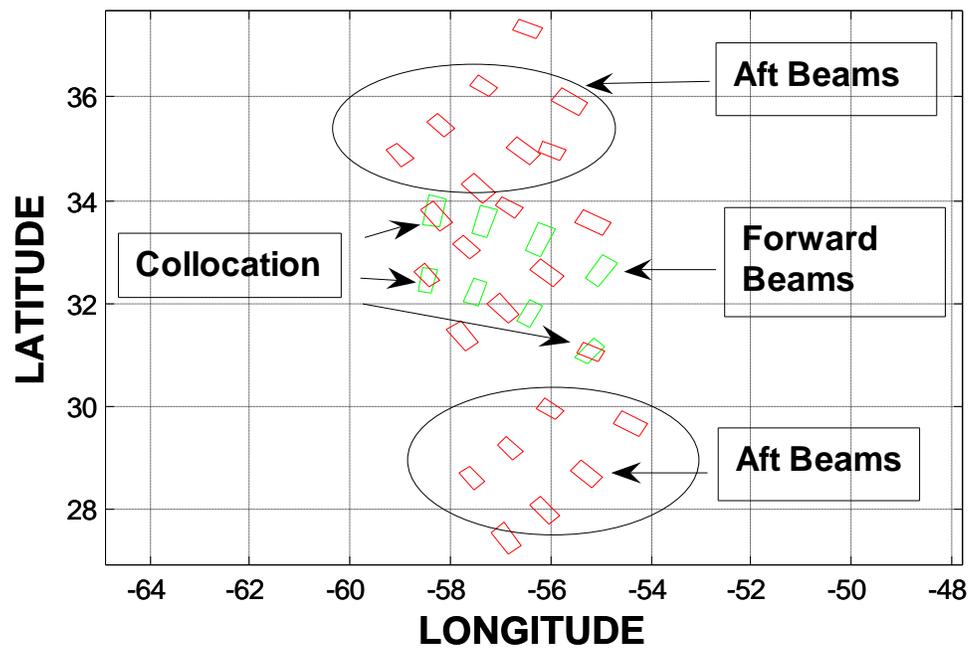


Figure 81: A Snapshot of 3 Forward Beams Collocating.

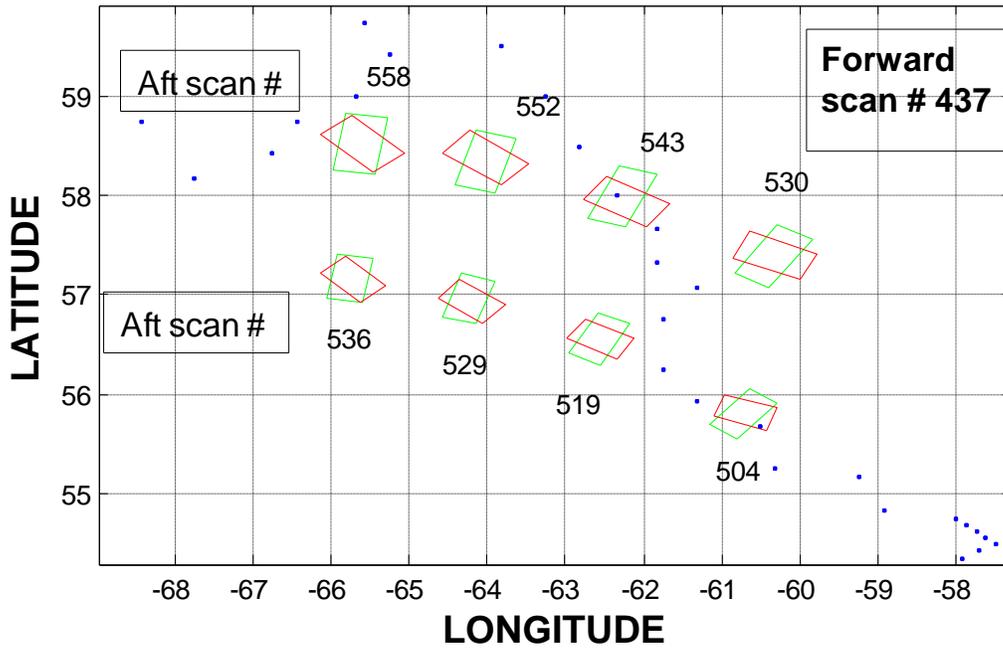


Figure 82: All Forward Beams Collocating with Different Aft Scans.

CHAPTER 5. CONCLUSIONS & FUTURE WORK

The 3-channel MWR T_b dataset was simulated using WindSat's IDR as an input source. One of the key advantages of using WindSat was its similarities in satellite ground-track, radiometric frequencies and swath overlap with MWR. Further, using WindSat's high resolution IDR data product resulted in an accurate averaging of IDR's T_b s into MWR pixels. However, due to the difference in incidence angles between WindSat and MWR, T_b normalizations were required by using a radiative transfer model to transform WindSat T_b to equivalent MWR incidence angles.

WindSat geometry imposed considerable azimuthal constraints on the simulation. Firstly, due to the difference in altitude between the two sensors, equivalent WindSat azimuth angles were calculated. However, the limitations of the angular measurements of WindSat swath, collocated between its K and Ka band, resulted in a mirrored left-sided MWR swath instead of the actual right-sided swath. Further, the WindSat EDR azimuthal limitations even reduced this left-sided mirrored swath by $\sim 6^\circ$, corresponding to two MWR beams. Since the actual MWR swath could not be simulated using WindSat IDR/EDR data, it was decided to discard this approach and to simulate the maximum possible number of MWR pixels for geophysical algorithm development. This resulted in nineteen equivalent MWR beams simulated in the azimuthal limitations of WindSat IDR/EDR.

The nineteen MWR beams were simulated with proper IFOV size, orientation, and relative beam to beam spatial sampling. Exact elliptical MWR IFOV dimensions were calculated for

the two incidence angles, but rectangular IFOVs were used instead of ellipses, since the accuracy of the MWR T_b spatial averaging did not decrease significantly, however, the complexity of implementation was considerably reduced. The nineteen MWR beams were simulated using the exact cross-track spacing as the actual MWR spacing of 46.4 Km. IDR beam fill fractions (along scan and along-track) were computed for the rectangular MWR IFOVs at 52° and 58° incidence angles. The IDR pixels, filling an MWR IFOV, were filtered around the center azimuth angle of the MWR beam so that the resultant MWR beam has an exact azimuthal orientation of its IFOV. Further, the MWR beams were sampled at the actual MWR sampling time of 1.92 sec corresponding to the MWR inter-scan distance of 13.1 Km. Finally, the T_b of an MWR beam was calculated as the average T_b of the IDR pixels filling the MWR beam IFOV (along-scan & along-track).

The results show that the simulated MWR beams have an accurate beam filling of IDR pixels (along-scan & along-track) at both incidence angles. Also, the orientation of rectangular MWR beams by azimuth angle is preserved because the IDR pixels, falling into an MWR beam (along-scan), are centered symmetrically around the beam's azimuth angle. Further, the 23.8 V and the 36.5 (V & H) channels of MWR are simulated as a forward swath highly collocated between the two frequency bands. It was also observed that the along-track sampling of the simulated MWR beams accurately resulted in the actual MWR inter-scan distance of 13.1 Km, which produced a periodic skipping of IDR scans after every ~ 25 MWR scans. Finally, in addition to the spatial/temporal accuracy, the MWR T_b simulation shows radiometric precision. Earth gridded T_b s (averaged into 0.5° lat/ln g boxes) of corresponding orbits of simulated MWR beams and IDR pixels, without applying incidence

angle corrections, show high similarity between the two T_b images. The mean difference between the two T_b images is ~ 0 Kelvin with a standard deviation of only ~ 2 Kelvin, which illustrates high radiometric accuracy of the simulation.

The yaw steering of Aquarius/SAC-D, provided by CONAE, was verified using STK and Matlab softwares. The forward and aft (yawed) beam boresight points were computed in Matlab using the sub-satellite points and satellite's velocity azimuth of Aquarius/SAC-D, generated in STK at the MWR's actual sequential beam sampling time of 0.24 sec. The collocation separation distance (Km), for the eight forward and aft beams pairs, was computed for one full orbit. The results show a mean collocation separation distance of ~ 4 Km for each forward/aft beam pairs, which is in accordance with the MWR beam collocation requirements.

To facilitate the post-launch radiometric calibration of MWR, an orbital analysis and a preliminary study on the swath collocation between the two sensors was also performed using STK. It was observed that the ground-tracks of the two satellite phased in and out periodically due to the difference in the altitudes and hence the velocities of the satellites. The similarities of their orbits result in many radiometric collocations between the two sensors, which increase and decrease with the phasing-in and phasing-out of the satellite ground-tracks, respectively. It was observed that after ~ 5 months in orbit, there will be ~ 1 million ocean collocations between the two sensors, which can be a useful asset for the post-launch radiometric calibration between the two satellites.

The MWR T_b simulation has been validated to meet the required radiometric and temporal/spatial accuracy. CONAE's yaw steering technique to collocate the forward & aft MWR beams has also been verified with a mean collocation separation distance of ~ 4 Km. These positive results will play a significant role in the development of pre-launch geophysical parameter retrieval algorithms. Also, the numerous ocean collocations (~ 1 Million in a 5 month period) between the two sensors can be very useful for the post-launch radiometric calibration of MWR.

In the future, a four month simulated T_b dataset for the 3-channel MWR will be delivered to CONAE for the pre-launch geophysical retrieval algorithms development with CFRSL's collaboration. The MWR retrievals will finally be validated through near simultaneous, collocated comparisons with WindSat's Environmental Data Records (EDRs).

APPENDIX A CONICAL SCANNING GEOMETRY

This appendix presents a brief description of conical scanning sensor geometry. Figure 83 shows the 3-D and polar view of a conical scanning sensor as it flies over a spherical Earth. The 3-D view shows the direction of flight of the sensor while its bore-sight (orange) conically scans along the curvature of the Earth. The IFOVs of a portion of the conical scan are also shown. The polar view shows the direction of sensor's flight (purple arrow) and the orientation of the IFOVs as it conically scans between $\pm \psi$. The projection of the conical scan in the across-track is shown as a black line orthogonal to the flight direction and passing through the Nadir.

