



CHALMERS



Beam Selection Optimization Algorithm for the Satcube Ku.

Master's thesis at the Department of Space, Earth and Environment

Lucas Olsson
Pouria Saber

MASTER'S THESIS IN SPACE, EARTH AND ENVIRONMENT -
SEEX30

Beam Selection Optimization Algorithm for the Satcube Ku terminal.

A study of the parameters affecting the performance of the Satcube
Ku terminal and the design of a algorithm to determine the best
beam available.

Lucas Olsson
Pouria Saber



CHALMERS

Department of space earth and environment
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023

Beam Selection Optimization Algorithm for the Satcube Ku terminal.
A study of the parameters affecting the performance of the Satcube Ku terminal
and the design of a algorithm to determine the best beam available.

Lucas Olsson
Pouria Saber

© Lucas Olsson, Pouria Saber
2023.

Examinator: Rudiger Haas, Department of space earth and environment

Handledare: Jan Johansson, Department of Space, Earth and Environment
Handledare: Calle Malm, Satcube

Master's thesis SEEX30
Department of space earth and environment
Chalmers University of Technology
SE-412 96 Gothenburg

Cover picture: SatCube Ku in action at a vulcano eruption taken from Yellon.

Abstract

The Satcube Ku is a satellite terminal which provides internet access to a user anywhere on Earth with the help of spot beam technology. For a given position on Earth, multiple intersecting beams will be present with varying signal strength. There is the need to distinguish one beam from another in order to establish a hierarchy. The purpose of this master thesis is to investigate a way to rank the available beams at a position on Earth and develop that into a new beam selection algorithm for the Satcube Ku terminal. The new algorithm must address both circular and elliptical HTS-beams as the beams are projected across the surface of the Earth.

There are many limitations to this project and many different approaches have been presented. The approach is to analyze user data and geospatial data combined with provided satellite information in order to find important aspects of the problem. A five step algorithm was ultimately decided. The algorithm takes the user position as an input and gives a sorted list from highest to lowest carrier-to-noise expected in that position as output.

The steps are: Step 1: Position and all available beams, Step 2: Beamcenter of beams, Step 3: CNR value at beamcenter, Step 4: CNR reduction to the user position, Step 5: CNR arranged list based on projection and accuracy. The step 1 was solved using polygon conversion of the beam contours given by the terminal and then calculating which polygons the user was inside of. In step 2 the beamcenter of the beams were estimated by measuring the distance from the polygon center to a beam center position provided by Intelsat for certain beams. In step 3 the each beamcenter was given a maximum theoretical value also provided by Intelsat. In step 4 three different methods were investigated to project what the received CNR would be at the user position. The methods were Concentric Contour Contraction, Concentric Elliptical Expansion and Least square fit method. The Concentric Elliptical expansion was the most appropriate method available and was used in step 5 for a test position in Gothenburg.

The resulting output was limited to only IS33 and IS37 satellites due to the lack of information regarding the beam center values. The final list projections compared to actual measurements performed within 1 dB.

Keywords: Satellite communication, Algorithm, Python, Data analysis, High-throughput satellite, Satcube Ku

Acknowledgements

This master thesis was carried out at Satcube AB and Chalmers university of technology. We would like to express our gratitude towards our supervisor at Chalmers, Jan Johansson, for the possibility to carry out this thesis and for the great support. Furthermore, we would like to extend our gratitude to Satcube AB for the opportunity to work on a realistic problem with their product. Specifically, we would like to express our appreciation to our supervisor at Satcube, Calle Malm and for his great support and guidance. Finally, we would also like to say thank you to everyone at Satcube who helped us in our project and made us feel welcome at their company.

Lucas Olsson, Gothenburg, June 2023 Pouria Saber, Gothenburg, June 2023

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

HTS	High-throughput satellite
C/N	Carrier-to-noise ratio
S/N	Signal-to-noise ratio
LO	Local oscillator
RF	Radio-frequency
IF	Intermediate-frequency
OTA	Over-the-air
Modem	Modulator-demodulator
FSS	Fixed-satellite service

Contents

List of Acronyms	vii
List of Figures	xiii
List of Tables	xvii
1 Introduction	1
1.1 Background	1
1.2 Problem description	1
1.3 Objectives	2
1.4 Limitations	2
1.5 Disposition	2
2 Theory	3
2.1 Beams and satellites	3
2.2 Python	3
2.2.1 Packages	3
2.3 Geospatial analysis	3
2.4 Transmission equations	5
2.4.1 Carrier to noise ratio	5
2.5 Antenna Characteristics	5
2.5.1 Beamwidth and footprint	5
2.5.2 Polarization	5
2.5.3 HTS beam	7
2.5.4 Beam Center	7
2.6 Satcube Ku	9
2.6.1 Operation and usage	9
2.6.2 Initial Limitations	11
2.6.3 OTA-files	12
2.6.4 Polygon center	13
2.6.5 Weather attenuation	14
2.6.6 Sweden's meteorological and hydrological institute	14
3 Methods	15
3.1 Planning and preliminary investigation	15
3.2 Beam center	15
3.2.1 Starting the terminal	16

3.2.2	Beam shape and analysis	16
3.2.3	Correction	17
3.2.4	Carrier-to-noise estimation at the beam center	19
3.3	Data analysis and testing	19
3.3.1	Available data	19
3.3.2	Analysis method and cleanup	20
3.3.3	Setting up analysis	20
3.3.4	Analysis with SMHI	21
3.3.5	Parameter targets	21
3.3.5.1	Terminal difference	21
3.3.5.2	Azimuth misalignment	21
3.3.6	Losses and uncertainties in the recorded user data file	22
3.4	Beam algorithm	22
3.4.1	Step 1: Position and all available beams	22
3.4.2	Step 2: Beamcenter of beams	22
3.4.3	Step 3: CNR value at beamcenter	23
3.4.4	Step 4: CNR reduction to the user position	23
3.4.5	Step 5: CNR arranged list based on projection and accuracy	23
3.4.6	Comments	23
4	Results	25
4.1	User data analysis	25
4.1.1	Terminal difference	25
4.1.2	Azimuth difference	29
4.2	Beam algorithm	33
4.2.1	Step 1: Position and all available beams	33
4.2.2	Step 2: Beamcenter of beams	33
4.2.3	Step 3: CNR value at beamcenter	37
4.2.3.1	Beam footprint providers	37
4.2.3.2	User data	37
4.2.3.3	The IntelSat BC relative file	38
4.2.4	Step 4: CNR reduction to the user position	38
4.2.4.1	Method 1 : Concentric Contour contraction or CCC method	38
4.2.4.2	Method 2 : Concentric Elliptical Expansion or CEE method	39
4.2.4.3	Method 3 : Least Square Fit or LSF method	45
4.2.5	Step 5: CNR sorted list and comparison with the current algorithm	46
5	Conclusion	51
5.1	User data analysis	51
5.1.1	Terminal difference	51
5.1.2	Azimuth difference	52
5.2	Beam algorithm	53
5.2.1	Step 1: Position and all available beams	53
5.2.1.1	Problems	53

5.2.1.1.1	Contours	53
5.2.1.1.2	Meridian crossing	53
5.2.2	Step 2: Beamcenter of beams	55
5.2.3	Step 3: CNR value at beamcenter	55
5.2.4	Step 4: CNR reduction to the user position	56
5.2.4.1	Method 1 : Concentric Contour contraction or CCC method	56
5.2.4.2	Method 2 : Concentric Elliptical Expansion or CEE method	56
5.2.4.3	Method 3 : Least Square Fit or LSF method	57
5.2.5	Step 5: CNR sorted list and comparison with the current al- gorithm	57
5.2.6	For future projects	58
5.3	Conclusions and areas of improvement	58
Bibliography		59

List of Figures

2.1	Three types of polarization. [4]	6
2.2	Representation of the difference between regular FSS satellite and HTS satellite coverage. Picture taken from Telenorsat. [8]	7
2.3	Beam contours of beam U36 from IS33 according to SatBeams. The center contour with the blue marker has a estimated EIRP of 55.6 dbW. For each expanded contour outwards the EIRP is reduced by 1 dbW.	8
2.4	Satcube Ku terminal.	9
2.5	Pointing instruction for the Satcube Ku	10
2.6	Terminal user interface	10
2.7	Satcube Ku technical specification taken from the Satcube Ku hand-book.	11
2.8	Satcube Ku environmental limitations.	11
2.9	HTS Beam map of the IS33e satellite, taken from Intelsat's website [1].	12
2.10	a: An example of the contours of a HTS beam from the IntelSat IS33, Beam ID 22 spanning over Mongolia. b: The polygon equivalent of the contour points with the polygon center and beam center included.	13
2.11	Attenuation due to rain for different frequencies according to Everything RF website.[7]	14
3.1	Example contour of a beam in Python. Longitude on the x-axis and latitude on the y-axis.	16
3.2	Distance to satellite vs distance error to correct beam center.	18
4.1	User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 0.8 West. Each point is a user connection to a satellite placed 1 west with the registered frequency below and the terminal ID above.	26
4.2	User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 1 West. Each point is a user connection to a satellite placed 1 west with the registered frequency below and the terminal ID above.	26
4.3	User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 18 East. Each point is a user connection to the IS37 satellite with the registered frequency below and the terminal ID above.	27

4.4	User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 60 East. Each point is a user connection to the IS33 satellite with the registered frequency below and the terminal ID above.	27
4.5	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 1 with the frequency 10,16538 GHz.	29
4.6	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 29 with the frequency 10,476 GHz.	30
4.7	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 40 with the frequency 10,476 GHz.	30
4.8	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 40 with the frequency 18,1639 GHz.	31
4.9	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 52 with the frequency 13,99149 GHz.	31
4.10	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 230.	32
4.11	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 1.	32
4.12	The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 24.	33
4.13	Gothenburg beam 1.	34
4.14	Gothenburg beam 2.	34
4.15	Gothenburg beam 3.	34
4.16	Gothenburg beam 4.	34
4.17	Gothenburg beam 5.	34
4.18	Gothenburg beam 6.	34
4.19	Gothenburg beam 7.	34
4.20	Gothenburg beam 8.	34
4.21	Gothenburg beam 9.	34
4.22	Distance to satellite vs distance error to correct beam center, with curve fit.	35
4.23	Gothenburg beam 1.	36
4.24	Gothenburg beam 2.	36
4.25	Gothenburg beam 3.	36
4.26	Gothenburg beam 4.	36
4.27	Gothenburg beam 5.	36
4.28	Gothenburg beam 6.	36
4.29	Gothenburg beam 7.	36
4.30	Gothenburg beam 8.	36
4.31	Gothenburg beam 9.	36
4.32	Concentric Contour Contraction of the beam 33 of IS33 with beam center as blue marker.	38
4.33	Concentric Elliptical Expansion of the beam 33 of IS33 with averaged user data for each ellipse.	39
4.34	Concentric Elliptical Expansion of the beam 33 of IS33 with averaged user data for each ellipse.	40

4.35	Concentric Elliptical Expansion of the beam 29 of IS33 averaged user data for each ellipse.	40
4.36	Concentric Elliptical Expansion of the beam 29 of IS33 averaged user data for each ellipse.	41
4.37	Concentric Elliptical Expansion of the beam 28 of IS33 with no user data.	41
4.38	Concentric Elliptical Expansion of the beam 28 of IS33 with no user data, a closer look.	42
4.39	Concentric Elliptical Expansion of the beam 1 of IS37 averaged user data for each ellipse.	42
4.40	Concentric Elliptical Expansion of the beam 2 of IS37.	43
4.41	Concentric Elliptical Expansion of the beam 4 of IS37.	43
4.42	Basic fit for the averaged values in beam 29 of IS33.	44
4.43	Basic fit for the averaged values in beam 33 of IS33.	44
4.44	Basic fit for the averaged values in beam 1 of IS37.	45
4.45	Illustration of the idea behind method 3 using Least Square Fit. . . .	46
4.46	Gothenburg test position in beam 1 of IS37 with CEE, 7 ellipses from the center.	47
4.47	Gothenburg position in beam 2 of IS37 with CEE, just about 8 ellipses out from the center.	47
4.48	Gothenburg position in beam 4 of IS37 with CEE, with more than 10 ellipses out.	48
4.49	Gothenburg position in beam 33 of IS33 with CEE, 5 ellipses from the center.	48
4.50	Gothenburg position in beam 29 of IS33 with CEE, 12 ellipses from the center.	49
5.1	Example of an unreliable beam contour	53
5.2	a: An example of a split HTS-beam. b: The same beam shifted around the prime meridian	55

List of Tables

4.1	Measured data at the Almedal office in Gothenburg with several terminals.	28
4.2	The predicted carrier to noise based on the CEE-method for a test position in Gothenburg, a sorted list can be made by changing the Beam 1 and Beam 2 positions based on the predicted values.	49
4.3	The table of beams available to the user in the test position in Gothenburg in the current beam algorithm.	49
4.4	The predicted carrier to noise based on the CEE-method some user positions inside some random beams to thest the algorithm outside of the Gothenburg area.	50

1

Introduction

1.1 Background

Satcube is a Gothenburg based company specializing in satellite communication. Example of stakeholders range from civil emergency workers to military personnel in circumstances where reliable communication is needed. Their main product is the Satcube Ku which will be the focus of this project. The Satcube Ku is a portable Wi-Fi hotspot that connects to the internet through Geostationary satellites. The connection is made primarily through High-throughput satellites (HTS). These satellites, unlike conventional satellites, utilizes a spot beam technology which focuses multiple beams to different geographically restricted areas. With this technique larger power can be focused to a certain spot allowing for higher data rate and smaller user terminals.

A stable and powerful connection is of most importance to Satcubes clients. Different clients may have different needs in term of the connection, for an example, journalists may need better upload speed to stream video while other clients may be in need of a higher download speed. Finding the most optimal beam is of great importance as the terminal is manually pointed towards the satellite by the user. It is therefore of great importance for Satcube to always make sure they are able to deliver the best service.

1.2 Problem description

With multiple HTS's in space more beams are created, thus leading to a high variety of beams in a given area. Therefore a problem arises in which beam to choose. The different beams will have different performance due to many factors. The optimal choice is always the beam with the best performance for the user. A beam with best performance is primarily defined by the Carrier-to-noise ratio (C/N). The beam is affected by factors such as: the distance between the satellite and the terminal, the elevation- and azimuth angle, atmospheric attenuation, transmitter strength and more.

The current algorithm implemented in the Satcube Ku terminal is functional but not optimal. It is an intricate algorithm that functions by selecting the beam which has its polygon center closest to the user. However, there are limitations to the algorithm such as.

- The modem in the terminal has no information of the beam strength
- The beam contours provided by the modem are inconsistent in their gradient levels.
- The beam center is not a 1d point, but rather a 2d area.

The primary problem is that the modem in the terminal has no information of the strength of the beam. This is a problem because the algorithm might suggest a beam which is closer than another beam, but weaker in strength. This could affect the performance significantly for users in field.

Beam contours has gradients due to the power distributed within the beam. A beam can be estimated with multiple contours forming a gradient of fading power level in relation to distance to the beam center. The attenuation is caused partly due to path loss and the distribution of power within the beam, more power is focused in the center and the power fades circularly. This problem is related to the first problem where the carrier-to-noise ratio is unknown.

The beam center area is determined to be the first gradient contour of a given beam where the signal strength is the strongest, this would be a more accurate representation.

1.3 Objectives

The objective with this project is to investigate methods to optimize the beam selection algorithm in the Satcube Ku. By combining prior knowledge of the current beam algorithm, calculations and data analysis methods of optimization should be investigated and deemed useful for a new beam selection algorithm.

1.4 Limitations

The focus is to develop the algorithm in Python. Translations to Rust will not be implemented as it is out of the scope of this project. Neither of us possess knowledge of the programming language Rust which the terminal utilizes. Implementation of the algorithm within the terminal was deemed too difficult and time demanding.

1.5 Disposition

The report will cover all of the details in the project. The theory chapter will include the theoretical background of every process. In the method chapter every step in the project will be documented in separated sections describing the breakdown of each problem. In the result and discussion the outcome of each section in the project will be showcased and discussed. A conclusion of the project in its entirety will also be presented.

2

Theory

In this chapter, we will provide a brief background, history, and theory behind the effects and methods that will be used in the project.

2.1 Beams and satellites

Beams transmitted from the satellites are projected in many different ways on the Earth surface, this is due to different distances traveled, elevation and azimuth location of the satellite and more. According to the website Satbeams the footprint of the beams that are projected directly below the satellite are circles [3]. Directly below the satellite equals to an elevation angle of 90° . When deviating from this point, which is also called the nadir point, the beam will be distorted proportionally to the distance from the nadir. This distance from the point can be described with the GEO-stationary distance and angles from the new point.

2.2 Python

Python is a high-level programming language which is designed to be easy to use. The syntax is easy to read and write making it a user friendly language. It is an object oriented language which is helpful when analyzing data, building scripts and creating applications. In this project python will be used to handle and analyze data and also be used to write the script for the beam algorithm.

2.2.1 Packages

Python has a vast amount of packages that can be installed and imported. Packages contain functions which serve different purposes. The most basic packages like NumPy and Matplotlib contain fundamental mathematical functions and analytical tools for data handling. A core package that will be utilized is Pandas. Pandas is an intricate but easy to use package that is important when dealing with data. It offers the ability to efficiently read excel files and create dataframes in order to process.

2.3 Geospatial analysis

As the earth is a 3d geoid, working with geospatial data is more complex compared to working with 2d-planes and cartesian coordinates.

Geospatial data is mostly used in this project, that is, data that is described and connected to a latitude and longitude position. Geospatial data is difficult to process with conventional tools as the geographic coordinate system is not similar to Cartesian coordinates. There are a lot of packages in python which are helpful when dealing with geospatial data. Some of these packages are: Geopandas, GEOjson, and geoplot. Geopandas is similar to Pandas but specifically programmed to handle geospatial data. Geoplot is a library that is used for geospatial visualization. Shapely is an additional core packages that offers functions to deal with any geometrical shape. Beam contours are difficult to handle due to the nature of the shapes, making calculations on uneven shapes is difficult, shapely is therefore a very efficient solution to this problem. It is worth noting that all calculations assume a spherical earth and not a geoid. This means that the calculations base around a uniform radius. All the geostationary satellites are located at the equator with a latitude of 0° .

To calculate the distance from a certain point on earth to a satellite in geostationary orbit spherical trigonometry had to be utilized. The distance is dependant on several geographical factors. The equation is a spherical implementation of the Pythagorean theorem.

$$d = \sqrt{(R_s - R_e \cos \lambda \cos \phi)^2 + (R_e \sin \lambda \cos \phi)^2 + (R_e \sin \phi)^2} \quad (2.1)$$

R_s and R_e in the equation is the distance to the satellite from the earth's core and earth's radius. These are static values $R_s = 42164$ km and $R_e = 6367$ km. λ is the difference in longitude between the points on earth's surface. That is, the nadir point of the satellite and the location of the user. ϕ is simply the latitude point of the user. In python the equation is implemented as following.

```
def dist_tosat(satpos_long,p):
    lat, lon = p
    Rs = 42164
    Re = 6367
    satpos = (0,satpos_long)
    dist_to_sat = np.sqrt((Rs-(Re*m.cos(m.radians(satpos_long -
    p[1]))*m.cos(m.radians(p[0]))))**2 +
    (Re*m.sin(m.radians(satpos_long -
    p[1]))*m.cos(m.radians(p[0])))**2 +
    (Re*m.sin(m.radians(p[0])))**2)
    return dist_to_sat
```

It is also relevant to calculate the elevation from a point on earth to a satellite [5]. The parameters needed for the elevation is similar to equation 2.1.

$$e = \arctan \left(\frac{\cos \lambda \cos \phi - 0.155}{\sqrt{1 - \cos \lambda^2 \cos \phi^2}} \right) \quad (2.2)$$

The azimuth between two points can be calculated using the "arctan2" function which is the 2-argument arctangent [6]. As in the previous equations 2.2 and 2.1,

λ is the difference in longitudes between the satellites nadir point and the user longitude. ϕ_1 and ϕ_2 is the user latitude and the satellite nadir latitude respectively. For an example, an azimuth of 90° means that the satellite point is east of the user point.

$$A = \text{atan2} \left(\sin \lambda \cos \phi_2, \cos \phi_1 \sin \phi_2 - \sin \phi_1 \cos \phi_2 \cos \lambda \right) \quad (2.3)$$

The equation will return a value that ranges from 0° to 360° . Or if desired, the azimuth value can be shifted to a value range of -180 to 180° .

2.4 Transmission equations

2.4.1 Carrier to noise ratio

The carrier to noise ratio, $\frac{C}{N}$ or CNR, is a measure of the received carrier strength relative to the strength of the received noise. A higher CNR ratio represents a better performance and higher reliability in communication applications. It is similar to the Friis equation with a added value for the noise level as can be seen in equation 2.4. The k_b is the Boltzmann constant, T is the system temperature and B is the bandwidth.

$$\frac{C}{N} = \frac{P_t \cdot G_t \cdot G_r}{k_B T B} \cdot \left(\frac{\lambda}{4\pi \cdot R} \right)^2 \quad (2.4)$$

2.5 Antenna Characteristics

2.5.1 Beamwidth and footprint

The satellite coverage is determined by the antenna directivity and gain function which is determined by the material and dimensional characteristics of the antenna. The beamwidth is the angle in degrees between two points where the directive gain has the same value. In satellite communications for reception the peak power is observed at the boresight of an antennas gain pattern. The signal strength reduces as the receiver moves away from the center point. If a satellite is placed above a flat surface the antenna beam shape is circular with a distinct center point with the peak signal power. If the satellite is instead placed in Geostationary orbit around Earth transmitting a beam to the surface, the beam in question depending on the inclined angle will take a elliptical shape. To determine peak power, center position and the 3dB beamwidth angles for the incident beams now grows in difficulty. To be able to predict signal power, beam center and 3 dB beamwidth angles there is a need for modelling and analysis.

2.5.2 Polarization

The polarization of an electromagnetic wave is the orientation of which the electric-field component (\vec{E}) oscillates in. An electromagnetic wave is electromagnetic radi-

ation propagating in a given direction. It consists of an electric-field and magnetic-field.

$$\vec{P} = \vec{E} \times \vec{H} \quad (2.5)$$

Where \vec{E} represents the electric field, \vec{H} the magnetic field and \vec{P} the poynting vector, which is the directional power flow of the wave. The E-field and H-field are perpendicular in relation to each other and perpendicular to the propagation direction. The propagation direction can be assumed to be in the z-direction of a given coordinate system, the E-field can then be described by the equation:

$$\vec{E}(t, z) = E_0 e^{(j\omega t - kz)} \quad (2.6)$$

The H-field is described similarly:

$$\vec{H} = H_0 e^{(j\omega t - kz)} \quad (2.7)$$

Polarizing an electromagnetic wave is a method of separating waves. Separating waves is important to avoid interference and to allow for communication to work. Separation is mainly done in the frequency domain and by the use of communication protocols. Equation 2.6 can be further elaborated to include the direction of oscillation as following:

$$\vec{E} = \begin{bmatrix} e_x \\ e_y \\ 0 \end{bmatrix} e^{(j\omega t - kz)} \quad (2.8)$$

There are three types of polarization, linear, circular and elliptical. In this project only linear polarization is relevant. Linear polarization can be horizontal, vertical or a combination of both, slant polarization.

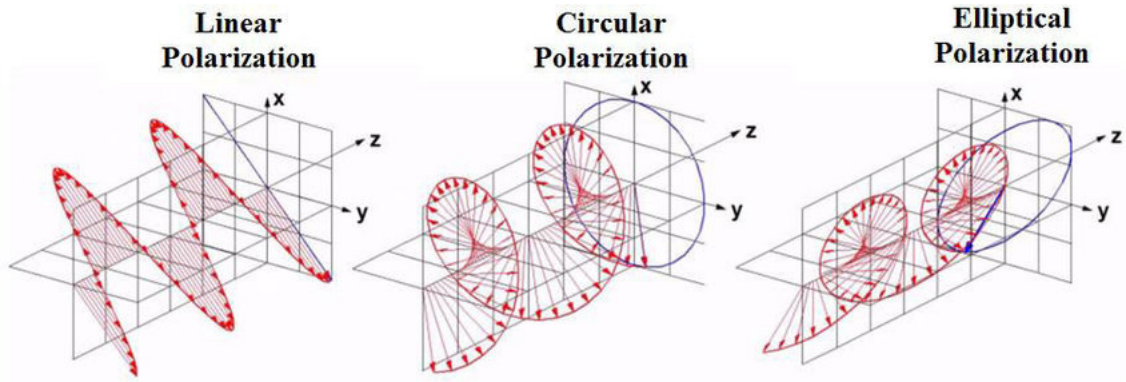


Figure 2.1: Three types of polarization. [4]

Figure 2.1 depicts the three types of polarizations of an electromagnetic wave. Linear polarization, unlike circular and elliptical, oscillates in the same plane. Horizontal or vertical polarization means linear oscillation along either the x-axis or the y-axis.

2.5.3 HTS beam

Beams describe the shape of the electromagnetic waves radiated from an antenna. From satellites beams are projected on Earth and spread according to their distance from Earth. On a conventional FSS-satellite the beams are designed to cover large areas. GEO-stationary satellites are typically FSS-satellites and illuminate very large areas of the visible side of Earth. The typical theoretical range of FSS-satellites is $\pm 70^\circ$ latitude and $\pm 30^\circ$ longitude. The coverage is large due to the distance from Earth to the GEO-stationary satellites, which is roughly 36 000 km from Earth's surface. The GEO-stationary orbit is one of the most common orbits for satellite as it is the orbit where the satellites orbit around Earth's axis with the same angular velocity as Earth. This allows the satellites to always look at the same point on Earth's surface.