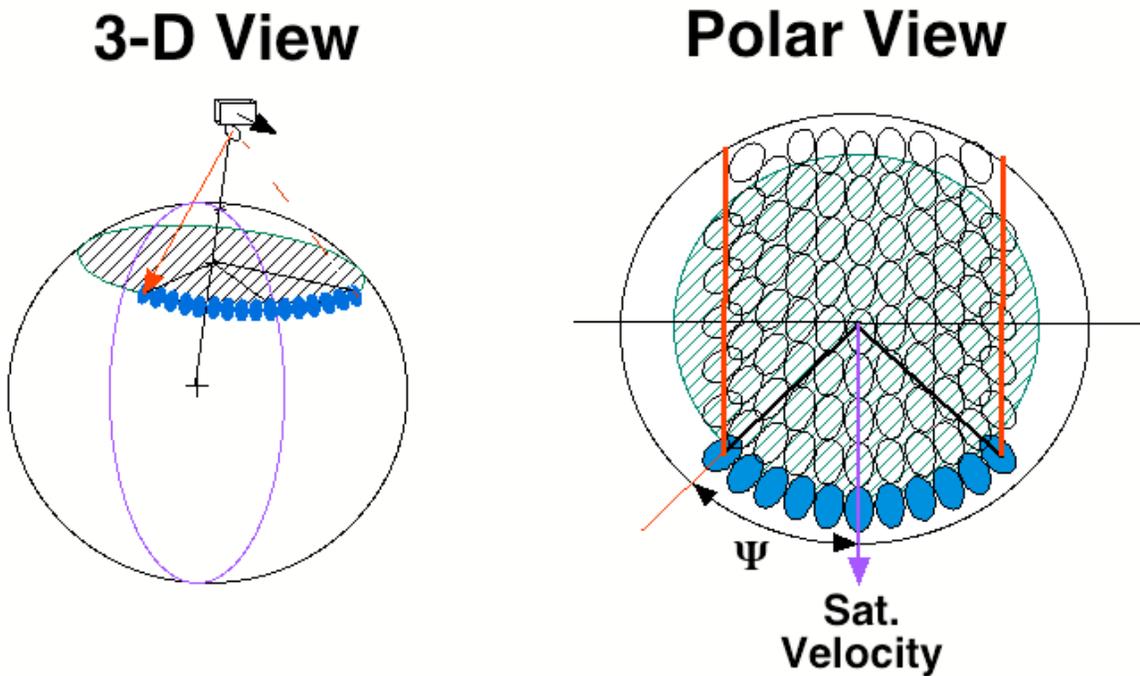


Figure 83: Conical Scanning Sensor Geometry (3-D and Polar View) [17].

Figure 84 shows the geometry for a conical scanning sensor on a satellite travelling with a velocity of $V_{S/C}$, earth incidence angle θ_i , cone angle ϕ , azimuth angle ψ , earth central angle γ , cross-track central angle γ_{ct} , height h , radius of earth ρ , slant range R & swath width SW .

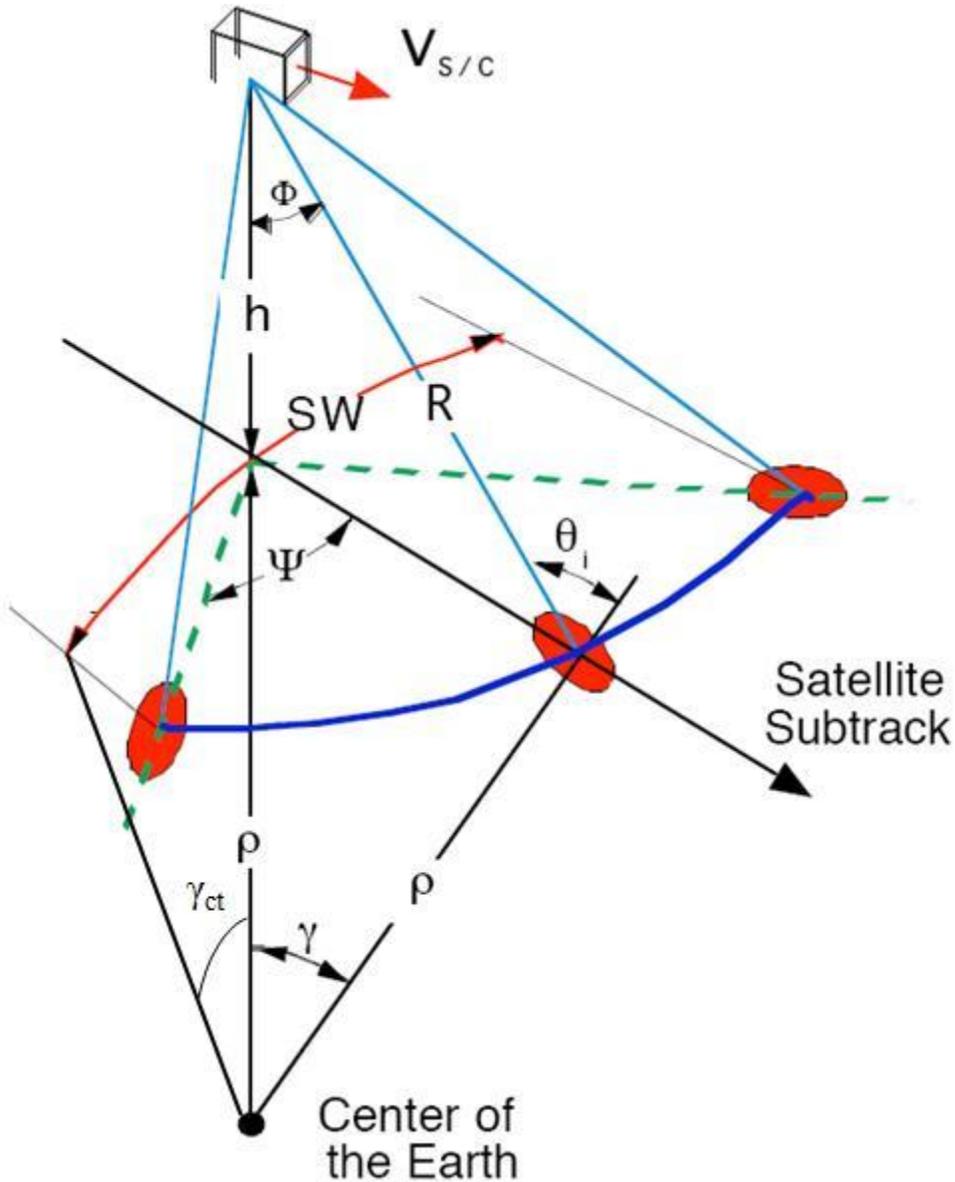


Figure 84: Conical Scanning Sensor Geometry [17].

As the satellite flies, the sensor conically scans between $\pm \psi$, along the blue arc. The orientation of the elliptical IFOVs changes along the scan. The projection of the conical scan in the across-track, (a red arc) along the curvature of the earth, perpendicular to the satellite sub-track and passing through the Nadir is the swath width. The angle that the center of the earth makes in the incidence plane between the beam bore-sight and the Nadir is the earth central angle, γ , while the angle that it makes between the edge of the swath and the Nadir is cross-track central angle, γ_{ct} .

The relationship between azimuth angle ψ , earth central angle γ & cross-track central angle γ_{ct} is given by the following equation:

$$\sin(\gamma_{ct}) = \sin(\gamma) \times \sin(\psi) \quad (\text{A-1})$$

The earth central angle, γ , can be calculated in the incident plane by the following equation:

$$\gamma = \phi - \theta_i \quad (\text{A-2})$$

The cone angle, ϕ , can be calculated by the following equation:

$$\phi = \sin^{-1} \left[\frac{\rho}{\rho + h} \sin \theta_i \right] \quad (\text{A-3})$$

Figure 85 shows the satellite sub-track plane view of conical scanning sensor geometry. The swath width, defined by the red arc along the curvature of the earth, is dictated by the cross-track central angle, γ_{ct} . The swath width can be calculated by the following equation:

$$SwathWidth = 2\rho\gamma_{ct} \quad (A-4)$$

where ρ is the radius of the earth and γ_{ct} is the cross-track central angle in radians.

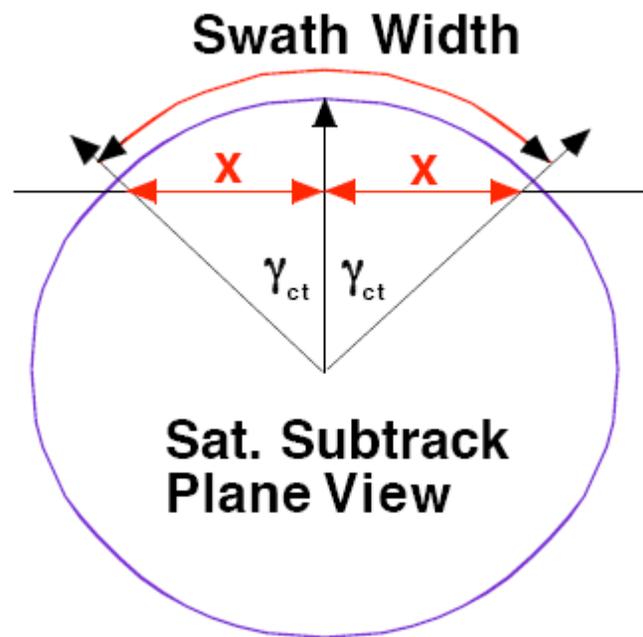


Figure 85: Conical Scanning Sensor Geometry (Satellite Sub-Track Plane View) [17].

APPENDIX B RADIATIVE TRANSFER THEORY

The ocean brightness temperature, viewed by a satellite-borne microwave radiometer, can be calculated using the electromagnetic theory of radiative transfer [18]. According to this theory, the ocean surface and the slightly absorbing atmosphere contribute to the microwave emissions. Since these emissions are incoherent noise, the various components can be summed together as scalars at the microwave radiometer antenna aperture to calculate the apparent brightness temperature, T_{ap} . The various sources of microwave emissions are:

1. The ocean T_b , which propagates upwards to the antenna through a partially absorbing atmosphere.
2. The sum of the down-welling T_b and the background cosmic radiations of the atmosphere, which propagate down to the ocean surface and is then reflected and transmitted upwards to the antenna.
3. Direct upwelling emission of the atmosphere to the antenna.

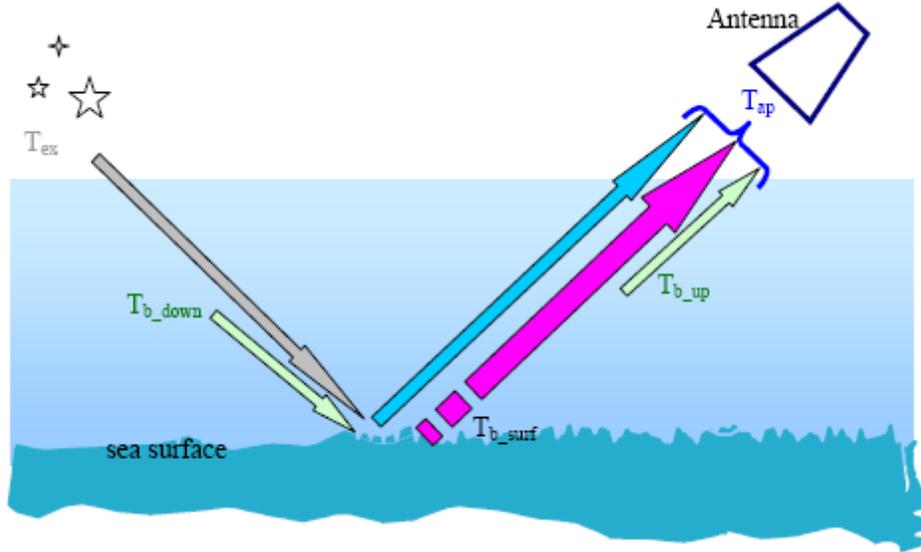


Figure 86: Radiative Transfer Theory Concept.