In this project HTS-satellites are the only relevant satellites. The HTS-satellites are GEO-stationary satellites that vastly differ from conventional FSS-satellites. HTS-satellites have an overall coverage similar to conventional satellites but using multiple beams instead of a singular beam. Multiple spot-beam technology allows the satellite to focus more power in a smaller area and allowing for a stronger signal in exchange for geographical coverage. This means that the throughput in the communication can become significantly higher.



Figure 2.2: Representation of the difference between regular FSS satellite and HTS satellite coverage. Picture taken from Telenorsat. [8]

2.5.4 Beam Center

The beamcenter is a theoretical point representing the position inside a beam with the highest EIRP, and in result the highest received CNR of a terminal. The figure 2.3 shows the 3-dB EIRP coverage of a IS33 beam.

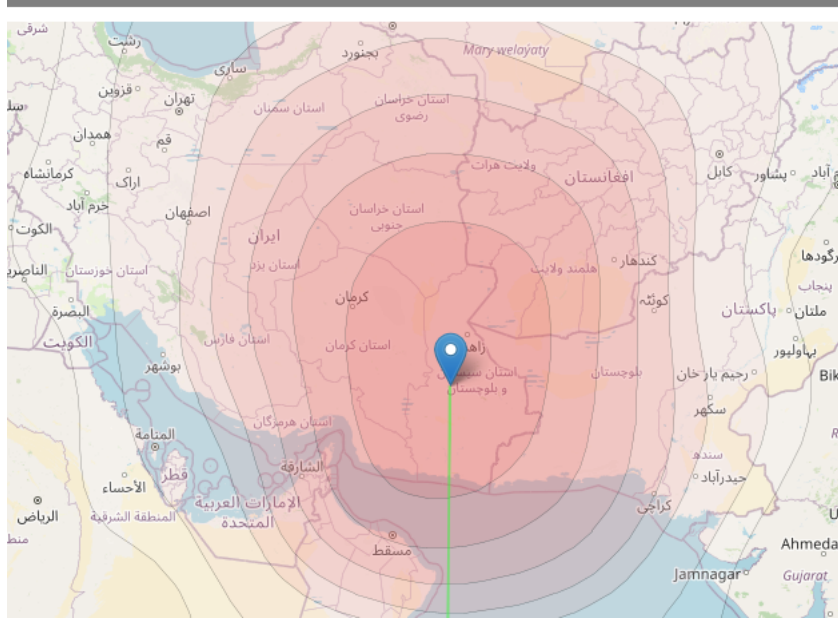


Figure 2.3: Beam contours of beam U36 from IS33 according to SatBeams. The center contour with the blue marker has a estimated EIRP of 55.6 dBW. For each expanded contour outwards the EIRP is reduced by 1 dBW.

2.6 Satcube Ku

The Satcube Ku is a manually pointed satellite terminal designed to provide instant internet access all over the globe. The directional antenna on the terminal is pointed by the user towards a given satellite. The technical specifications can be found in figure 2.7.



Figure 2.4: Satcube Ku terminal.

2.6.1 Operation and usage

The Satcube Ku is flexible and lightweight which is designed for field use. It weighs only 8kg according to the technical specifications in figure 2.7. Batteries are primarily used to run the terminal. There are three slots available for batteries where hot-swapping batteries is possible. It is also possible to run the terminal directly through a power-connection which can be useful when, as an example, is utilized in camps where there is power available.

Upon starting the terminal the user is given a list of beams to choose from. When a beam is chosen the user has to manually aim the antenna towards the satellite which the beam originates from. There are three axes of rotation which has to be aimed, azimuth, elevation and polarization. Built-in sensors guide the user to aim the antenna in the right direction. Elevation and polarization has a maximum

pointing error of $\pm 0.5^\circ$ programmed into the firmware, this is to ensure minimal loss in carrier-to-noise ratio due to pointing error. The display gives information of the carrier-to-noise ratio when the beam is found and will also ask the user if an internet connection wants to be established.



Figure 21 Terminal positioning angles. From left to right: Azimuth, Elevation and Polarization.

Figure 2.5: Pointing instruction for the Satcube Ku



Figure 2.6: Terminal user interface

Item	Specification
Dimensions	470 / 300 / 55 mm (18.50 / 11.81 / 2.17 inches)
Total weight	8 kg (17.6 lbs)
Data rate	Minimum 4 Mbps full duplex, up to 20 Mbps via HTS satellite
Batteries	2-3 hours transmit time (additional hot-swappable batteries pack are optional).
Frequencies	Tx 13.75 – 14.50 GHz Rx 10.95 – 12.75 GHz
Polarization	Linear Horizontal / Vertical V/H or H/V
EIRP	Typical 46 dBW
G/T	9 dB/K
Satellite modem	iDirect Evolution or UHP Networks
Power consumption	130 W (MAX)
Secondary links	WLAN
Interfaces	2 x LAN, 1 x USB host, WLAN Access Point (IEEE 802.11 b/g)
User interface	Satcube GUI, Display, Pointing system
Water and dust protection	IP65
DC input	24 VDC
AC/DC input range	100 – 240 VAC, 47 – 63 Hz, 150 W
Ambient temperature tolerance	-20°C (4°F) to 50°C (122°F)

Figure 2.7: Satcube Ku technical specification taken from the Satcube Ku handbook.

2.6.2 Initial Limitations

In the user manual environmental limitations are listed to alert the user to avoid certain extreme weather conditions when operating the terminal. The limitations are listed in figure 2.8. The terminal has global coverage except for the poles of the planet.

- Wind strength**
 The Satcube Ku terminal should not be operated at wind speeds greater than 18 m/s or 40mph. The highest risk for the terminal to flip over is when the wind comes straight from behind the transmission direction and the antenna is at an upright position.
- Temperature limitations**
 The Satcube Ku terminal has an operating temperature range of -20°C to 50°C.
 To avoid risk of high temperatures, please avoid placing the terminal in direct sunlight exposure.
- Presence of reactive substances**
 Do not operate the equipment in the presence of flammable gases, liquids, fumes or equivalent.
- Water and dust**
 The Satcube Ku terminal is designed to endure rough conditions and is IP65 rated (Ingress protection). See Figure 1 for a full explanation of different IP ratings and its implications to condition limitations.



Figure 2.8: Satcube Ku environmental limitations.

2.6.3 OTA-files

OTA is an abbreviation for "Over-the-air" and are files used for wirelessly update of the systems with critical information. In the modem there is a OTA-file that is updated once a month from the satellite provider which includes information regarding satellite position, beams emitting from the satellites, frequency of beams, symbolrate, beam contours and much more. It is from this file the terminal calculates available beams in and around the user location and the necessary information to correctly orient to target a satellite position.

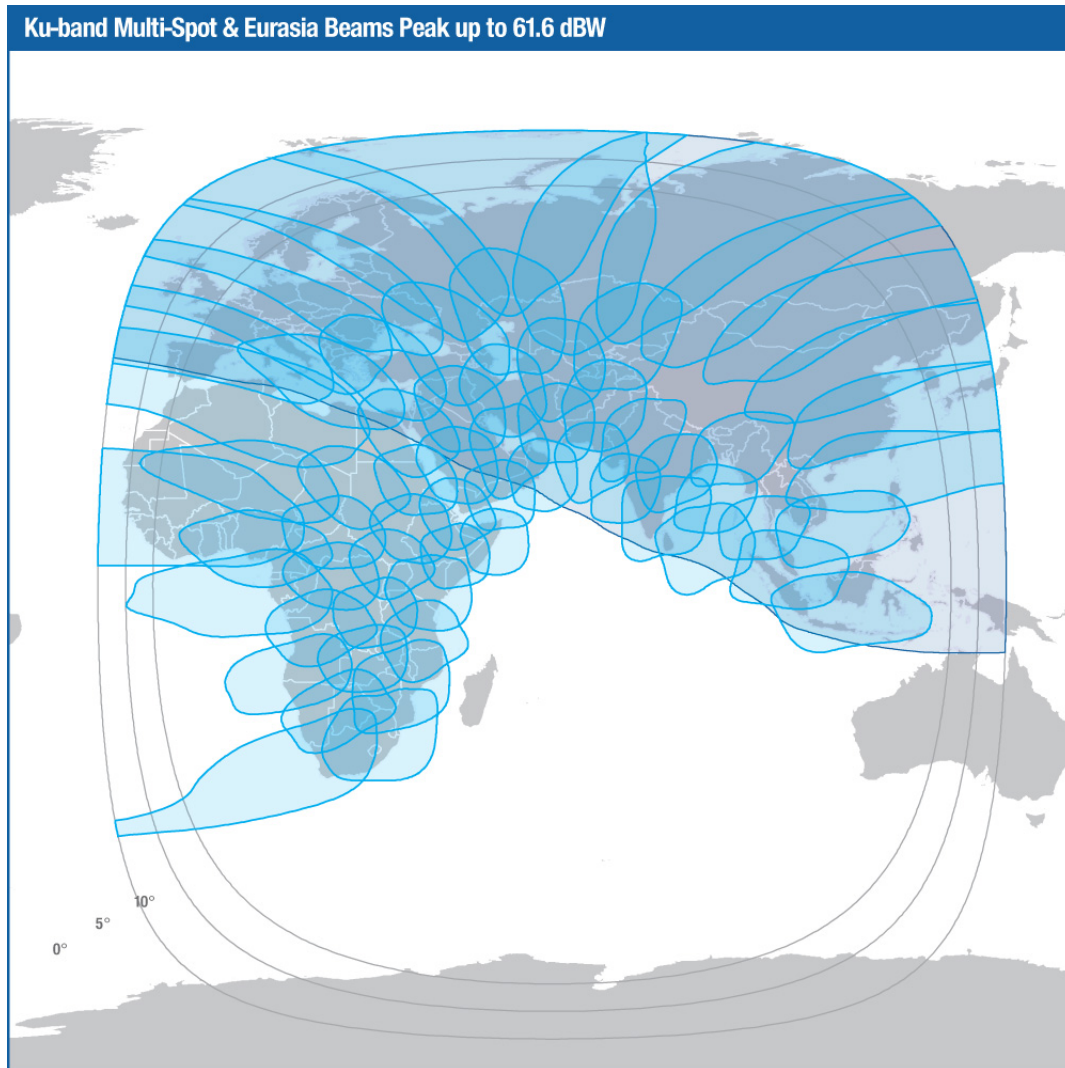


Figure 2.9: HTS Beam map of the IS33e satellite, taken from Intelsat's website [1].

Figure 2.9 depicts the beam map from Intelsat's satellite IS33. Spot beam technology is clearly visible on this map. The small blue circles correspond to HTS-beams while the larger blue corresponds to the Eurasia beam which is of less priority.

2.6.4 Polygon center

The contours can be formed into a polygon with a corresponding polygon center. This center is not related to the antenna beam center due to the fact that it is independent of incident angel from the satellite. For near zero incident angles between the satellite and polygon center the distance from polygon center to beam center is very small. As the angle increases the distance between the two centers increases. There is a excel file where some of the beam centers have been measured by Intel-Sat and this file will be used to derive a function to estimate the beam center of unknown beams with only the beam contours and satellite position.

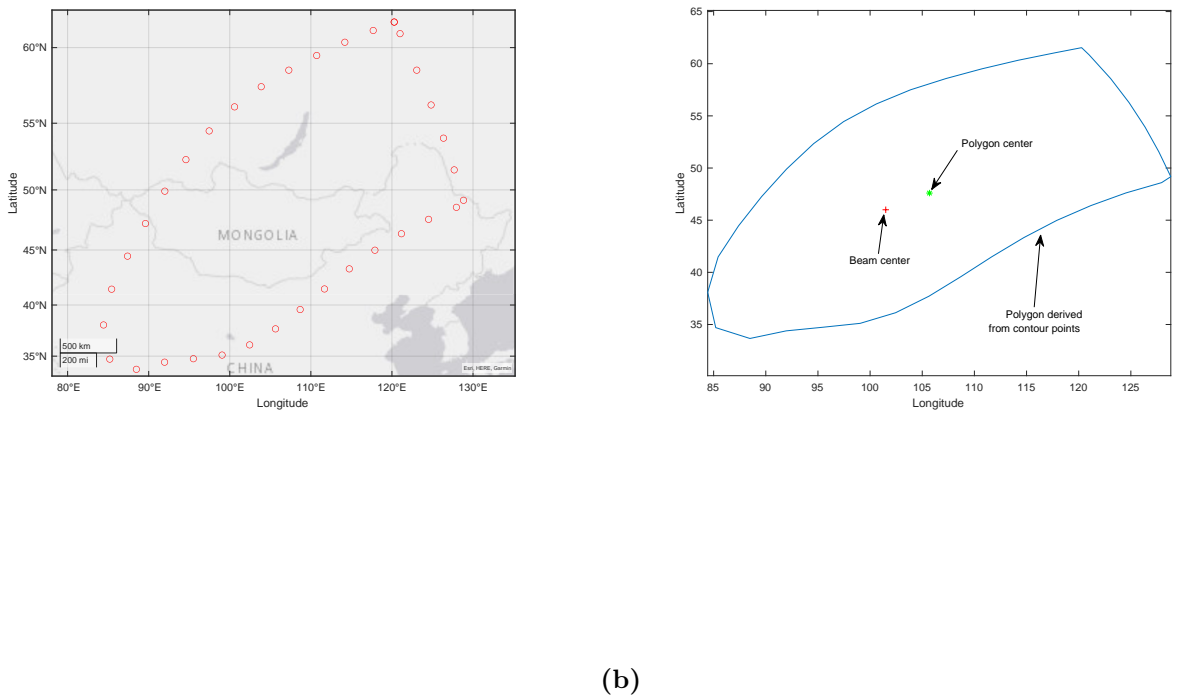


Figure 2.10: a: An example of the contours of a HTS beam from the IntelSat IS33, Beam ID 22 spanning over Mongolia. b: The polygon equivalent of the contour points with the polygon center and beam center included.

2.6.5 Weather attenuation

Water molecules and rain can cause attenuation of RF-signals due to absorption and scattering.

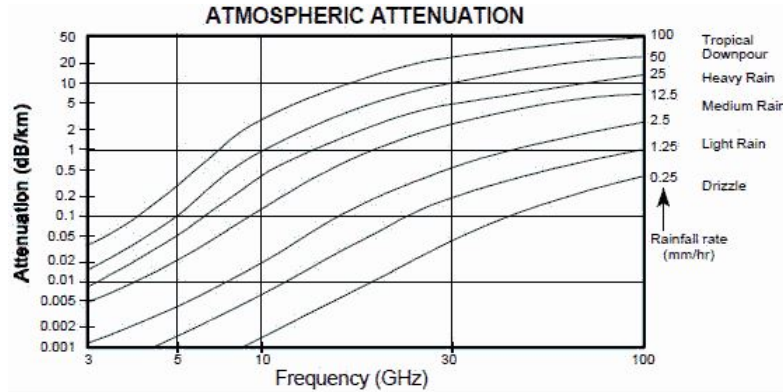


Figure 2.11: Attenuation due to rain for different frequencies according to Everything RF website.[7]

As can be seen in figure 2.11 depending on the rainfall rate the in the 10 to 15 GHz region the loss can be up to 5 dB loss per kilometer. In satellite communication these losses must be calculated with the slant range to the satellite and scale height of the atmosphere.

2.6.6 Sweden's meteorological and hydrological institute

SMHI is a Swedish government research agency that works with weather, water and climate. The agency has a fleet of weather data stations distributed over the entire country that collects data in an interval ranging from days to minutes. The weather data information can be downloaded from the SMHI website in CSV format and be used in combination with historic user data to eliminate or compensate for the varying weather variations.

3

Methods

The methods used in this project are divided in two subcategories, Beam algorithm and data analysis. The Beam algorithm section will involve how it works and the way it operates and the data analysis part involves the assumptions made for each analysis.

3.1 Planning and preliminary investigation

The first step of this project is to devise a plan of approach and how to schedule the time and planning during the project time span. There is a multitude of parameters involved in satellite communications and to be able to know what to focus on there is a need to find the limitations of the terminal and data available.

The preliminary investigation consists of several parts. First and foremost an introduction to the Satcube Ku terminal and the beam algorithm that is implemented in the terminal. The user interface on the terminal is very well designed and it is key to understand how the user interacts with the terminal in order to evaluate a new algorithm and analyse user data.

Documentation for the current algorithm was provided through Gitlab. The documentation contains information about functionality, limitations, source code and other relevant information. This was seen as a starting point and introduction to the current algorithm which will be used as a benchmark for the new algorithm.

The goal of this project is to investigate the influence of certain parameters, to suggest a new and more accurate beam algorithm for the Satcube KU terminal and to suggest improvements where it can be found. All of the goals are interlinked and part of the project but the beam algorithm design and testing is a priority.

3.2 Beam center

Working geographically with the beam was the first issue raised in this project. It is an important and difficult problem to solve. The beams geographical information is essentially the only known parameter in field.

3.2.1 Starting the terminal

For the terminal to be able to establish a uplink it requires to be manually pointed to a specific satellite position. The position of these GEO-stationary satellites and the beam area coverage is all dependent of the position of the terminal which is determined by GNSS positioning satellites when the terminal starts. Onboard the terminal there are OTA-files that provide the terminal with frequency, symbol rate and a list of contour points which are unique for each HTS-beam from the satellite as can be seen in figure 2.10a. With this information the user must choose a beam that is available and then manually point the terminal to the satellite to achieve duplex communications link.

3.2.2 Beam shape and analysis

The contour points are of a non-homogenous nature. The beams are projected as circles on earth and the further away from the satellite they travel the more non-circular they become. The beam center is not located at the geometric center of the beam, as the beam is stretched out across the surface of the earth the beam center is located at a point on Earth much closer to the satellite. It is important to differentiate between a beam center which is the maximum point of power received inside the beam and a polygon center which is the geometric center of the beam contour shape. Figure 3.1 shows an example of a beam contour in Python.

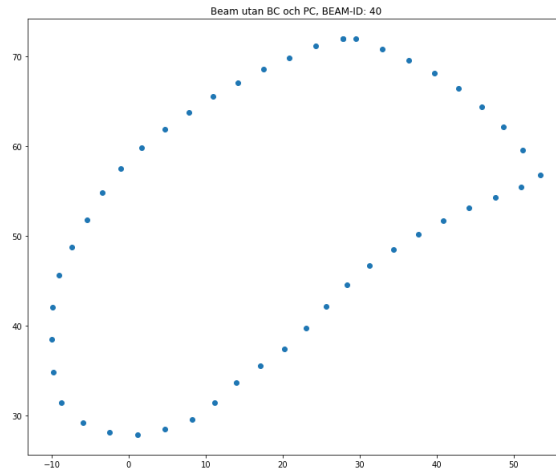


Figure 3.1: Example contour of a beam in Python. Longitude on the x-axis and latitude on the y-axis.

The beam contour provided are data points as tuple values which can be seen in figure 3.1. The tuple contains latitude and longitude values and the contour is formed by a list of multiple points. The beam contour can then be visualized in python by scatter plotting the data points.

3.2.3 Correction

The contours provided from the modem is the primary input and starting point for the algorithm. With the points a polygon can be created and a polygon center point can be calculated using the python library Shapely. The polygon center coincides with the beam center for high elevation angles as the beam highlights a circular area on the ground, as the elevation angle decreases the beam becomes more ellipsoidal and the distance between the polygon center and beam center increases.

To calculate the separation distance in all the beams depending on the satellite position an excel-sheet of data of beams and their correct beam center positions and power levels according to the satellite providers Intelsat was used. In order to correct the polygon center to a more accurate beam center approximation two things were needed, amount of correction needed, that is, distance in kilometers and the direction of the correction, which was assumed to be towards the satellite, or more specifically, the nadir point of the satellite.

To estimate the distance needed to correct the polygon center the aforementioned excel-sheet was used. The excel-sheet contained information of three satellites: IS-33e, IS-37-e and H-3e. Only the information from the satellite IS-33e was used as it covered areas ranging from northern Europe to Australia and in total roughly 55 beams were used. For every beam a polygon was created using its contour and a polygon center was calculated. A distance was calculated from the polygon center to the alleged correct beam center from the excel sheet, this distance was seen as the "error distance". The distance from the polygon center to the satellite was calculated using equation 2.1. After that a scatter plot was created with the error distance on the y-axis and the distance to the satellite on the x-axis. The idea was to be able to analyze how the parameters correlate and ultimately determine the error distance as a function of the distance from the satellite.

A function that accurately predicts the distance error can be found by curve fitting aforementioned data. The curve fit will return parameters which can be mapped to a suitable function. After calculating an error distance based on the distance to the satellite the azimuth from the polygon center to the nadir point of the satellite had to be determined. The azimuth was determined using equation 2.3.

Figure 3.2 depicts the model for the curve fitting. The x-axis contains the distance to the satellite of each polygon center from the beams, the y-axis contains the measured alleged beam center to the same beam. In the figure it is clear that the relation between the parameters are of parabolic nature. To create a function of parabolic nature to fit the data points the package SciPy was used. SciPy is a package used for scientific calculations and analysis. With help of the function `curve_fit(function,x,y)` the data is easily inserted along with the wanted function for optimization, the function in this case is a parabolic function. The wanted parabolic function is defined in python as:

```
def objectivesq(x,a,b,c):
    return a*x +b*x**2 + c
```

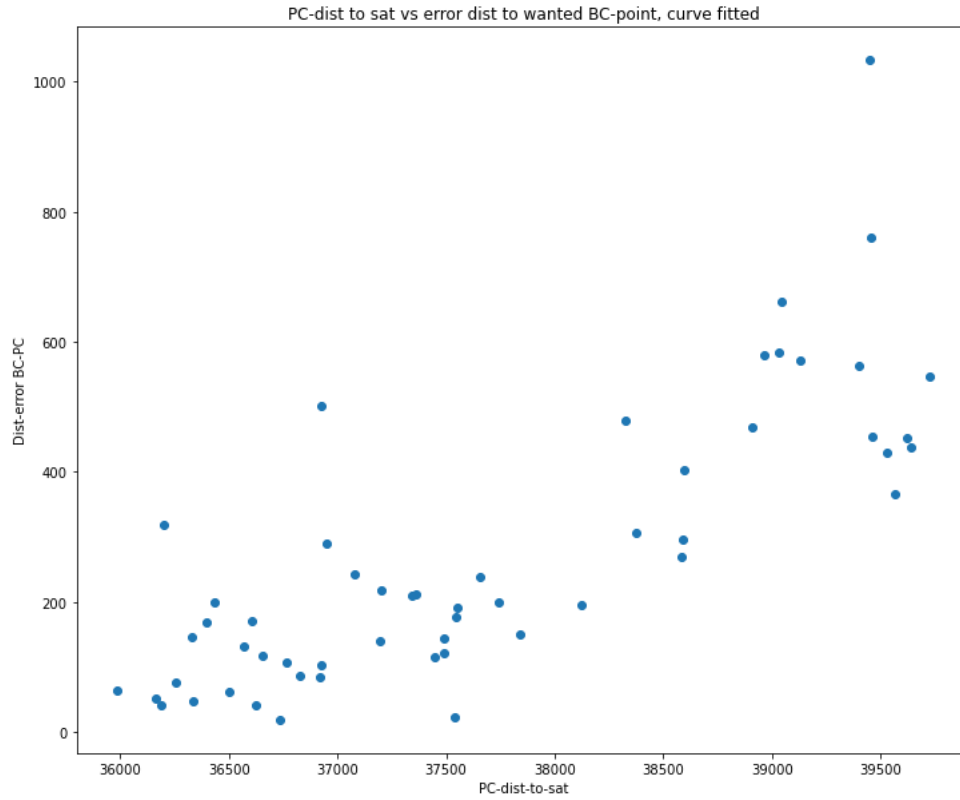


Figure 3.2: Distance to satellite vs distance error to correct beam center.

The curve fit function is then called by this line of code:

```
popt, _ = curve_fit(objectivesq, dist_pc_data, dist_er_bcpc)
```

Popt stores the variables a , b and c . These variables are then fed into the aforementioned $\text{objectivesq}(x, a, b, c)$ function along with the distance to the satellite. The function will return the desired distance error.

To calculate the azimuth equation 2.3 was implemented into a function in python accordingly:

```
def azimuth_calc(point_sat, point_pc):
    point_sat_lat, point_sat_long = point_sat
    point_pc_lat, point_pc_long = point_pc
    azi = np.degrees(m.atan2(m.radians(m.sin(m.radians(point_sat_
long-point_pc_long))*m.cos(m.radians(point_sat_lat))),
m.radians(m.cos(m.radians(point_pc_lat))*m.sin(m.radians(point_sat_lat))-
m.sin(m.radians(point_pc_lat))
*m.cos(m.radians(point_sat_lat))*m.cos(m.radians(point_sat_long-
point_pc_long))))))
    return azi
```

The azimuth and distance error along with the polygon center point was ultimately fed into a function which utilizes the Geoid package. Geoid has a complete function that takes a point, longitude, latitude, a distance and an azimuth as input parameters. The output of the function is latitude and longitude coordinates and back azimuth of a new point that has been moved the specified distance in the azimuth direction. In python the function is applied as following:

```
def correction(p,direction,distance):
    geoid = Geod(ellps="WGS84")
    lat,lon = p
    distance = distance*1000
    lng_new, lat_new, azi_back =
    geoid.fwd(lon,lat,direction, distance)
    return lat_new, lng_new, azi_back
```

3.2.4 Carrier-to-noise estimation at the beam center

One of the key limitations to the beam algorithm was that the modem inside the terminal has no information about the signal strength of the beams. The beam-center function is only designed to estimate the strongest point of a beam. The beams are not equally distributed in power. This raises an issue as one beam can be geographically close to the user but weaker in strength. The decision to choose the closer beam will not be optimal in that case. To estimate a carrier-to-noise ratio of a beam is difficult. In this section the methods taken to estimate the carrier-to-noise ratio will be presented.