Figure 86 shows the major microwave emissions that contribute to T_{ap} . The sky brightness temperature, T_{sky} is defined as a sum of atmosphere down-welling and attenuated external (space) brightness temperature.

$$T_{sky} = \tau T_{ex} + T_{b_down} \quad (B.1)$$

where τ is the atmospheric power transmissivity. The ocean surface reflects the sky brightness with some loss.

$$T_{refl} = (1 - \epsilon) T_{sky} \quad (B.2)$$

where, ϵ is the ocean surface emissivity and $(1 - \epsilon)$ is Fresnel power reflectivity. The ocean brightness temperature is

$$T_{b_surf} = \epsilon * SST \quad (B.3)$$

where, SST is the sea surface physical temperature in Kelvin. At the radiometer antenna, the apparent brightness temperature is the incoherent summation

$$T_{ap} = T_{b_up} + \tau(T_{b_surf} + T_{refl}) \quad (B.4)$$

The emissivity (ϵ) varies with the complex dielectric constant, the surface roughness, the presence of sea foam and the radiometer to sea surface viewing geometry (incidence angle and polarization). Using the Debye equation [19], the dielectric properties of sea water are determined as a function of the salinity, the physical temperature and the microwave frequency (wavelength).

From the radiative transfer equations, the $[1 - \epsilon(\theta, \lambda, \rho)]$ term represents the ocean Fresnel specular power reflection coefficient, $\Gamma(\theta, \lambda, \rho)$. From EM field theory [19], these coefficients (power) for horizontal and vertical EM wave polarizations and an arbitrary incidence angle, θ are given as

$$\Gamma(\theta, \lambda, h) = \left| \frac{\cos\theta - \sqrt{e_w - \sin^2\theta}}{\cos\theta + \sqrt{e_w - \sin^2\theta}} \right|^2, \text{ ratio} \quad (\text{B.5})$$

$$\Gamma(\theta, \lambda, v) = \left| \frac{e_w \cos\theta - \sqrt{e_w - \sin^2\theta}}{e_w \cos\theta + \sqrt{e_w - \sin^2\theta}} \right|^2, \text{ ratio} \quad (\text{B.6})$$

where, e_w , is the complex dielectric constant of sea water, h symbolizes a horizontally polarized signal and v , a vertically polarized signal.

For realistic wind-roughened ocean surfaces, the microwave emissions depend on both the smooth surface Fresnel power reflection coefficients and the degree of surface roughness. Thus, the specular emissivity is modified by an additive emissivity term to account for roughness effects of ocean waves, which includes the effects of foam produced by the breaking ocean waves.

In the microwave frequency range; there are two resonant absorption lines, 22.2 GHz and 183 GHz, for the water vapor molecule [20]. The amount of absorption is dependent on the water vapor density and the air temperature and pressure of the atmosphere along the observation path. The second major contributor to atmospheric emission is oxygen. Electromagnetic absorption by oxygen molecules occurs at 60 GHz and 118.75 GHz. Like water vapor, the strength of the absorption depends on the number of oxygen molecules, altitude, pressure, and air temperature. As altitude increases, the number of oxygen molecules decreases, but the decrease in pressure reduces the absorption bandwidth causing an increase in emission.

APPENDIX C MATLAB SCRIPT

```

%The starting point of the MWR simulation

function Generate_MWR_Data(path)

%Get the IDR files

files = ls ([path, '\idr_r*']);

%Get the month of IDR files

files_indices = str2num(files(:,20:21));

%Filter files by month

files_Jan = files(find(files_indices(:,2) == 2), :);

files_indices = str2num(files_Jan(:,12:13));

%Separate the month files into 23.8 & 36.5 GHz

files23 = files_Jan(find(files_indices(:,2) == 23), :);

files37 = files_Jan(find(files_indices(:,2) == 37), :);

[m23, n23] = size(files23);

[m37, n37] = size(files37);

%Generate an MWR file for each IDR file

for file = 1 : 1: m37

mwr_37_data_preparation_miss_scans( files37(file, :), files37( file + 1, :));

mwr_23_data_preparation_miss_scans( files23(file, :), files23( file + 1, :));

end

```

```

function [mwr_pixels_37] = mwr_37_data_preparation_miss_scans( file1, file2)

tic

%Load IDR file 1 and 2

idr37_1 = load(['x:\IDR_SDR_DATA\',file1]);
idr37_2 = load(['x:\IDR_SDR_DATA\',file2]);

%Remove overlapping scans from current file i.e. file 1 after comparing
% with file 2

[tf, loc] = ismember( idr37_2.idr37( :, 1), idr37_1.idr37( :, 1));

clear('idr37_2');

clear('tf');

matched_ids = find(loc(:,1) ~= 0);
matched_ids = loc(matched_ids,1);

clear('loc');

idr37_1.idr37(matched_ids, :) = [];

clear('matched_ids');

%Remove duplicates within file1 and then sort

[b,m,n] = unique(idr37_1.idr37(:,1), 'rows');

clear('b', 'n');

idr37 = idr37_1.idr37(m, :);

clear('idr37_1', 'm');

%Convert longitude from +-180 to 0-360

negative_long = find(idr37(:, 7) < 0);

idr37(negative_long, 7) = idr37(negative_long, 7) + 360;

```

```

%Convert CAA from radians to degrees

idr37(:, 10) = idr37(:, 10) * (180/pi);

%Remove irregular scans so that the length remains divisible by 769 (forward scan length of 37
Ghz IDR)

for scan_num = min(idr37(:, 15)):1:max(idr37(:, 15))

row_ids = find(idr37(:, 15) == scan_num);

if length(row_ids) ~= 769

idr37(row_ids, :) = [];

end

end

%Normalize the scan numbers

min_scan_number = min(idr37(:, 15));

idr37(:, 15) = idr37(:, 15) - min_scan_number + 1;

%Remove duplicate rows

[unique_scans,m,n] = unique(idr37(:,15), 'rows');

clear('m', 'n');

%Scan centers for IDR 37 GHz

idr_scan_center37 = [unique_scans ((unique_scans - 1) * 12.6) + 10];

%Compute MWR scan starts

mwr_scan_start = [0:13.1025:idr_scan_center37(length(idr_scan_center37), 2)]';

idr_start_scan = [];

%Eliminate skipped IDR scans

for mwr_scan_index = 1:length(mwr_scan_start)

```

```

%find start idr scan

ids = find(idr_scan_center37(:,2) > mwr_scan_start(mwr_scan_index, 1));

idr_start_scan = [idr_start_scan;idr_scan_center37( ids(1,1), 1)];

end

[idr_start_scan,m,n] = unique(idr_start_scan(:,1), 'rows');

clear('m', 'n');

%Add the scan angle (Azimuth) as the last column to the IDR data

idr37 = [idr37 ((idr37(:, 16)*-0.0036657) + 3.61236329)*180/pi];

%Filter out invalid ids for 58 deg beams of 37 GHz frequency

valid_ids = find( (idr37(:,16) >= 803 & idr37(:,16) <= 810) | (idr37(:,16) >= 836 & idr37(:,16)
<= 843) | (idr37(:,16) >= 867 & idr37(:,16) <= 874) | (idr37(:,16) >= 900 & idr37(:,16) <= 907)
| (idr37(:,16) >= 927 & idr37(:,16) <= 934) | (idr37(:,16) >= 955 & idr37(:,16) <= 962) |
(idr37(:,16) >= 1112 & idr37(:,16) <= 1119) | (idr37(:,16) >= 1079 & idr37(:,16) <= 1086) |
(idr37(:,16) >= 1051 & idr37(:,16) <= 1058) | (idr37(:,16) >= 1022 & idr37(:,16) <= 1029));

idr37_58 = idr37(valid_ids,:);

%Filter out invalid ids for 52 deg beams of 37 GHz frequency

valid_ids = find( (idr37(:,16) >= 819 & idr37(:,16) <= 825) | (idr37(:,16) >= 852 & idr37(:,16)
<= 858) | (idr37(:,16) >= 883 & idr37(:,16) <= 889) | (idr37(:,16) >= 913 & idr37(:,16) <= 919)
| (idr37(:,16) >= 941 & idr37(:,16) <= 947) | (idr37(:,16) >= 1098 & idr37(:,16) <= 1104) |
(idr37(:,16) >= 1066 & idr37(:,16) <= 1072) | (idr37(:,16) >= 1037 & idr37(:,16) <= 1043) |
(idr37(:,16) >= 1008 & idr37(:,16) <= 1014));

idr37_52 = idr37(valid_ids,:);

```

```

clear('valid_ids', 'idr37');

clear('idr_scans','idr_scan_center37', 'mwr_scan_start', 'mwr_scan_index');

mwr_scan_width58 = 80;

mwr_pixel_width58 = 8;

mwr_scan_width52 = 63;

mwr_pixel_width52 = 7;

mwr_pixel_indices = [];

mwr_pixels_37 = [];

%Average MWR pixel fields along-scan only at 58 degs

%as/dsc flag

temp = dec2bin(idr37_58(:,14),32);

ascdsc_flag58 = bin2dec(temp(:,23));

ascdscstuff =

mean(reshape(ascdsc_flag58(:,1),mwr_pixel_width58,numel(ascdsc_flag58(:,1))/mwr_pixel_wid
th58)',2);

ascdsc = reshape(ascdscstuff,10,numel(ascdscstuff)/10);

%TbV