3.3 Data analysis and testing

For this project we had access to terminal user data with entry CNR values, Intelsat files regarding Beam center of certain satellites and relative CNR value for each beam and also OTA-files from the modem which provided beam characteristic data such as frequency and contours of each beam. Furthermore SMHI data was also included for compensation and analysis. How the data was used will be explained in this section.

3.3.1 Available data

An excel-sheet with approximately 11600 rows and 29 columns of stored user data was provided by Satcube. The 29 columns in the document consist of a vast amount data points that covers user experience, technical specifications, location, etc. Data from Intelsat was also provided which contained information regarding the beams transmitted from HTS-satellites. The data was packaged as JSON-format. The Intelsat data is a key component to the development of the beam selection algorithm as it contains the input to the algorithm.

3.3.2 Analysis method and cleanup

The user data file that was provided by Satcube contains all entries made by all terminals from October of 2019 to September of 2021. The user data file is a excel-sheet with approximately 11600 rows and 29 columns of stored user data. The file contained among other things the collected data regarding software edition, terminal series number, terminal location in latitude and longitude, GNSS satellites above, time to network, time to link up from starting, carrier to noise level received, longitude of target satellite, frequency, bandwidth, polarization, target azimuth, target elevation and the actual orientation of the terminal antenna in azimuth and elevation. For every successful link these values are stored once.

To be able to do analysis of what could have an impact on the performance of the terminal and to what extent there is the need to filter the data so that certain influencing factors that are not important be reduced or removed, such as weather at time the terminal was active. In the user data file the entries are recorded all over the world and scattered over the span of a day. To be able to remove this fluctuating unknown parameter there is the need to either compensate or remove data points when it is either raining or is cloudy.

Globally there were difficulties of finding a source for the cloud and rain content at a certain time frame, but in Sweden there are SMHI weather stations that are available to public use and also contain a high resolution of cloud coverage and rain data for each hour of the day.

Considering this limitation the analysis was divided in two parts, one which made use of the SMHI input and could be used to analyse the performance of terminals individually or as a group over a long period of time, and one which focuses on clusters of recorded data points which take place in a short period of time at the same location so that the change in the environment around the terminal remains constant.

Furthermore there was also the need to remove certain entries in the user data file which involved test rigs with unreasonable high C/N or entries without location data. An undefined location may occur when a user can not locate enough GNSS-satellites or when the location is not stored in the user data for privacy reasons.

3.3.3 Setting up analysis

The goal of the analysis is to find a way to look at each beam individually and the performance experienced by the user inside this beam. The excel-sheet did not include specific information of which beam the user was connected to when using the terminal. Using the IF frequency and user location from Satcubes user data Excel-sheet and beam frequencies and beam id from Intelsats Json file it was possible to link which beam the user was connected to. The IF frequency had to be matched to the RF-frequency from Intelsats data. The terminal has three different LO-frequencies depending on the RF-frequency and in the downconversion the IF-frequency is calculated by subtracting the RF-frequency from this LO-frequency. It

is important to connect the user data to specific beams in order to rate performance within a given beam and area.

Combining the two functions creates the possibility to isolate the area and the beam which is necessary for analysing. A great portion of the user data file consists of entries connected to beams belonging to satellites not existent in the OTA-file. These user data entries can not be used to analyse beam characteristics but can be beneficial for studying the terminal instead.

3.3.4 Analysis with SMHI

The Vinga SMHI station is located west of center of Gothenburg and can was used in combination with the data recorded in the region. The offices of Satcube where a majority of testing and recording is relatively close to this station and was used to filter out cloudy and rainy days.

3.3.5 Parameter targets

The user data includes a great deal of measurements taken from Satcubes office. Doing measurements from the office allows for full aspect of the surroundings such as the weather, obstacles in line of sight and full control of the terminal. Aiming the terminal is a crucial part in achieving the best carrier-to-noise ratio possible. The terminal does not allow the user to diverge more than $\pm 0.5^\circ$ degrees from the target in elevation and polarization axis, this does not apply to the azimuth target angle and a slight pointing error may give a significant loss in signal strength which needs to be investigated.

3.3.5.1 Terminal difference

It is important to account for the difference in terminals when analyzing the data. Every terminal is not built perfectly same and deviations in construction can increase the uncertainty of user data. To try and observe the error contribution of the differences between the terminals certain conditions have to set. Such as:

- All entries made to the file within 57 to 58 degrees latitude and 11.9 to 12.2 degrees longitude
- No rain and cloud density less than 20% according to SMHI station
- Pointing errors in the elevation and polarization axis of less than 0.2 degrees
- Measurements during the same period of time
- Same frequency

3.3.5.2 Azimuth misalignment

Continuing on what was referenced in section 3.3.5 the azimuth presents a error loss in every measurement collected in the user data file and is difficult to track. The azimuth sensor in comparison to the other targeting pointing angle sensors, as the elevation and polarization target angles, is more sensitive to interference caused by

local magnetic sources and power lines. Any moving surface beneath the terminal can also cause misalignment. To estimate the influence azimuth pointing error has on the total performance we used historic user data.

The user data analysis regarding azimuth utilizes previously recorded azimuth measurements cross referenced with the CNR value of that specific connection. For each user entry made in the terminal the azimuth target and azimuth orientation is recorded once. The azimuth target angle is not useful due to the lack of information regarding the local environment when the data was collected, but the recorded azimuth orientation angle is. By only observing what CNR value a user has gained with a certain azimuth orientation there is the ability to check the following entry made to the user data file by the same user at the same time and position, and observe the change made in azimuth and the effect on performance. With enough points a distribution caused by the azimuth can be plotted. To able to isolate the error caused by the azimuth misalignment the constrains on the user data file entries will be same as in Terminal difference case and including:

- Same terminal
- Max received CNR among several points

3.3.6 Losses and uncertainties in the recorded user data file

When using the user data to build a algorithm the accuracy of the model must be within acceptable bounds. The error sources which exists in the user data entries due to difference in frequency, terminal RF-parts, local azimuth interference and more must be included as a margin of error if possible. In this project the focus is on weather, terminal difference and azimuth as these are regarded as the most mayor souces of error.

3.4 Beam algorithm

The beam algorithm function is to take in the position of the user and give out a sorted list of beams and satellites to choose from, based on a projected CNR value. The algorithm must be able to function for beams of circular shape and beams cutoff by the side of the planet. We assume that the contours given by the OTA-file is true and that the beam is assumed to be 0 outside of the contours.

3.4.1 Step 1: Position and all available beams

The position of the user must be the input to the algorithm and a subset of all the beams in the Intelsat OTA-file must be created to only present the available beams to the user.

3.4.2 Step 2: Beamcenter of beams

For the beams derived from step 1 a beam center will be calculated for each beam based on the available data.

3.4.3 Step 3: CNR value at beamcenter

Each beam will have a specific CNR value at the derived beam center, this could be possible by comparing user data, beam characteristics from the OTA-files or other methods.

3.4.4 Step 4: CNR reduction to the user position

The position of the user, if not placed close to the beamcenter, requires a new CNR value derived from the beamcenter CNR value. The attenuation must go to zero at the contours and a nonzero value at contour edges that are cutoff by the Earth curvature. The output of this step must be a list of all the available beams position dependent CNR value.

3.4.5 Step 5: CNR arranged list based on projection and accuracy

Considering the beams are of different shape, have different beam center locations and are of different power levels there should be a difference in the value of the beams at the position of the user. In step 4 those values are derived and contain acceptable margin of error so that they can be sorted from highest to lowest and be presented to the user.

3.4.6 Comments

For any method developed to solve for the 5 steps above the final result must have a accuracy smaller than the error range caused by the steps used. If the difference between two beams is less than the error range of the model then the precision of the entire method can come in question. For a functioning model the difference between the beams must be larger than the margin of error caused by the steps in the algorithm, so that the beams can accurately and reliably be distinguished and presented to the user.

4

Results

In this chapter the results for the methods used in the previous chapter will be presented.

4.1 User data analysis

The analysis of the user data and effecting parameters to the performance of the Satcube Ku will be presented in this section. For this part SMHI data was embedded with the date and time of the recorded user data entries and for each day that recorded a cloud density less than 20% the terminal data was plotted together with the RF frequency transmitted from the satellite. For this case the VINGA SMHI weather station was chosen and only the user entries from the Almedal office was used for analysis. There were also further stricter restrictions on the polarization and elevation misalignment, only user data points that are ± 0.2 degrees from the target were used in comparison to the ± 0.5 of Satcube.

4.1.1 Terminal difference

A series of plots displaying different terminals connected to the same satellites but different beams.

4. Results

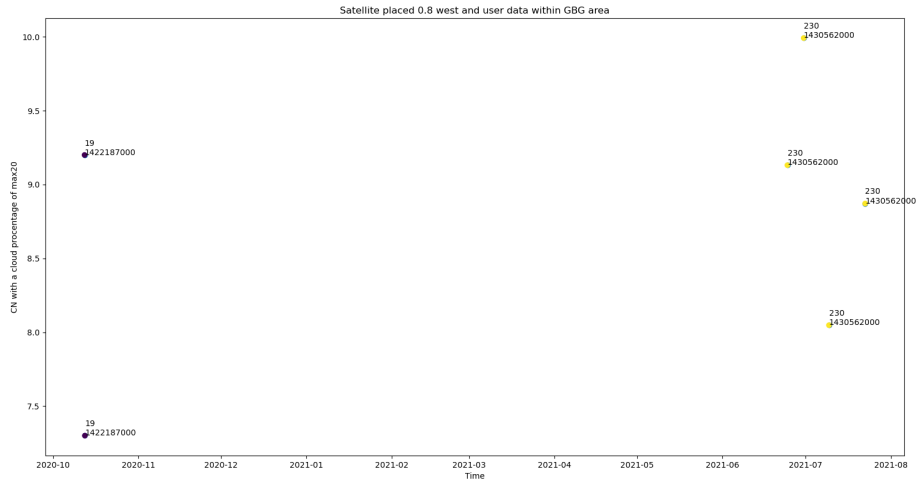


Figure 4.1: User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 0.8 West. Each point is a user connection to a satellite placed 1 west with the registered frequency below and the terminal ID above.

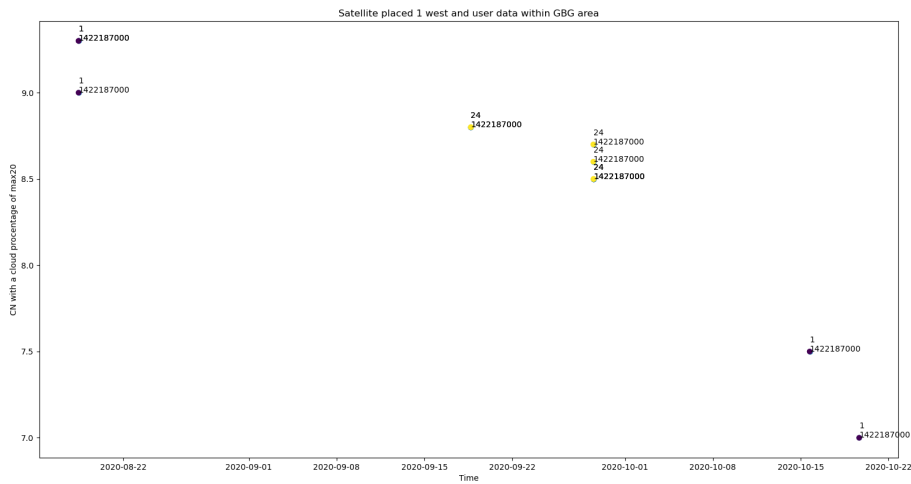


Figure 4.2: User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 1 West. Each point is a user connection to a satellite placed 1 west with the registered frequency below and the terminal ID above.

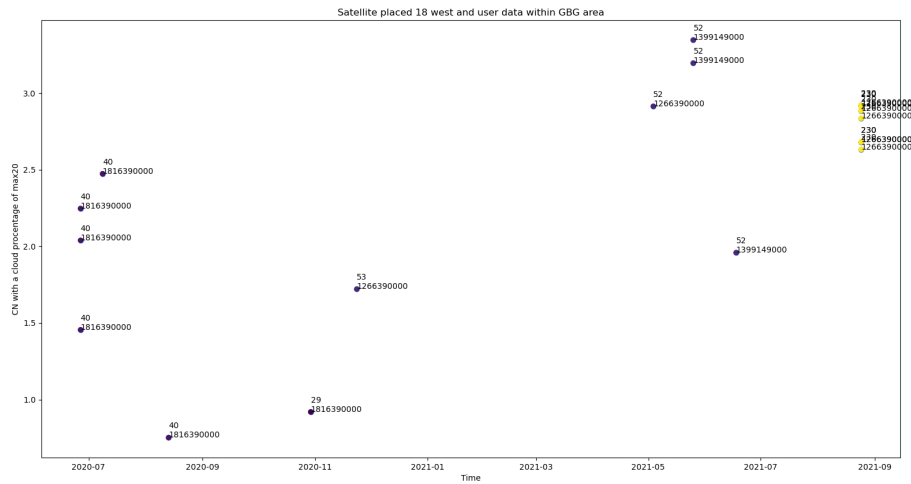


Figure 4.3: User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 18 East. Each point is a user connection to the IS37 satellite with the registered frequency below and the terminal ID above.

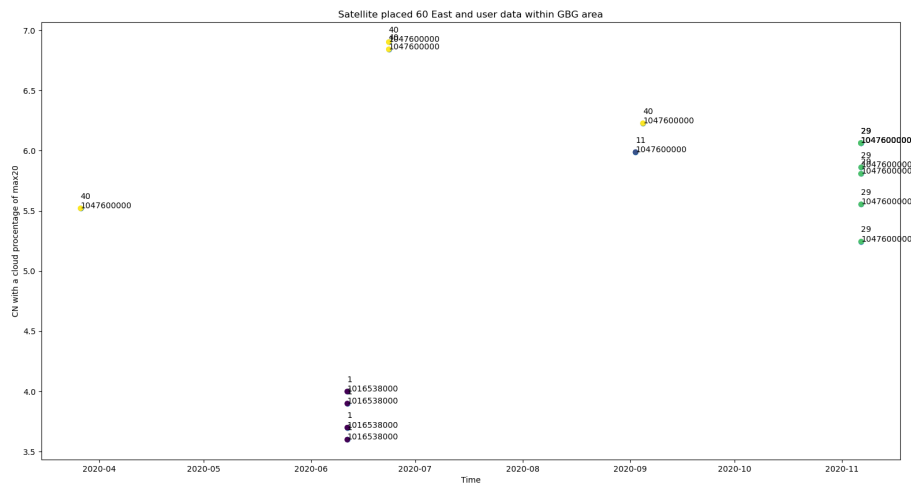


Figure 4.4: User data entries within the Gothenburg area of days with cloud density less than 20% for a satellite location of 60 East. Each point is a user connection to the IS33 satellite with the registered frequency below and the terminal ID above.

4. Results

Table 4.1: Measured data at the Almedal office in Gothenburg with several terminals.

Time of measurement	cn	Beam name	satellite name	coords	Weather conditions
2/27/23 15:00	3	2	is37e	18w	Clear
2/27/23 15:05	3.2	1	is37e	18w	Clear
2/27/23 15:31	7.1	33	is33	60e	Clear
2/27/23 15:32	3.7	29	is33	60e	Clear
2/27/23 15:37	2.2	1	is17	66e	Clear
2/28/23 11:42	3.3	1	is37e	18W	Sunny and Clear
2/28/23 11:43	3.3	2	is37e	18W	Sunny and Clear
2/28/23 11:56	7	33	IS33	60E	Sunny and Clear
2/28/23 11:57	3.9	29	is33	60e	Sunny and Clear
3/2/23 9:58	3.2	2	is37e	18W	Sunny and Clear
3/2/23 11:00	3.4	1	is37e	18W	Sunny and Clear
3/2/23 11:01	0	2	is37e	18W	Sunny and Clear
New terminal 1					
3/14/23 10:17	3	1	is37e	18w	Cloudy
3/14/23 10:18	2.3	2	is37e	18w	Cloudy
3/14/23 10:41	8.1	33	is33	60e	Cloudy
3/14/23 10:47	6.1	29	is33	60e	Cloudy
3/14/23 10:49	1.9	1	is17	66e	Cloudy
New terminal 2					
3/15/23 12:49	2.7	1	is37e	18w	Little cloudy sunny
3/15/23 12:50	2.3	2	is37e	18w	Little cloudy sunny
3/15/23 13:13	7	33	is33	60e	Little cloudy sunny
3/15/23 13:15	4.7	29	is33	60e	Little cloudy sunny
3/15/23 13:17	1.5	1	is17	66e	Little cloudy sunny
3/16/23 15:15	2.6	1	is37e	18w	Very cloudy
3/16/23 15:17	1.9	2	is37e	18w	Very cloudy
3/16/23 15:28	0.0	4	is37e	18w	Very cloudy
3/16/23 15:36	7.1	33	is33	60e	Very cloudy
3/16/23 15:37	4.8	29	is33	60e	Very cloudy
3/16/23 15:39	1.3	1	is17	66e	Very cloudy
3/17/23 13:53	0.0	1	is37e	18w	Very cloudy
3/17/23 13:54	3.1	2	is37e	18w	Very cloudy
3/17/23 14:05	6.2	33	is33	60e	Very cloudy
3/17/23 14:07	4.8	29	is33	60e	Very cloudy
3/17/23 14:09	0.4	1	is17	66e	Very cloudy
3/20/23 14:46	3.5	1	is37e	18w	Cloudy
3/20/23 14:58	3.4	2	is37e	18w	Cloudy
3/20/23 15:06	6.1	33	is33	60e	Cloudy
3/20/23 15:08	4.6	29	is33	60e	Cloudy
3/20/23 15:08	1.3	1	is17	66e	Cloudy
New terminal nr 405					
3/21/23 14:56	7.0	33	is33	60e	Cloudy
3/21/23 14:57	4.9	29	is33	60e	Cloudy
3/21/23 15:00	0.0	1	is17	66e	Cloudy
3/21/23 15:02	3.0	1	is37e	18w	Cloudy
3/21/23 15:02	3.0	2	is37e	18w	Cloudy
3/22/23 14:42	2.7	1	is37e	18w	Cloudy and rain
3/22/23 14:39	2.1	2	is37e	18w	Cloudy and rain
3/22/23 14:52	5	33	is33	60e	Cloudy and rain
3/22/23 14:54	3.6	29	is33	60e	Cloudy and rain
3/22/23 14:55	0	1	is17	66e	Cloudy and rain

4.1.2 Azimuth difference

For certain cluster of points displayed in figures 4.1, 4.2, 4.3 and 4.4 shown in the Terminal difference section taken during the same time interval the carrier to noise can be plotted with respect to the azimuth difference in the same terminals connected to the same beams. In this subsection those clusters will be shown and the decrease due to the azimuth pointing error can be investigated.

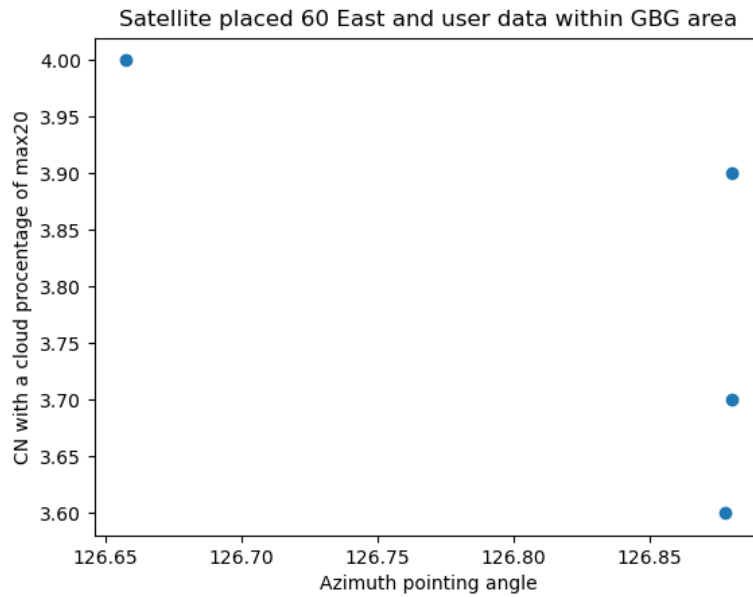


Figure 4.5: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 1 with the frequency 10,16538 GHz.

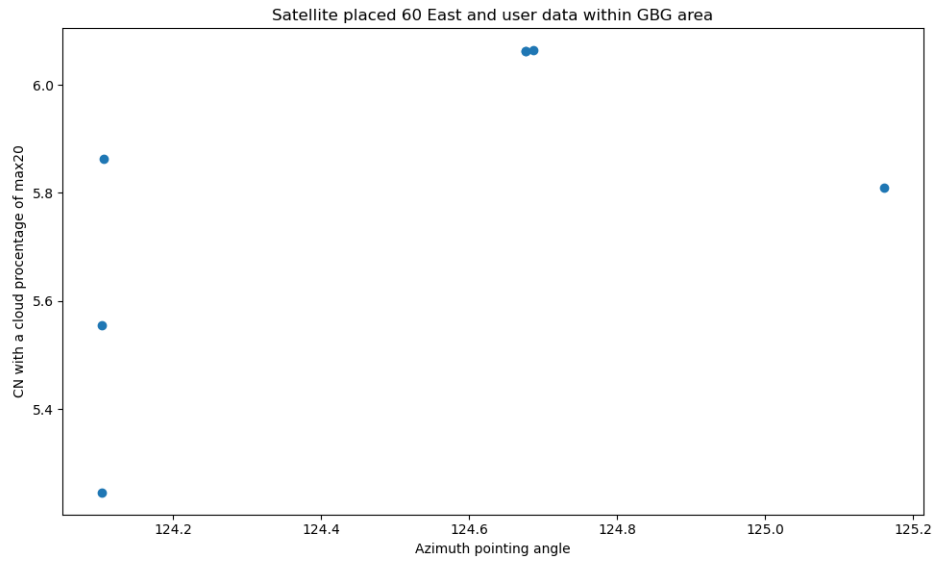


Figure 4.6: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 29 with the frequency 10,476 GHz.

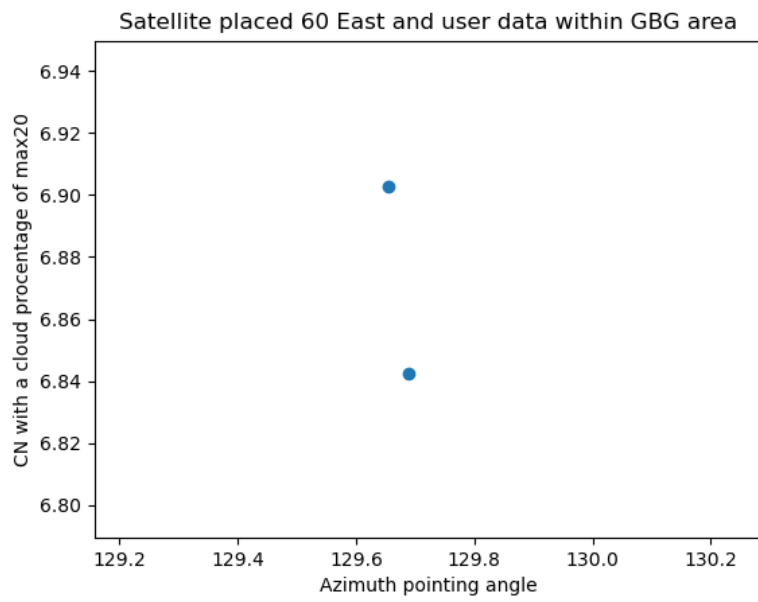


Figure 4.7: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 40 with the frequency 10,476 GHz.

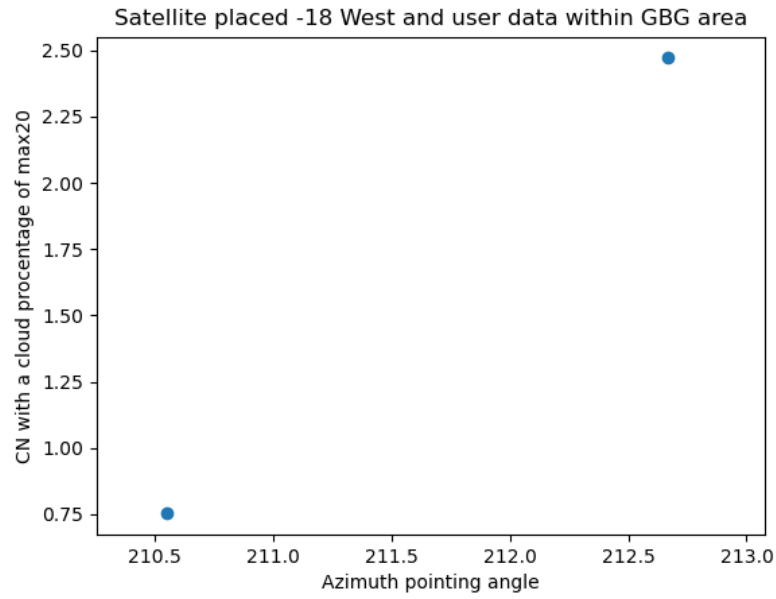


Figure 4.8: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 40 with the frequency 18,1639 GHz.