```

```
tbvstuff =  
mean(reshape(idr37_58(:,2),mwr_pixel_width58,numel(idr37_58(:,2))/mwr_pixel_width58)',2);  
tbv = reshape(tbvstuff,10,numel(tbvstuff)/10)';
```

```
tbhstuff =  
mean(reshape(idr37_58(:,3),mwr_pixel_width58,numel(idr37_58(:,3))/mwr_pixel_width58)',2);  
tbh = reshape(tbhstuff,10,numel(tbhstuff)/10)';
```

```
stoke1stuff =  
mean(reshape(idr37_58(:,4),mwr_pixel_width58,numel(idr37_58(:,4))/mwr_pixel_width58)',2);  
stoke1 = reshape(stoke1stuff,10,numel(stoke1stuff)/10)';
```

```
stoke2stuff =  
mean(reshape(idr37_58(:,5),mwr_pixel_width58,numel(idr37_58(:,5))/mwr_pixel_width58)',2);  
stoke2 = reshape(stoke2stuff,10,numel(stoke2stuff)/10)';
```

```
timestuff =  
mean(reshape(idr37_58(:,1),mwr_pixel_width58,numel(idr37_58(:,1))/mwr_pixel_width58)',2);  
time = reshape(timestuff,10,numel(timestuff)/10)';
```

```
latstuff =  
median(reshape(idr37_58(:,6),mwr_pixel_width58,numel(idr37_58(:,6))/mwr_pixel_width58)',2  
);
```

```

lat = reshape(latstuff,10,numel(latstuff)/10)';

lonstuff =
handle_transition_in_degrees(reshape(idr37_58(:,7),mwr_pixel_width58,numel(idr37_58(:,7))/m
wr_pixel_width58)', 0);

lon = reshape(lonstuff,10,numel(lonstuff)/10)';

azimuthstuff =
mean(reshape(idr37_58(:,19),mwr_pixel_width58,numel(idr37_58(:,19))/mwr_pixel_width58)',2
);

azimuth = reshape(azimuthstuff,10,numel(azimuthstuff)/10)';

caastuff =
handle_transition_in_degrees(reshape(idr37_58(:,10),mwr_pixel_width58,numel(idr37_58(:,10))
/mwr_pixel_width58)', 0);

caa = reshape(caastuff,10,numel(caastuff)/10)';

%If all the IDR pixels along scan, correspond to a common surface type
%the surface type of the result would be a specific type otherwise it would
%be -1

surfacestuff =
reshape(idr37_58(:,17),mwr_pixel_width58,numel(idr37_58(:,17))/mwr_pixel_width58)';

mwr_surface_type = [];

```

```

for i = 1:1:length(surfacestuff)

    %Land = 0

    %Not used = 1

    %Near coast = 2

    %Ice = 3

    %Possible ice = 4

    %Ocean = 5

    %coast = 6

    %spare = 7

    surface_type = -1;

ids = find(surfacestuff(i,:) == 0);

if length(ids) == mwr_pixel_width58

    surface_type = 0;

end

ids = find(surfacestuff(i,:) == 2);

if length(ids) == mwr_pixel_width58

    surface_type = 2;

end

ids = find(surfacestuff(i,:) == 3);

if length(ids) == mwr_pixel_width58

    surface_type = 3;

end

ids = find(surfacestuff(i,:) == 4);

```

```

if length(ids) == mwr_pixel_width58
    surface_type = 4;
end
ids = find(surfacestuff(i,:) == 5);
if length(ids) == mwr_pixel_width58
    surface_type = 5;
end
ids = find(surfacestuff(i,:) == 6);
if length(ids) == mwr_pixel_width58
    surface_type = 6;
end
ids = find(surfacestuff(i,:) == 7);
if length(ids) == mwr_pixel_width58
    surface_type = 7;
end
mwr_surface_type = [mwr_surface_type; surface_type];
end
surface = reshape(mwr_surface_type,10,numel(mwr_surface_type)/10)';

%Average MWR pixel fields along-scan only at 52 degs
%asc/dsc flag
temp = dec2bin(idr37_52(:,14),32);
ascdsc_flag52 = bin2dec(temp(:,23));

```

```

ascdscstuff =
mean(reshape(ascdsc_flag52(:,1),mwr_pixel_width52,numel(ascdsc_flag52(:,1))/mwr_pixel_wid
th52)',2);
ascdsc52 = reshape(ascdscstuff,9,numel(ascdscstuff)/9)';
%TbV
tbvstuff =
mean(reshape(idr37_52(:,2),mwr_pixel_width52,numel(idr37_52(:,2))/mwr_pixel_width52)',2);
tbv52 = reshape(tbvstuff,9,numel(tbvstuff)/9)';

tbhstuff =
mean(reshape(idr37_52(:,3),mwr_pixel_width52,numel(idr37_52(:,3))/mwr_pixel_width52)',2);
tbh52 = reshape(tbhstuff,9,numel(tbhstuff)/9)';

stoke1stuff =
mean(reshape(idr37_52(:,4),mwr_pixel_width52,numel(idr37_52(:,4))/mwr_pixel_width52)',2);
stoke1_52 = reshape(stoke1stuff,9,numel(stoke1stuff)/9)';

stoke2stuff =
mean(reshape(idr37_52(:,5),mwr_pixel_width52,numel(idr37_52(:,5))/mwr_pixel_width52)',2);
stoke2_52 = reshape(stoke2stuff,9,numel(stoke2stuff)/9)';

timestuff =
mean(reshape(idr37_52(:,1),mwr_pixel_width52,numel(idr37_52(:,1))/mwr_pixel_width52)',2);

```

```

time52 = reshape(timestuff,9,numel(timestuff)/9)';

latstuff =
median(reshape(idr37_52(:,6),mwr_pixel_width52,numel(idr37_52(:,6))/mwr_pixel_width52)',2
);
lat52 = reshape(latstuff,9,numel(latstuff)/9)';

lonstuff =
handle_transition_in_degrees(reshape(idr37_52(:,7),mwr_pixel_width52,numel(idr37_52(:,7))/m
wr_pixel_width52)', 0);
lon52 = reshape(lonstuff,9,numel(lonstuff)/9)';

azimuthstuff =
mean(reshape(idr37_52(:,19),mwr_pixel_width52,numel(idr37_52(:,19))/mwr_pixel_width52)',2
);
azimuth52 = reshape(azimuthstuff,9,numel(azimuthstuff)/9)';

caastuff =
handle_transition_in_degrees(reshape(idr37_52(:,10),mwr_pixel_width52,numel(idr37_52(:,10))
/mwr_pixel_width52)', 0);
caa52 = reshape(caastuff,9,numel(caastuff)/9)';

%If all the IDR pixels along scan, correspond to a common surface type

```

```

%the surface type of the result would be a specific type otherwise it would
%be -1

surfacestuff =
reshape(idr37_52(:,17),mwr_pixel_width52,numel(idr37_52(:,17))/mwr_pixel_width52);

mwr_surface_type = [];

for i = 1:1:length(surfacestuff)

    %Land = 0

    %Not used = 1

    %Near coast = 2

    %Ice = 3

    %Possible ice = 4

    %Ocean = 5

    %coast = 6

    %spare = 7

    surface_type = -1;

ids = find(surfacestuff(i,:) == 0);

if length(ids) == mwr_pixel_width52

    surface_type = 0;

end

ids = find(surfacestuff(i,:) == 2);

if length(ids) == mwr_pixel_width52

    surface_type = 2;

end

```

```

ids = find(surfacestuff(i,:) == 3);
if length(ids) == mwr_pixel_width52
    surface_type = 3;
end
ids = find(surfacestuff(i,:) == 4);
if length(ids) == mwr_pixel_width52
    surface_type = 4;
end
ids = find(surfacestuff(i,:) == 5);
if length(ids) == mwr_pixel_width52
    surface_type = 5;
end
ids = find(surfacestuff(i,:) == 6);
if length(ids) == mwr_pixel_width52
    surface_type = 6;
end
ids = find(surfacestuff(i,:) == 7);
if length(ids) == mwr_pixel_width52
    surface_type = 7;
end
mwr_surface_type = [mwr_surface_type; surface_type];
end
surface_52 = reshape(mwr_surface_type,9,numel(mwr_surface_type)/9)';

```

%58

mwrpixonlat= [];

mwrpixonlon= [];

mwrpixonbv= [];

mwrpixonbh= [];

mwrpixon3rdstoke= [];

mwrpixon4thstoke= [];

mwrpixontime= [];

mwrpixonsurface = [];

mwrpixonazimuth = [];

mwrpixoncaa = [];

mwrpixonscan = [];

mwrscdsc = [];

%52

mwrpixonlat52= [];

mwrpixonlon52= [];

mwrpixonbv52= [];

mwrpixonbh52= [];

mwrpixon3rdstoke52= [];

mwrpixon4thstoke52= [];

mwrpixontime52= [];

mwrpixonsurface52 = [];

```

mwrpiazimuth52 = [];

mwrpicaa52 = [];

mwrpiscan52 = [];

mwrascdsc52 = [];

%For each IDR scan, excluding the skipped scans,
%average MWR pixel fields along-track at 58 & 52 degs.
%Since the along-scan of IDR pixels is at least 4, stop at the
%5th last IDR scan
for i = 1:length(idr_start_scan) - 4

    %Get scan sequence for 58 degs incidence angle

    [scanrange58] = get_current_scan_sequence(idr_start_scan(i,1), idr_start_scan, 4, length(lat));

    %Get scan sequence for 52 degs incidence angle

    [scanrange52] = get_current_scan_sequence(idr_start_scan(i,1), idr_start_scan, 3, length(lat));

    %Average IDR fields along-scan at 58 degs

    mwrpilat = [mwrpilat ; median(lat(scanrange58,:), 1)];

    mwrpilong = [mwrpilong ; handle_transition_in_degrees(lon(scanrange58,:), 1)];

    mwrpibtv = [mwrpibtv ; mean(tbv(scanrange58,:), 1)];

    mwrpibtbh = [mwrpibtbh ; mean(tbh(scanrange58,:), 1)];

    mwrpix3rdstoke = [mwrpix3rdstoke ; mean(stoke1(scanrange58,:), 1)];

    mwrpix4thstoke = [mwrpix4thstoke ; mean(stoke2(scanrange58,:), 1)];

    mwrpixtime = [mwrpixtime ; mean(time(scanrange58,:), 1)];

```

```

mwrpixmap = [mwrpixmap ; mean(azimuth(scanrange58,:), 1)];
mwrpixmap = [mwrpixmap ; handle_transition_in_degrees(caa(scanrange58, :), 1)];
mwrascdsc = [mwrascdsc ; mean(ascdsc(scanrange58,:), 1)];
scan = repmat(i, 1, 10);
mwrpixmap = [mwrpixmap ; scan];

%Average IDR fields along-scan at 52 degs
mwrpixmap = [mwrpixmap ; median(lat52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; handle_transition_in_degrees(lon52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(tbv52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(tbh52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(stoke1_52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(stoke2_52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(time52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; mean(azimuth52(scanrange52,:), 1)];
mwrpixmap = [mwrpixmap ; handle_transition_in_degrees(caa52(scanrange52, :), 1)];
mwrascdsc52 = [mwrascdsc52 ; mean(ascdsc52(scanrange52,:), 1)];
scan = repmat(i, 1, 9);
mwrpixmap52 = [mwrpixmap52 ; scan];

%If all the IDR pixels along track, correspond to a common surface type
%the surface type of the result would be a specific type otherwise it would
%be -1

```

```

ids58 = [];

ids52 = [];

surface_type58 = -1;

surface_type52 = -1;

for j = 1:1:10

    %Surface type at 58

temp = find(surface( scanrange58, j) == 0);

if length(temp) == length(scanrange58)

    surface_type58 = 0;

end

temp = find(surface( scanrange58, j) == 2);

if length(temp) == length(scanrange58)

    surface_type58 = 2;

end

temp = find(surface( scanrange58, j) == 3);

if length(temp) == length(scanrange58)

    surface_type58 = 3;

end

temp = find(surface( scanrange58, j) == 4);

if length(temp) == length(scanrange58)

    surface_type58 = 4;

end

temp = find(surface( scanrange58, j) == 5);

```

```

if length(temp) == length(scanrange58)
    surface_type58 = 5;
end
temp = find(surface( scanrange58, j) == 6);
if length(temp) == length(scanrange58)
    surface_type58 = 6;
end
temp = find(surface( scanrange58, j) == 7);
if length(temp) == length(scanrange58)
    surface_type58 = 7;
end
ids58 = [ids58 surface_type58];

```

%Surface type at 52

```

if j < 10
temp = find(surface( scanrange52, j) == 0);
if length(temp) == length(scanrange52)
    surface_type52 = 0;
end
temp = find(surface( scanrange52, j) == 2);
if length(temp) == length(scanrange52)
    surface_type52 = 2;
end

```

```

temp = find(surface( scanrange52, j) == 3);
if length(temp) == length(scanrange52)
    surface_type52 = 3;
end
temp = find(surface( scanrange52, j) == 4);
if length(temp) == length(scanrange52)
    surface_type52 = 4;
end
temp = find(surface( scanrange52, j) == 5);
if length(temp) == length(scanrange52)
    surface_type52 = 5;
end
temp = find(surface( scanrange52, j) == 6);
if length(temp) == length(scanrange52)
    surface_type52 = 6;
end
temp = find(surface( scanrange52, j) == 7);
if length(temp) == length(scanrange52)
    surface_type52 = 7;
end
ids52 = [ids52 surface_type52];
end
end

```

```

mwrpixsurface = [mwrpixsurface;ids58];
mwrpixsurface52 = [mwrpixsurface52;ids52];
end

%Reshape the MWR fields as column vectors
%Reshape into column vectors at 58
[m n] = size(mwrpixlat);
mwrpixlat = reshape(mwrpixlat, m * n , 1 );
[m n] = size(mwrpixlon);
mwrpixlon = reshape(mwrpixlon, m * n , 1 );
negative_long = find (mwrpixlon(:, 1) > 180);
mwrpixlon(negative_long, 1) = mwrpixlon(negative_long, 1) - 360;
[m n] = size(mwrpixtbv);
mwrpixtbv = reshape(mwrpixtbv, m * n , 1 );
[m n] = size(mwrpixtbh);
mwrpixtbh = reshape(mwrpixtbh, m * n , 1 );
[m n] = size(mwrpix3rdstoke);
mwrpix3rdstoke = reshape(mwrpix3rdstoke, m * n , 1 );
[m n] = size(mwrpix4thstoke);
mwrpix4thstoke = reshape(mwrpix4thstoke, m * n , 1 );
[m n] = size(mwrpixtime);
mwrpixtime = reshape(mwrpixtime, m * n , 1 );
[m n] = size(mwrpixsurface);

```

```

mwrpixsurface = reshape(mwrpixsurface, m * n , 1 );
[m n] = size(mwrpixazimuth);
mwrpixazimuth = reshape(mwrpixazimuth, m * n , 1 );
[m n] = size(mwrpixcaa);
mwrpixcaa = reshape( mwrpixcaa, m * n , 1 )*(pi/180);
[m n] = size(mwrpixscan);
mwrpixscan = reshape( mwrpixscan, m * n , 1 );
[m n] = size(mwrascdsc);
mwrascdsc = reshape( mwrascdsc, m * n , 1 );
mwrascdsc(find(mwrascdsc(:, 1) < 0.5), 1) = 0;
mwrascdsc(find(mwrascdsc(:, 1) >= 0.5), 1) = 1;
%Incidence angle adjustment at 37 V 58 & 37 H 58
bias_37_58_V = 0.032*mwrpixtbv + 2.825;
bias_37_58_H = 0.108*mwrpixtbh - 18.25;
incidence_angle = repmat(58, m*n, 1);
%MWR pixels at 58 degs
mwr_pixels_37_58 = [mwrpixtime mwrpixlat mwrpixlon mwrpixtbv mwrpixtbh
mwrpix3rdstoke mwrpix4thstoke mwrpixsurface mwrpixazimuth mwrpixcaa bias_37_58_V
bias_37_58_H incidence_angle mwrpixscan mwrascdsc];

%Reshape the MWR fields as column vectors
%Reshape into column vectors at 52
[m n] = size(mwrpixlat52);

```



```

[m n] = size(mwrascdsc52);
mwrascdsc52 = reshape( mwrascdsc52, m * n , 1 );
mwrascdsc52(find(mwrascdsc52(:, 1) < 0.5), 1) = 0;
mwrascdsc52(find(mwrascdsc52(:, 1) >= 0.5), 1) = 1;
%Incidence angle adjustment at 37 V 58 & 37 H 52
bias_37_52_V = -0.0425*mwrpixtbv52 + 6.212;
bias_37_52_H = -0.0738*mwrpixtbh52 + 9.796;
incidence_angle = repmat(52, m*n, 1);
%MWR pixels at 52 degs
mwr_pixels_37_52 = [mwrpixtime52 mwrpixlat52 mwrpixlon52 mwrpixtbv52 mwrpixtbh52
mwrpix3rdstoke52 mwrpix4thstoke52 mwrpixsurface52 mwrpixazimuth52 mwrpixcaa52
bias_37_52_V bias_37_52_H incidence_angle mwrpixscan52 mwrascdsc52];
%MWR pixels at 37
mwr_pixels_37 = [mwr_pixels_37_58; mwr_pixels_37_52];
mwr_pixels_37 = sortrows(mwr_pixels_37, 1);

%Add beam # to each MWR pixel
mwr_pixels_37 = [mwr_pixels_37 repmat(0, length(mwr_pixels_37), 1)];
%Beam # vector
%Beam1
ids_beam1 = find(mwr_pixels_37(:, 9) >= 37.5 & mwr_pixels_37(:, 9) <= 37.6);
mwr_pixels_37(ids_beam1, 16) = 1;
%Beam2

```

```
ids_beam2 = find(mwr_pixels_37(:, 9) >= 34.3 & mwr_pixels_37(:, 9) <= 34.4);
mwr_pixels_37(ids_beam2, 16) = 2;

%Beam3

ids_beam3 = find(mwr_pixels_37(:, 9) >= 30.6 & mwr_pixels_37(:, 9) <= 30.7);
mwr_pixels_37(ids_beam3, 16) = 3;

%Beam4

ids_beam4 = find(mwr_pixels_37(:, 9) >= 27.3 & mwr_pixels_37(:, 9) <= 27.4);
mwr_pixels_37(ids_beam4, 16) = 4;

%Beam5

ids_beam5 = find(mwr_pixels_37(:, 9) >= 24.1 & mwr_pixels_37(:, 9) <= 24.2);
mwr_pixels_37(ids_beam5, 16) = 5;

%Beam6

ids_beam6 = find(mwr_pixels_37(:, 9) >= 20.8 & mwr_pixels_37(:, 9) <= 20.9);
mwr_pixels_37(ids_beam6, 16) = 6;

%Beam7

ids_beam7 = find(mwr_pixels_37(:, 9) >= 17.2 & mwr_pixels_37(:, 9) <= 17.3);
mwr_pixels_37(ids_beam7, 16) = 7;

%Beam8

ids_beam8 = find(mwr_pixels_37(:, 9) >= 14.5 & mwr_pixels_37(:, 9) <= 14.6);
mwr_pixels_37(ids_beam8, 16) = 8;

%Beam9

ids_beam9 = find(mwr_pixels_37(:, 9) >= 11.5 & mwr_pixels_37(:, 9) <= 11.6);
mwr_pixels_37(ids_beam9, 16) = 9;
```

```

%Beam10

ids_beam10 = find(mwr_pixels_37(:, 9) >= 8.6 & mwr_pixels_37(:, 9) <= 8.8);

mwr_pixels_37(ids_beam10, 16) = 10;

%Beam11

ids_beam11 = find(mwr_pixels_37(:, 9) >= 5.6 & mwr_pixels_37(:, 9) <= 5.7);

mwr_pixels_37(ids_beam11, 16) = 11;

%Beam12

ids_beam12 = find(mwr_pixels_37(:, 9) >= -5.4 & mwr_pixels_37(:, 9) <= -5.3);

mwr_pixels_37(ids_beam12, 16) = 12;

%Beam13

ids_beam13 = find(mwr_pixels_37(:, 9) >= -8.45 & mwr_pixels_37(:, 9) <= -8.35);

mwr_pixels_37(ids_beam13, 16) = 13;

%Beam14

ids_beam14 = find(mwr_pixels_37(:, 9) >= -11.5 & mwr_pixels_37(:, 9) <= -11.4);

mwr_pixels_37(ids_beam14, 16) = 14;

%Beam15

ids_beam15 = find(mwr_pixels_37(:, 9) >= -14.55 & mwr_pixels_37(:, 9) <= -14.45);

mwr_pixels_37(ids_beam15, 16) = 15;

%Beam16

ids_beam16 = find(mwr_pixels_37(:, 9) >= -17.6 & mwr_pixels_37(:, 9) <= -17.5);

mwr_pixels_37(ids_beam16, 16) = 16;

%Beam17

ids_beam17 = find(mwr_pixels_37(:, 9) >= -20.4 & mwr_pixels_37(:, 9) <= -20.3);

```

```
mwr_pixels_37(ids_beam17, 16) = 17;

%Beam18

ids_beam18 = find(mwr_pixels_37(:, 9) >= -24.3 & mwr_pixels_37(:, 9) <= -24.2);
mwr_pixels_37(ids_beam18, 16) = 18;

%Beam19

ids_beam19 = find(mwr_pixels_37(:, 9) >= -27.4 & mwr_pixels_37(:, 9) <= -27.3);
mwr_pixels_37(ids_beam19, 16) = 19;

idr37_52 = [];
idr37_58 = [];

%Save MWR 37 Ghz file

mwr_file_name = ['mwr_fullswath', file1(end-23:end)];

save(['x:\IDR_SDR_DATA\incorrectbias\', mwr_file_name], 'mwr_pixels_37');

toc
```

```

function [mwr_pixels_23] = mwr_23_data_preparation_miss_scans( file1, file2)

tic

%Load IDR file 1 and 2

idr23_1 = load(['x:\IDR_SDR_DATA\',file1]);

idr23_2 = load(['x:\IDR_SDR_DATA\',file2]);

%Remove overlapping scans from current file i.e. file 1 after comparing
% with file 2

[tf, loc] = ismember( idr23_2.idr23( :, 1), idr23_1.idr23( :, 1));

clear('idr23_2');

clear('tf');

matched_ids = find(loc(:,1) ~= 0);

matched_ids = loc(matched_ids,1);

clear('loc');

idr23_1.idr23(matched_ids, :) = [];

clear('matched_ids');

%Remove duplicates within file1 and then sort

[b,m,n] = unique(idr23_1.idr23(:,1), 'rows');

clear('b', 'n');

idr23 = idr23_1.idr23(m, :);

clear('idr23_1', 'm');

%Convert longitude from +-180 to 0-360

negative_long = find(idr23(:, 7) < 0);

idr23(negative_long, 7) = idr23(negative_long, 7) + 360;