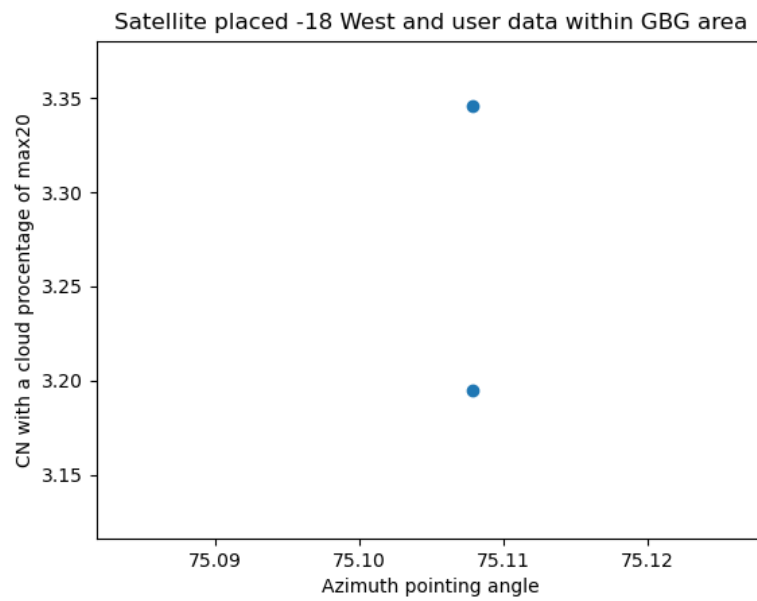


Figure 4.9: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 52 with the frequency 13,99149 GHz.

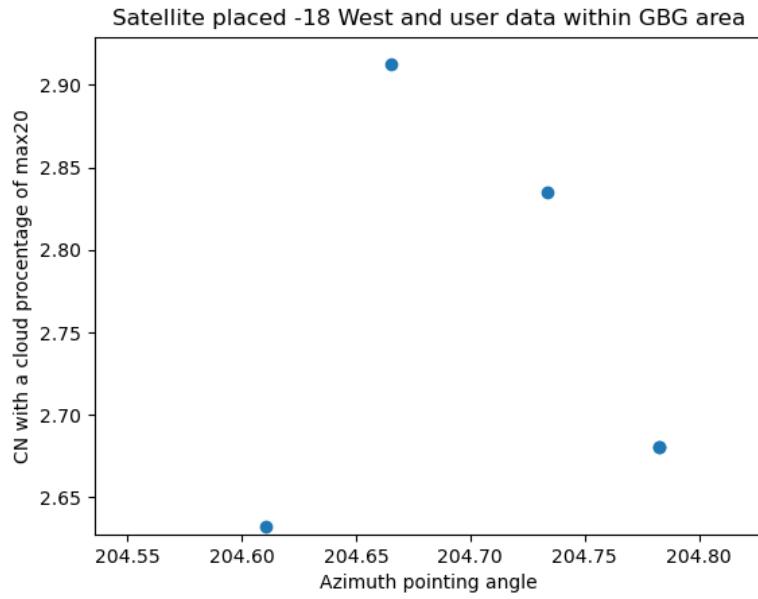


Figure 4.10: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 230.

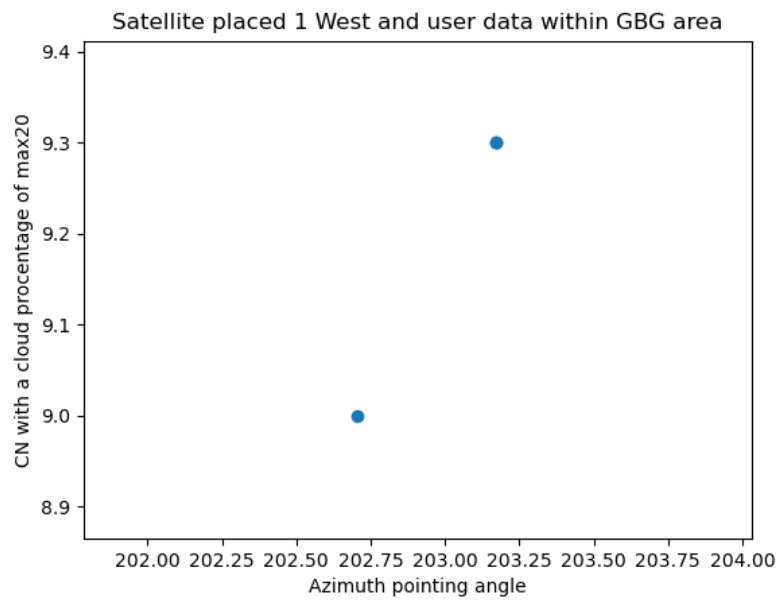


Figure 4.11: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 1.

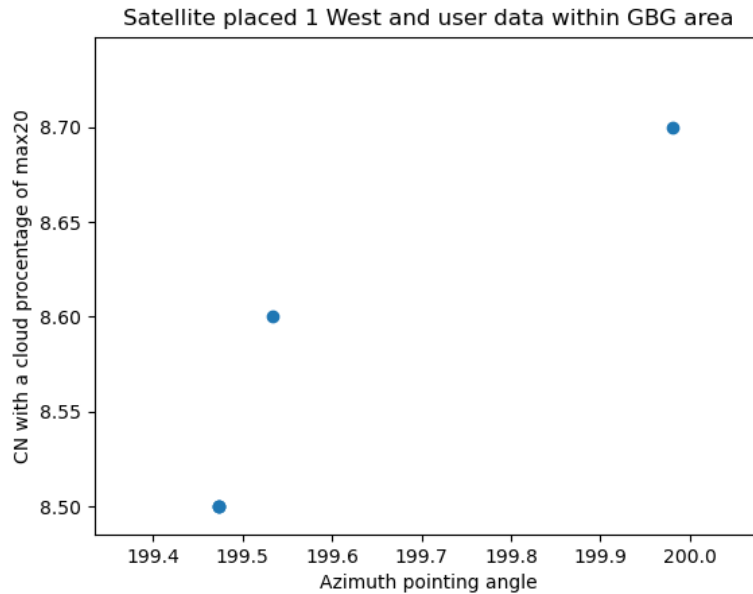


Figure 4.12: The carrier to noise ration in dB for less than 20% cloud density in Gothenburg for terminal 24.

4.2 Beam algorithm

4.2.1 Step 1: Position and all available beams

The terminal has a GNSS chip that can acquire the position of the user and has a OTA-file with all available beam contours. These can be used in combination to create polygons of the area inside the contours and check if the position as a point is within the polygon. For the contours that cross both sides of the prime meridian there is the need to shift these contour points to be able to create the polygons required. The output of this step results in a list of beams and corresponding satellites that are available to the user in that specific position. For beams outside the contour we assume that the power levels are negligible in comparison to the beams that the user is inside of according to the OTA file. In figure 2.9 the beams and their contours can be seen. Figure 5.2 is an example of the polygon that will check whether or not if the user is inside the beam.

4.2.2 Step 2: Beamcenter of beams

Finding a more accurate beam center could solve the issue of finding the strongest gradient within a beam. The goal was to create a function which has input parameters that are always known when used in field. This means that the algorithm is easily applicable in field. It is also efficient for the terminal to not require any pre-downloaded data in order for the algorithm to work. The only data required are the OTA-files provided by the satellites which contain information about available beams.

4. Results

One location that was regularly used for analysis and algorithm development was Gothenburg's location, which is roughly (57, 12). This location will be taken as an example when demonstrating the beam center algorithm in this section. Two input parameters are needed to start the algorithm, the aforementioned location and the information of all the beams from relevant satellites in the OTA-files. Firstly, the algorithm will present all available beams to the location, that is, the beams that the user location is inside. Along with the beams some additional information will be given such as which satellite the beam is from and channel information regarding the beam. The channel information is such as center frequency, symbol rate, polarization although this information is not utilized in this section of the algorithm. For visual purposes all relevant beams for the given user location are plotted with longitude on the x-axis and latitude on the y-axis.

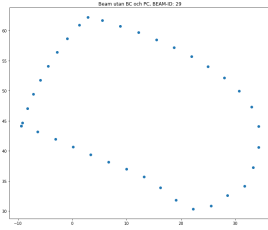


Figure 4.13:
Gothenburg beam 1.

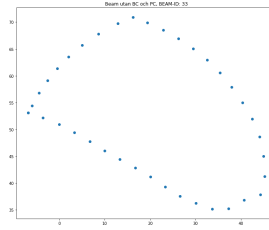


Figure 4.14:
Gothenburg beam 2.

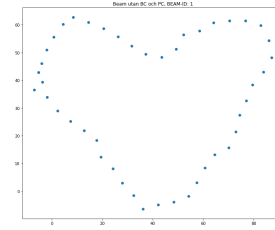


Figure 4.15:
Gothenburg beam 3.

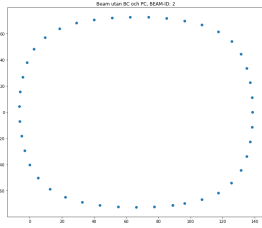


Figure 4.16:
Gothenburg beam 4.

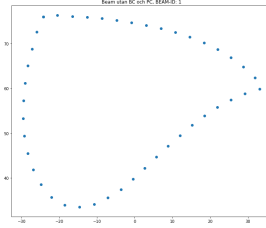


Figure 4.17:
Gothenburg beam 5.

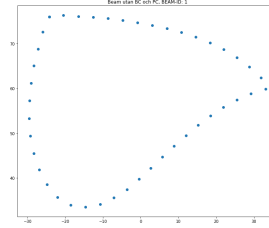


Figure 4.18:
Gothenburg beam 6.

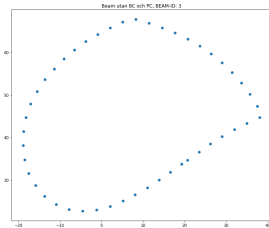


Figure 4.19:
Gothenburg beam 7.

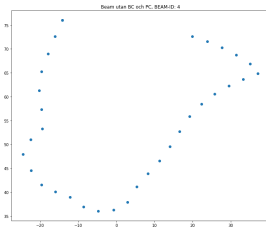


Figure 4.20:
Gothenburg beam 8.

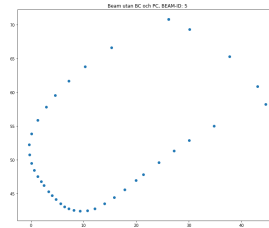


Figure 4.21:
Gothenburg beam 9.

Figures 4.13 to 4.21 show the available beams in Gothenburg based on the given information from the OTA-files. Each contour is then handled as described in the method chapter. Firstly polygons are created of each beam and each polygon center is calculated. The distance to the satellite is calculated from each polygon center and can then be passed to the parabolic function to determine the correction distance.

Parameters for the parabolic function was as mentioned found by the curve fit function from SciPy.

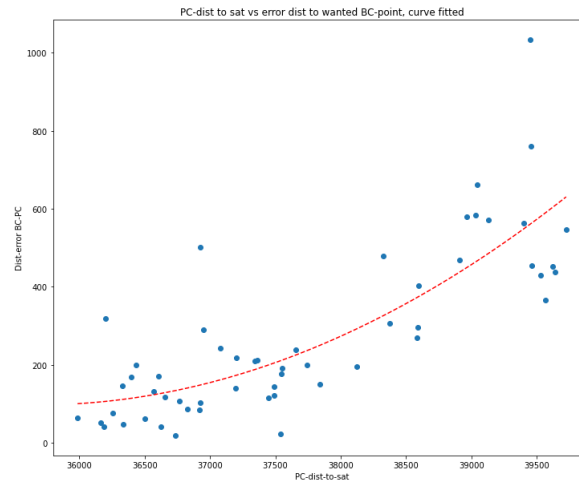


Figure 4.22: Distance to satellite vs distance error to correct beam center, with curve fit.

Figure 4.22 depicts the result from the curve fit function. According to the equation in the python code covered in the method chapter regarding beam center correction three parameters are needed to define the parabolic function, a , b and c . The resulting values for the parameters were found to be:

- $a = -2.31745$
- $b = 3.2484e-05$
- $c = 41428$

The distance error along with the direction of correction is fed to the function which estimates the beam center and a list of all corrected beam centers are created. Finally, the distance from each beam center to the user position is calculated to determine which beam center is the closest. From the nine given beams the resulting beam center approximation is as following:

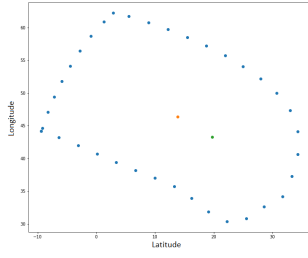


Figure 4.23:
Gothenburg beam 1.