```

```

%Convert CAA from radians to degrees

idr23(:, 10) = idr23(:, 10) * (180/pi);

%Remove irregular scans so that the length remains divisible by 615
for scan_num = min(idr23(:, 15)):1:max(idr23(:, 15))
row_ids = find(idr23(:, 15) == scan_num);

if length(row_ids) ~= 615
idr23(row_ids, :) = [];

end

end

%Normalize the scan numbers

min_scan_number = min(idr23(:, 15));

idr23(:, 15) = idr23(:, 15) - min_scan_number + 1;

%Remove duplicate rows

[unique_scans,m,n] = unique(idr23(:,15), 'rows');

clear('m', 'n');

%Scan centers for IDR 37 GHz

idr_scan_center23 = [unique_scans ((unique_scans - 1) * 12.6) + 10];

%Compute MWR scan starts

mwr_scan_start = [0:13.1025:idr_scan_center23(length(idr_scan_center23), 2)]';

idr_start_scan = [];

%Eliminate skipped IDR scans

for mwr_scan_index = 1:length(mwr_scan_start)

    %find start idr scan

```

```

ids = find(idr_scan_center23(:,2) > mwr_scan_start(mwr_scan_index, 1));

idr_start_scan = [idr_start_scan;idr_scan_center23( ids(1,1), 1)];

end

[idr_start_scan,m,n] = unique(idr_start_scan(:,1), 'rows');

clear('m', 'n');

%Add the scan angle (Azimuth) as the last column to the IDR data
idr23 = [idr23 ((idr23(:, 16)*-0.005463639) + 3.71527479)*180/pi];

%Filter out invalid ids for 58 deg beams of 23 GHz frequency
valid_ids = find( (idr23(:,16) >= 558 & idr23(:,16) <= 562) | (idr23(:,16) >= 580 & idr23(:,16)
<= 584) | (idr23(:,16) >= 601 & idr23(:,16) <= 605) | (idr23(:,16) >= 623 & idr23(:,16) <= 627)
| (idr23(:,16) >= 641 & idr23(:,16) <= 645) | (idr23(:,16) >= 660 & idr23(:,16) <= 664) |
(idr23(:,16) >= 765 & idr23(:,16) <= 769) | (idr23(:,16) >= 743 & idr23(:,16) <= 747) |
(idr23(:,16) >= 724 & idr23(:,16) <= 728) | (idr23(:,16) >= 705 & idr23(:,16) <= 709));
idr23_58 = idr23(valid_ids,:);

%Filter out invalid ids for 52 deg beams of 23 GHz frequency
valid_ids = find( (idr23(:,16) >= 569 & idr23(:,16) <= 572) | (idr23(:,16) >= 591 & idr23(:,16)
<= 594) | (idr23(:,16) >= 612 & idr23(:,16) <= 615) | (idr23(:,16) >= 632 & idr23(:,16) <= 635)
| (idr23(:,16) >= 651 & idr23(:,16) <= 654) | (idr23(:,16) >= 756 & idr23(:,16) <= 759) |
(idr23(:,16) >= 735 & idr23(:,16) <= 738) | (idr23(:,16) >= 715 & idr23(:,16) <= 718) |
(idr23(:,16) >= 696 & idr23(:,16) <= 699));
idr23_52 = idr23(valid_ids,:);

clear('valid_ids', 'idr23');

clear('idr_scans','idr_scan_center23', 'mwr_scan_start', 'mwr_scan_index');

```

```

mwr_scan_width58 = 50;

mwr_pixel_width58 = 5;

mwr_scan_width52 = 36;

mwr_pixel_width52 = 4;

mwr_pixel_indices = [];

mwr_pixels_23 = [];

%Average MWR pixel fields along-scan only at 58 degs

%asc/dsc flag

temp = dec2bin(idr23_58(:,14),32);

ascdsc_flag58 = bin2dec(temp(:,23));

ascdscstuff =

mean(reshape(ascdsc_flag58(:,1),mwr_pixel_width58,numel(ascdsc_flag58(:,1))/mwr_pixel_wid
th58)',2);

ascdsc = reshape(ascdscstuff,10,numel(ascdscstuff)/10);

%TbV

tbvstuff =

mean(reshape(idr23_58(:,2),mwr_pixel_width58,numel(idr23_58(:,2))/mwr_pixel_width58)',2);

tbv = reshape(tbvstuff,10,numel(tbvstuff)/10);

```

```

tbhstuff =
mean(reshape(idr23_58(:,3),mwr_pixel_width58,numel(idr23_58(:,3))/mwr_pixel_width58)',2);
tbh = reshape(tbhstuff,10,numel(tbhstuff)/10)';

timestuff =
mean(reshape(idr23_58(:,1),mwr_pixel_width58,numel(idr23_58(:,1))/mwr_pixel_width58)',2);
time = reshape(timestuff,10,numel(timestuff)/10)';

latstuff =
median(reshape(idr23_58(:,6),mwr_pixel_width58,numel(idr23_58(:,6))/mwr_pixel_width58)',2
);
lat = reshape(latstuff,10,numel(latstuff)/10)';

lonstuff =
handle_transition_in_degrees(reshape(idr23_58(:,7),mwr_pixel_width58,numel(idr23_58(:,7))/m
wr_pixel_width58)', 0);
lon = reshape(lonstuff,10,numel(lonstuff)/10)';

azimuthstuff =
mean(reshape(idr23_58(:,19),mwr_pixel_width58,numel(idr23_58(:,19))/mwr_pixel_width58)',2
);
azimuth = reshape(azimuthstuff,10,numel(azimuthstuff)/10)';

```

```

caastuff =
handle_transition_in_degrees(reshape(idr23_58(:,10),mwr_pixel_width58,numel(idr23_58(:,10))
/mwr_pixel_width58)', 0);
caa = reshape(caastuff,10,numel(caastuff)/10)';

%If all the IDR pixels along scan, correspond to a common surface type
%the surface type of the result would be a specific type otherwise it would
%be -1
surfacestuff = reshape(idr23_58(:,17), mwr_pixel_width58,
numel(idr23_58(:,17))/mwr_pixel_width58)';
mwr_surface_type = [];
for i = 1:1:length(surfacestuff)
    %Land = 0
    %Not used = 1
    %Near coast = 2
    %Ice = 3
    %Possible ice = 4
    %Ocean = 5
    %coast = 6
    %spare = 7
    surface_type = -1;
ids = find(surfacestuff(i,:) == 0);
if length(ids) == mwr_pixel_width58

```

```

        surface_type = 0;
end

ids = find(surfacestuff(i,:) == 2);
if length(ids) == mwr_pixel_width58
    surface_type = 2;
end

ids = find(surfacestuff(i,:) == 3);
if length(ids) == mwr_pixel_width58
    surface_type = 3;
end

ids = find(surfacestuff(i,:) == 4);
if length(ids) == mwr_pixel_width58
    surface_type = 4;
end

ids = find(surfacestuff(i,:) == 5);
if length(ids) == mwr_pixel_width58
    surface_type = 5;
end

ids = find(surfacestuff(i,:) == 6);
if length(ids) == mwr_pixel_width58
    surface_type = 6;
end

ids = find(surfacestuff(i,:) == 7);

```

```

if length(ids) == mwr_pixel_width58
    surface_type = 7;
end

mwr_surface_type = [mwr_surface_type; surface_type];
end

surface = reshape(mwr_surface_type,10,numel(mwr_surface_type)/10)';

%Average MWR pixel fields along-scan only at 52 degs

temp = dec2bin(idr23_52(:,14),32);
ascdsc_flag52 = bin2dec(temp(:,23));

ascdscstuff =
mean(reshape(ascdsc_flag52(:,1),mwr_pixel_width52,numel(ascdsc_flag52(:,1))/mwr_pixel_wid
th52)',2);

ascdsc52 = reshape(ascdscstuff,9,numel(ascdscstuff)/9)';

%TbV

tbvstuff =
mean(reshape(idr23_52(:,2),mwr_pixel_width52,numel(idr23_52(:,2))/mwr_pixel_width52)',2);

tbv52 = reshape(tbvstuff,9,numel(tbvstuff)/9)';

tbhstuff =
mean(reshape(idr23_52(:,3),mwr_pixel_width52,numel(idr23_52(:,3))/mwr_pixel_width52)',2);

tbh52 = reshape(tbhstuff,9,numel(tbhstuff)/9)';

```

```

timestuff =
mean(reshape(idr23_52(:,1),mwr_pixel_width52,numel(idr23_52(:,1))/mwr_pixel_width52)',2);
time52 = reshape(timestuff,9,numel(timestuff)/9)';

latstuff =
median(reshape(idr23_52(:,6),mwr_pixel_width52,numel(idr23_52(:,6))/mwr_pixel_width52)',2
);
lat52 = reshape(latstuff,9,numel(latstuff)/9)';

lonstuff =
handle_transition_in_degrees(reshape(idr23_52(:,7),mwr_pixel_width52,numel(idr23_52(:,7))/m
wr_pixel_width52)', 0);
lon52 = reshape(lonstuff,9,numel(lonstuff)/9)';

azimuthstuff =
mean(reshape(idr23_52(:,19),mwr_pixel_width52,numel(idr23_52(:,19))/mwr_pixel_width52)',2
);
azimuth52 = reshape(azimuthstuff,9,numel(azimuthstuff)/9)';

caastuff =
handle_transition_in_degrees(reshape(idr23_52(:,10),mwr_pixel_width52,numel(idr23_52(:,10))
/mwr_pixel_width52)', 0);
caa52 = reshape(caastuff,9,numel(caastuff)/9)';

```

```

%If all the IDR pixels along scan, correspond to a common surface type
%the surface type of the result would be a specific type otherwise it would
%be -1

surfacestuff =

reshape(idr23_52(:,17),mwr_pixel_width52,numel(idr23_52(:,17))/mwr_pixel_width52)';

mwr_surface_type = [];

for i = 1:1:length(surfacestuff)

    %Land = 0

    %Not used = 1

    %Near coast = 2

    %Ice = 3

    %Possible ice = 4

    %Ocean = 5

    %coast = 6

    %spare = 7

    surface_type = -1;

ids = find(surfacestuff(i,:) == 0);

if length(ids) == mwr_pixel_width52

    surface_type = 0;

end

ids = find(surfacestuff(i,:) == 2);

if length(ids) == mwr_pixel_width52

```

```

        surface_type = 2;
end

ids = find(surfacestuff(i,:) == 3);
if length(ids) == mwr_pixel_width52
    surface_type = 3;
end

ids = find(surfacestuff(i,:) == 4);
if length(ids) == mwr_pixel_width52
    surface_type = 4;
end

ids = find(surfacestuff(i,:) == 5);
if length(ids) == mwr_pixel_width52
    surface_type = 5;
end

ids = find(surfacestuff(i,:) == 6);
if length(ids) == mwr_pixel_width52
    surface_type = 6;
end

ids = find(surfacestuff(i,:) == 7);
if length(ids) == mwr_pixel_width52
    surface_type = 7;
end

mwr_surface_type = [mwr_surface_type; surface_type];

```

end

```
surface_52 = reshape(mwr_surface_type,9,numel(mwr_surface_type)/9)';
```

%58

```
mwrpixlat= [];
```

```
mwrpixlon= [];
```

```
mwrpixtbv= [];
```

```
mwrpixtbh= [];
```

```
mwrpixtime= [];
```

```
mwrpixsurface = [];
```

```
mwrpixazimuth = [];
```

```
mwrpixcaa = [];
```

```
mwrpixscan = [];
```

```
mwrascdsc = [];
```

%52

```
mwrpixlat52= [];
```

```
mwrpixlon52= [];
```

```
mwrpixtbv52= [];
```

```
mwrpixtbh52= [];
```

```
mwrpixtime52= [];
```

```
mwrpixsurface52 = [];
```

```
mwrpixazimuth52 = [];
```

```
mwrpixcaa52 = [];
```

```

mwrpixscan52 = [];
mwrascdsc52 = [];

%For each IDR scan, excluding the skipped scans,
%average MWR pixel fields along-track at 58 & 52 degs.
%Since the along-scan of IDR pixels is at least 3, stop at the
%4th last IDR scan
for i = 1:length(idr_start_scan) - 3
    %Get scan sequence for 58 degs incidence angle
    [scanrange58] = get_current_scan_sequence(idr_start_scan(i,1), idr_start_scan, 3, length(lat));
    %Get scan sequence for 52 degs incidence angle
    [scanrange52] = get_current_scan_sequence(idr_start_scan(i,1), idr_start_scan, 2, length(lat));

    %Average IDR fields along-scan at 58 degs
    mwrpixlat = [mwrpixlat ; median(lat(scanrange58,:), 1)];
    mwrpixlon = [mwrpixlon ; handle_transition_in_degrees(lon(scanrange58,:), 1)];%lon is already
in radians
    mwrpixtbv = [mwrpixtbv ; mean(tbv(scanrange58,:), 1)];
    mwrpixtbh = [mwrpixtbh ; mean(tbh(scanrange58,:), 1)];
    mwrpixtime = [mwrpixtime ; mean(time(scanrange58,:), 1)];
    mwrpixazimuth = [mwrpixazimuth ; mean(azimuth(scanrange58,:), 1)];
    mwrpixcaa = [mwrpixcaa ; handle_transition_in_degrees(caa(scanrange58, :), 1)];
    mwrascdsc = [mwrascdsc ; mean(ascdsc(scanrange58,:), 1)];

```

```

scan = repmat(i, 1, 10);

mwrpixscan = [mwrpixscan ; scan];

%Average IDR fields along-scan at 52 degs

mwrpixlat52 = [mwrpixlat52 ; median(lat52(scanrange52,:), 1)];

mwrpixlon52 = [mwrpixlon52 ; handle_transition_in_degrees(lon52(scanrange52,:), 1)];%lon is
already in radians

mwrpixtbv52 = [mwrpixtbv52 ; mean(tbv52(scanrange52,:), 1)];

mwrpixtbh52 = [mwrpixtbh52 ; mean(tbh52(scanrange52,:), 1)];

mwrpixtime52 = [mwrpixtime52 ; mean(time52(scanrange52,:), 1)];

mwrpixazimuth52 = [mwrpixazimuth52 ; mean(azimuth52(scanrange52,:), 1)];

mwrpixcaa52 = [mwrpixcaa52 ; handle_transition_in_degrees(caa52(scanrange52, :), 1)];

mwrascdsc52 = [mwrascdsc52 ; mean(ascdsc52(scanrange52,:), 1)];

scan = repmat(i, 1, 9);

mwrpixscan52 = [mwrpixscan52 ; scan];

%If all the IDR pixels along track, correspond to a common surface type

%the surface type of the result would be a specific type otherwise it would

%be -1

ids58 = [];

ids52 = [];

surface_type58 = -1;

surface_type52 = -1;

```

```

for j = 1:1:10
    %Surface type at 58
temp = find(surface( scanrange58, j) == 0);
if length(temp) == length(scanrange58)
    surface_type58 = 0;
end
temp = find(surface( scanrange58, j) == 2);
if length(temp) == length(scanrange58)
    surface_type58 = 2;
end
temp = find(surface( scanrange58, j) == 3);
if length(temp) == length(scanrange58)
    surface_type58 = 3;
end
temp = find(surface( scanrange58, j) == 4);
if length(temp) == length(scanrange58)
    surface_type58 = 4;
end
temp = find(surface( scanrange58, j) == 5);
if length(temp) == length(scanrange58)
    surface_type58 = 5;
end
temp = find(surface( scanrange58, j) == 6);

```

```

if length(temp) == length(scanrange58)
    surface_type58 = 6;
end
temp = find(surface( scanrange58, j) == 7);
if length(temp) == length(scanrange58)
    surface_type58 = 7;
end

ids58 = [ids58 surface_type58];

%Surface type at 52
if j < 10
temp = find(surface( scanrange52, j) == 0);
if length(temp) == length(scanrange52)
    surface_type52 = 0;
end
temp = find(surface( scanrange52, j) == 2);
if length(temp) == length(scanrange52)
    surface_type52 = 2;
end
temp = find(surface( scanrange52, j) == 3);
if length(temp) == length(scanrange52)
    surface_type52 = 3;
end

```

```

temp = find(surface( scanrange52, j) == 4);
if length(temp) == length(scanrange52)
    surface_type52 = 4;
end
temp = find(surface( scanrange52, j) == 5);
if length(temp) == length(scanrange52)
    surface_type52 = 5;
end
temp = find(surface( scanrange52, j) == 6);
if length(temp) == length(scanrange52)
    surface_type52 = 6;
end
temp = find(surface( scanrange52, j) == 7);
if length(temp) == length(scanrange52)
    surface_type52 = 7;
end
ids52 = [ids52 surface_type52];
end
end
mwrpixsurface = [mwrpixsurface;ids58];
mwrpixsurface52 = [mwrpixsurface52;ids52];
end

```

```

%Reshape the MWR fields as column vectors

%Reshape into column vectors at 58

[m n] = size(mwrpixonlat);

mwrpixonlat = reshape(mwrpixonlat, m * n , 1 );

[m n] = size(mwrpixonlon);

mwrpixonlon = reshape(mwrpixonlon, m * n , 1 );

negative_long = find (mwrpixonlon(:, 1) > 180);

mwrpixonlon(negative_long, 1) = mwrpixonlon(negative_long, 1) - 360;

[m n] = size(mwrpixonlatbv);

mwrpixonlatbv = reshape(mwrpixonlatbv, m * n , 1 );

[m n] = size(mwrpixonlatbh);

mwrpixonlatbh = reshape(mwrpixonlatbh, m * n , 1 );

[m n] = size(mwrpixonlattime);

mwrpixonlattime = reshape(mwrpixonlattime, m * n , 1 );

[m n] = size(mwrpixonlatheight);

mwrpixonlatheight = reshape(mwrpixonlatheight, m * n , 1 );

[m n] = size(mwrpixonlatazimuth);

mwrpixonlatazimuth = reshape(mwrpixonlatazimuth, m * n , 1 );

[m n] = size(mwrpixonlatca);

mwrpixonlatca = reshape( mwrpixonlatca, m * n , 1 )*(pi/180);

[m n] = size(mwrpixonlatscan);

mwrpixonlatscan = reshape( mwrpixonlatscan, m * n , 1 );

[m n] = size(mwrpixonlatdsc);

```

```

mwrascdsc = reshape( mwrascdsc, m * n , 1 );
mwrascdsc(find(mwrascdsc(:, 1) < 0.5), 1) = 0;
mwrascdsc(find(mwrascdsc(:, 1) >= 0.5), 1) = 1;
%Incidence angle adjustment at 23 V 58
bias_23_58_V = -0.0379*mwrpixtbv + 18.62;
%A columns vector of -9999 to represent invalid data
no_data = repmat(-9999, m * n , 1 );
incidence_angle = repmat(58, m*n, 1);
%MWR pixels at 58 degs
mwr_pixels_23_58 = [mwrpixtime mwrpixlat mwrpixlon mwrpixtbv mwrpixtbh no_data
no_data mwrpixsurface mwrpixazimuth mwrpixcaa bias_23_58_V no_data incidence_angle
mwrpixscan mwrascdsc];

%Reshape the MWR fields as column vectors
%Reshape into column vectors at 52
[m n] = size(mwrpixlat52);
mwrpixlat52 = reshape(mwrpixlat52, m * n , 1 );
[m n] = size(mwrpixlon52);
mwrpixlon52 = reshape(mwrpixlon52, m * n , 1 );
negative_long = find (mwrpixlon52(:, 1) > 180);
mwrpixlon52(negative_long, 1) = mwrpixlon52(negative_long, 1) - 360;
[m n] = size(mwrpixtbv52);
mwrpixtbv52 = reshape(mwrpixtbv52, m * n , 1 );

```



```

mwr_pixels_23_52 = [mwrpixtime52 mwrpixlat52 mwrpixlon52 mwrpixtbv52 mwrpixtbh52
no_data no_data mwrpixsurface52 mwrpixazimuth52 mwrpixcaa52 bias_23_52_V no_data
incidence_angle mwrpixscan52 mwrascdsc52];

%MWR pixels at 23

mwr_pixels_23 = [mwr_pixels_23_58; mwr_pixels_23_52];

mwr_pixels_23 = sortrows(mwr_pixels_23, 1);

%Add beam # to each MWR pixel

mwr_pixels_23 = [mwr_pixels_23 repmat(0, length(mwr_pixels_23), 1)];

%Beam # vector

%Beam1

ids_beam1 = find(mwr_pixels_23(:, 9) >= 37.5 & mwr_pixels_23(:, 9) <= 37.6);
mwr_pixels_23(ids_beam1, 16) = 1;

%Beam2

ids_beam2 = find(mwr_pixels_23(:, 9) >= 34.2 & mwr_pixels_23(:, 9) <= 34.3);
mwr_pixels_23(ids_beam2, 16) = 2;

%Beam3

ids_beam3 = find(mwr_pixels_23(:, 9) >= 30.6 & mwr_pixels_23(:, 9) <= 30.7);
mwr_pixels_23(ids_beam3, 16) = 3;

%Beam4

ids_beam4 = find(mwr_pixels_23(:, 9) >= 27.3 & mwr_pixels_23(:, 9) <= 27.4);
mwr_pixels_23(ids_beam4, 16) = 4;

%Beam5

```

```

ids_beam5 = find(mwr_pixels_23( :, 9) >= 24.1 & mwr_pixels_23( :, 9) <= 24.2);
mwr_pixels_23(ids_beam5, 16) = 5;

%Beam6

ids_beam6 = find(mwr_pixels_23( :, 9) >= 20.8 & mwr_pixels_23( :, 9) <= 20.9);
mwr_pixels_23(ids_beam6, 16) = 6;

%Beam7

ids_beam7 = find(mwr_pixels_23( :, 9) >= 17.2 & mwr_pixels_23( :, 9) <= 17.3);
mwr_pixels_23(ids_beam7, 16) = 7;

%Beam8

ids_beam8 = find(mwr_pixels_23( :, 9) >= 14.5 & mwr_pixels_23( :, 9) <= 14.6);
mwr_pixels_23(ids_beam8, 16) = 8;

%Beam9

ids_beam9 = find(mwr_pixels_23( :, 9) >= 11.5 & mwr_pixels_23( :, 9) <= 11.6);
mwr_pixels_23(ids_beam9, 16) = 9;

%Beam10

ids_beam10 = find(mwr_pixels_23( :, 9) >= 8.6 & mwr_pixels_23( :, 9) <= 8.7);
mwr_pixels_23(ids_beam10, 16) = 10;

%Beam11

ids_beam11 = find(mwr_pixels_23( :, 9) >= 5.6 & mwr_pixels_23( :, 9) <= 5.7);
mwr_pixels_23(ids_beam11, 16) = 11;

%Beam12

ids_beam12 = find(mwr_pixels_23( :, 9) >= -5.5 & mwr_pixels_23( :, 9) <= -5.4);
mwr_pixels_23(ids_beam12, 16) = 12;

```

```

%Beam13

ids_beam13 = find(mwr_pixels_23(:, 9) >= -8.5 & mwr_pixels_23(:, 9) <= -8.4);

mwr_pixels_23(ids_beam13, 16) = 13;

%Beam14

ids_beam14 = find(mwr_pixels_23(:, 9) >= -11.5 & mwr_pixels_23(:, 9) <= -11.4);

mwr_pixels_23(ids_beam14, 16) = 14;

%Beam15

ids_beam15 = find(mwr_pixels_23(:, 9) >= -14.45 & mwr_pixels_23(:, 9) <= -14.35);

mwr_pixels_23(ids_beam15, 16) = 15;

%Beam16

ids_beam16 = find(mwr_pixels_23(:, 9) >= -17.7 & mwr_pixels_23(:, 9) <= -17.6);

mwr_pixels_23(ids_beam16, 16) = 16;

%Beam17

ids_beam17 = find(mwr_pixels_23(:, 9) >= -20.4 & mwr_pixels_23(:, 9) <= -20.3);

mwr_pixels_23(ids_beam17, 16) = 17;

%Beam18

ids_beam18 = find(mwr_pixels_23(:, 9) >= -24.3 & mwr_pixels_23(:, 9) <= -24.2);

mwr_pixels_23(ids_beam18, 16) = 18;

%Beam19

ids_beam19 = find(mwr_pixels_23(:, 9) >= -27.3 & mwr_pixels_23(:, 9) <= -27.2);

mwr_pixels_23(ids_beam19, 16) = 19;

%Save MWR 23 Ghz file

idr23_52 = [];

```

```
idr23_58 = [];  
mwr_file_name = ['mwr_fullswath', file1(end-23:end)];  
save(['x:\IDR_SDR_DATA\', mwr_file_name], 'mwr_pixels_23');  
toc
```

```

%Get the IDR scan sequence to be averaged for the current MWR scan
function [actual_sequence] = get_current_scan_sequence(start_scan, idr_start_scans,
numofscans, maxscan)
%Desired scan sequence
desired_sequence = [start_scan:1:numofscans];
%Actual scan sequence will be different from desired sequence
%because of scan skips
actual_sequence = [];
for i = start_scan:1:start_scan + numofscans
id = find(idr_start_scans(:,1) == i);
if isempty(id) == 0 & idr_start_scans(id) <= maxscan
actual_sequence = [actual_sequence; idr_start_scans(id)];
end
end
end

```

```
%Handles the transition from 180 to -180 degs in longitudes
```

```
%Also handles the transition of CAA from 0 to 360
```

```
function [result] = handle_transition_in_degrees(m,colwise)
```

```
[r, c] = size(m);
```

```
result = [];
```

```
if colwise == 0
```

```
    even_check = mod(c, 2);
```

```
for i = 1:r
```

```
    min_angle = min(m(i,:));
```

```
    max_angle = max(m(i,:));
```

```
    if max_angle - min_angle > 180
```

```
        if even_check == 0
```

```
            result(i, :) = median(m(i, 2:c), 2);
```

```
        else
```

```
            result(i, :) = median(m(i, :), 2);
```

```
        end
```

```
    else
```

```
        result(i, :) = median(m(i, :), 2);
```

```
    end
```

```
end
```

```
else
```

```
    even_check = mod(r, 2);
```

```
for i = 1:c
```

```
min_angle = min(m(:,i));  
max_angle = max(m(:,i));  
if max_angle - min_angle > 180  
if even_check == 0  
result(:, i) = median(m(2:r, i), 1);  
else  
result(:, i) = median(m(:, i), 1);  
end  
else  
result(:, i) = median(m(:, i), 1);  
end  
end  
end  
end
```

LIST OF REFERENCES

- [1] Y. K. A. Sen, D. Caruso, G. Lagerloef, R. Colomb, D. Le Vine, and S. Yueh, "Aquarius/SAC-D mission overview," in *Proc. SPIE Conf. Sens., Syst., and Next-Generation Satellites X*, Stockholm, Sweden, 2006, p. 63610I.
- [2] P. W. S. G. Gaiser, K.M.; Twarog, E.M.; Poe, G.A.; Purdy, W.; Richardson, D.; Grossman, W.; Jones, W.L.; Spencer, D.; Golba, G.; Cleveland, J.; Choy, L.; Bevilacqua, R.M.; Chang, P.S., "The WindSat spaceborne polarimetric microwave radiometer: sensor description and early orbit performance," *Geoscience and Remote Sensing, IEEE Transactions on* vol. 42, pp. 2347-2361, Nov. 2004.
- [3] F. R. C. Gary Lagerloef, David Le Vine , Frank Wentz , Simon Yueh, Christopher Ruf, Jonathan Lilly, John Gunn , Yi Chao, and G. F. Annettede Charon , and Calvin Swift, "The Aquarius/SAC-D Mission Designed to Meet the Salinity Remote Sensing Challenge," *Oceanography*, vol. 21, 2008.
- [4] "Annual Mean Salinity (PSS) at the Surface," World Ocean Atlas 2001, Ocean Climate Laboratory/NODC.
- [5] F. R. C. Gary Lagerloef, "The Aquarius/SAC-D Satellite Mission," in *South Atlantic Workshop*, Buenos Aires, 7-10 May, 2007.
- [6] F. David M. Le Vine, IEEE, Gary S. E. Lagerloef, Fernando Raúl Colomb, Simon H. Yueh, Senior Member, IEEE, and Fernando A. Pellerano, Senior Member, IEEE, "Aquarius: An Instrument to Monitor Sea Surface Salinity From Space," *Geoscience and Remote Sensing*, vol. 45, pp. 2040-2050, July 2007.
- [7] W. J. W. Simon H. Yueh, Wendy Edelstein, Don Farra, Mark Johnson, Fernando Pellerano, David LeVine, and Peter Hilderbrand, "Aquarius Instrument Design For Sea Surface Salinity Measurements," *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings*, vol. 4, pp. 2795- 2797, 21-25 July 2003 2003.
- [8] D. M. L. Le Vine, G.; Pellerano, F.; Colomb, F.R., "The Aquarius/SAC-D mission and status of the Aquarius instrument," in *Microwave Radiometry and Remote Sensing of the Environment, 2008. MICRORAD 2008* Firenze, pp. 1 - 4.
- [9] "<http://aquarius.nasa.gov/pdfs/instrument.pdf>," NASA, p. Aquarius Selected Instrument Concept.
- [10] J. C. Gallo, "Aquarius/SAC-D Microwave Radiometer Critical Design Review," Buenos Aires, Argentina, August 22-24, 2007.
- [11] G. L. David M. Le Vine, "http://smap.jpl.nasa.gov/files/0352_LeVine.pdf," NASA, Jet Propulsion Laboratory, pp. Aquarius: Active/Passive L-band Remote Sensing from Space.
- [12] P. G. a. t. W. T. Mike Bettenhausen, "WindSat Data Products and Assimilation," in *Workshop on Applications of Remotely Sensed Observations in Data Assimilation* University of Maryland, 2007.
- [13] "Satellite Toolkit - <http://www.stk.com/>."
- [14] L. Hong, "Inter-Satellite Microwave Radiometer Calibration," in *Electrical Engineering and Computer Science*. vol. PhD Orlando: University of Central Florida, 2008, p. 166.

- [15] L. N. C. Connor, P.S.; Jelenak, Z.; Wang, N.-Y.; Mavor, T.P., "WindSat validation datasets: an overview," in *Geoscience and Remote Sensing Symposium*, 2004, pp. 386-389.
- [16] "http://www.cpi.com/twiki/pub/WindSat/WindSatNews/EDR_Format_c202.pdf."
- [17] W. L. Jones, "Conical Scanning Sensor Geometry," in *EEL 5432 Satellite Remote Sensing* Orlando: University of Central Florida.
- [18] R. K. M. F. T. Ulaby, and A.K.Fung, *Microwave Remote Sensing--Active and passive: Fundamentals and Radiometry* vol. 1: Artech House Inc., 1981.
- [19] R. K. M. F. T. Ulaby, and A. K. Fung, "Microwave Remote Sensing, vol 3. Appendix E," 1981, pp. 2017-2025.
- [20] C. Elachi, *Introduction to the Physics and Techniques of Remote Sensing*: John Wiley & Sons, Inc., New York, 1987.