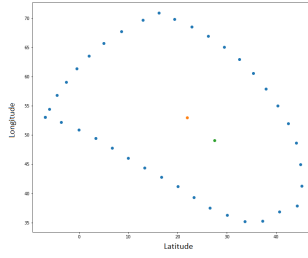


Figure 4.24:
Gothenburg beam 2.

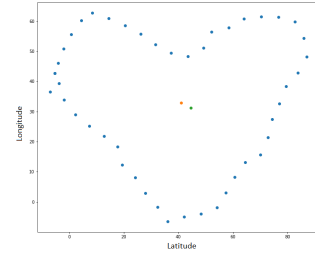


Figure 4.25:
Gothenburg beam 3.

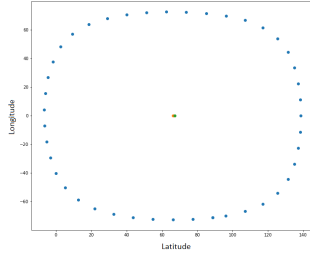


Figure 4.26:
Gothenburg beam 4.

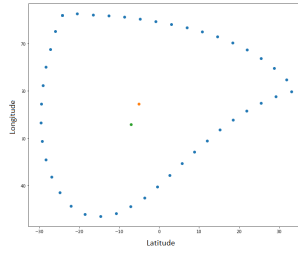


Figure 4.27:
Gothenburg beam 5.

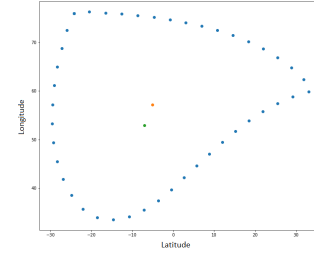


Figure 4.28:
Gothenburg beam 6.

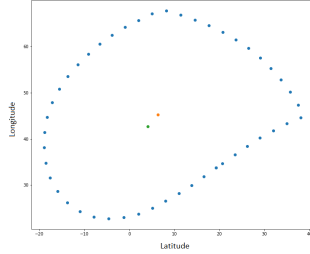


Figure 4.29:
Gothenburg beam 7.

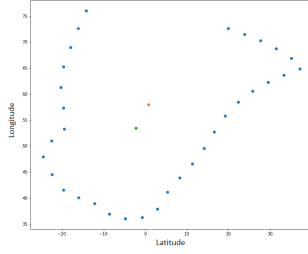


Figure 4.30:
Gothenburg beam 8.

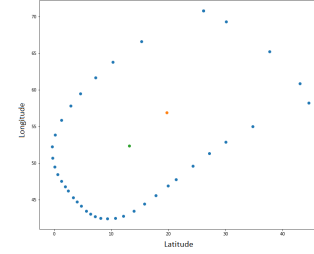


Figure 4.31:
Gothenburg beam 9.

As seen in figures 4.23 to 4.31 two points can be found inside the contour of every beam. The orange point is the polygon center and the green point is the corrected beam center. This approximation of a beam center is more accurate compared to a polygon center. For beams that have been distorted due to the curvature of earth the algorithm estimates the center more accurately. For beams such as 4.26 the beam center is roughly the same as the polygon center as the contour is mostly circular, meaning that the beam is either located close to the satellites nadir point or that the beam is a FSS-beam.

4.2.3 Step 3: CNR value at beamcenter

The beamcenter of each beam has a specific dBW power level that determines the maximum received power achievable inside the contours of that beam. These values differ from each other according to IntelSat KU spotbeam Key parameters section [1]. To be able to find the values at each beam center there is the need for a methodical method that achieves the objective without assuming a large margin of error.

4.2.3.1 Beam footprint providers

According to the Intelsat fleetmaps the multispot coverage of the IS33 satellite has a typical EIRP of 48.9 to 62.5 dBW [1] and the IS37 satellite has a typical EIRP 48.1 up to 61.1 dBW [2]. Based on the satellite coverage maps available at Satbeams the max EIRP of all beams emitted from the IS33 satellite is 55.6 dBW which is the average of the typical EIRP given by Intelsat. For the IS37 the values are either 49 dBW or 56.4 dBW. The 49 dBW is for no apparent reason and the 56.4 dBW is also a average for the Algeria region which is also included in the key parameters of the IS37 satellite [2]. The EIRP range of approximately 13 dBW provided by Intelsat and Satbeams is too high to be used as a basis to determine a specific beam center value for the beams. Testing has shown that the difference in the beams emitted from the same satellites are within 4 dB CNR and an uncertainty of 13 dBW EIRP is detrimental.

4.2.3.2 User data

The user data file has CNR values of previous users inside the beams which we are not allowed to share in this report. These CNR values are heavily influenced by weather, pointing error and the fact that they could have been recorded when the beam contour highlighted a similar area of effect but not exactly the same. These user data entries of beam 29 and 33 of IS33 are also subject to rogue CNR values that are the result of test conditions. In such cases there could be such as antennas that are pointed wrongly, blocking of the LOS path in front of the terminal and other different scenarios that could be saved as user data entries in the user data file. There are also terminals that have been sent in to service which showcase lower than usual CNR values in for example Mars and then reappear much better in July after it has been serviced. Seeing this without knowing all the facts gives the false impression that the seasons of the year could have a 2-3 dB influence which is absolutely not true.

4.2.3.3 The IntelSat BC relative file

The BC-relative file provided by Intelsat has relative beamcenter CNR values at different beam centers. It supposedly can be used to determine the value of any position inside the beam relative to another beam. The BC relative file is used in the way that if you can measure the CN inside the beamcenter of a beam then you can calculate the beamcenter of another beam with the equation:

$$CN_{\text{Measured for beam 1}} - CN_{\text{Relative for beam 1}} = CN_{\text{Unknown for beam 2}} - CN_{\text{Relative for beam 2}} \quad (4.1)$$

The BC relative file has the relative values for IS33, IS37 and H-3e beams. Some of the contours have moved from the time the file was created, specially for the IS37 beam where some of the beam center points given are not inside the beam contours in the OTA-file. Due to fact that the beam center EIRP can not be calculated, this forces a limitation to the algorithm to only consider beams which are represented in the BC-relative file. If this file can be expanded with the maximum received CNR of other HTS-beams in the future the algorithm will include those beams aswell and lift this limitation, but for now the algorithm will only work with IS33 and IS37 beams.

4.2.4 Step 4: CNR reduction to the user position

Depending on the position of the user inside the contours of the beam in respect to the beam center the beam center CNR must be adapted to the terminal location. This can be done in many ways but in this project three methods were investigated and are represented below.

4.2.4.1 Method 1 : Concentric Contour contraction or CCC method

This method contracts the contours given by the OTA-file inwards by a factor of 30% for each iteration.

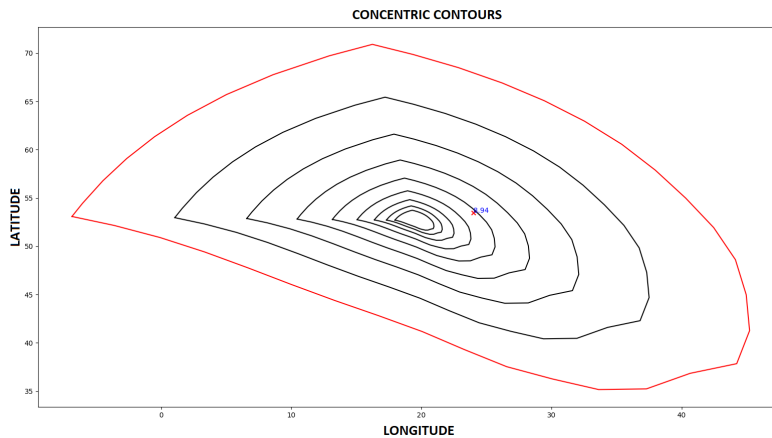


Figure 4.32: Concentric Contour Contraction of the beam 33 of IS33 with beam center as blue marker.

4.2.4.2 Method 2 : Concentric Elliptical Expansion or CEE method

For this method there are ellipses expanded within the contours of the beams, the elliptical center is placed on top of the beamcenter location given in the BC relative file as well the value as can be seen as a red cross. Inside the figures 4.33, 4.36 and 4.39 user data exists which is not presented in this report due to privacy reasons but these values have been averaged and the average can be seen for each ellipse count expanding outwards.

The ellipse is tilted along the angle which transects the beam center from the satellite position according to the Haversine formula. For beam 33 and 28 the ellipses were set to expand in 10 ellipses ranging from the center to the contour edges. For beam 29 that was set to 20 ellipses to encompass a far out positioned point.

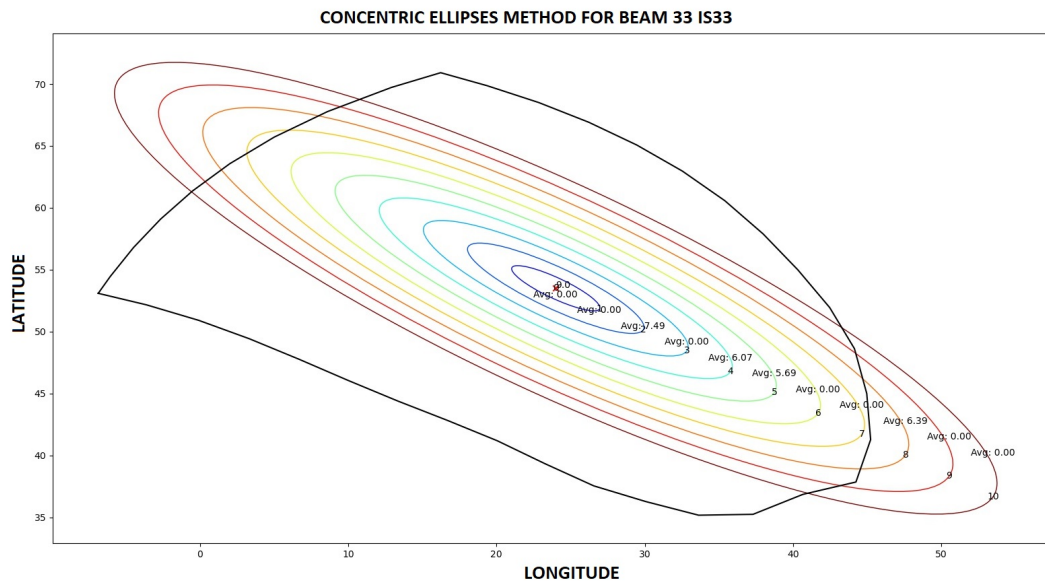


Figure 4.33: Concentric Elliptical Expansion of the beam 33 of IS33 with averaged user data for each ellipse.

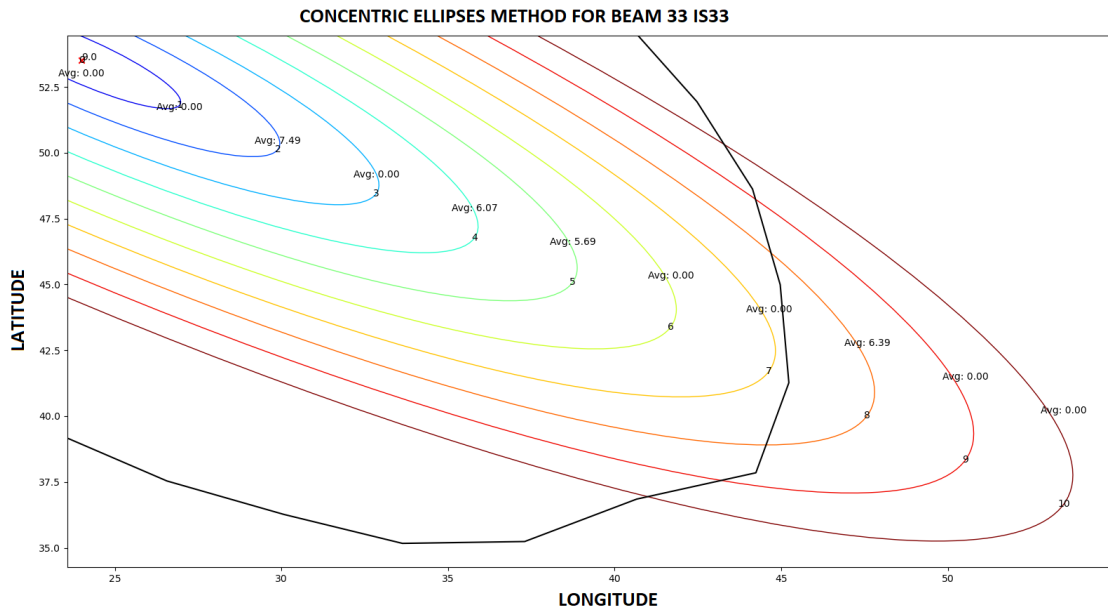


Figure 4.34: Concentric Elliptical Expansion of the beam 33 of IS33 with averaged user data for each ellipse.

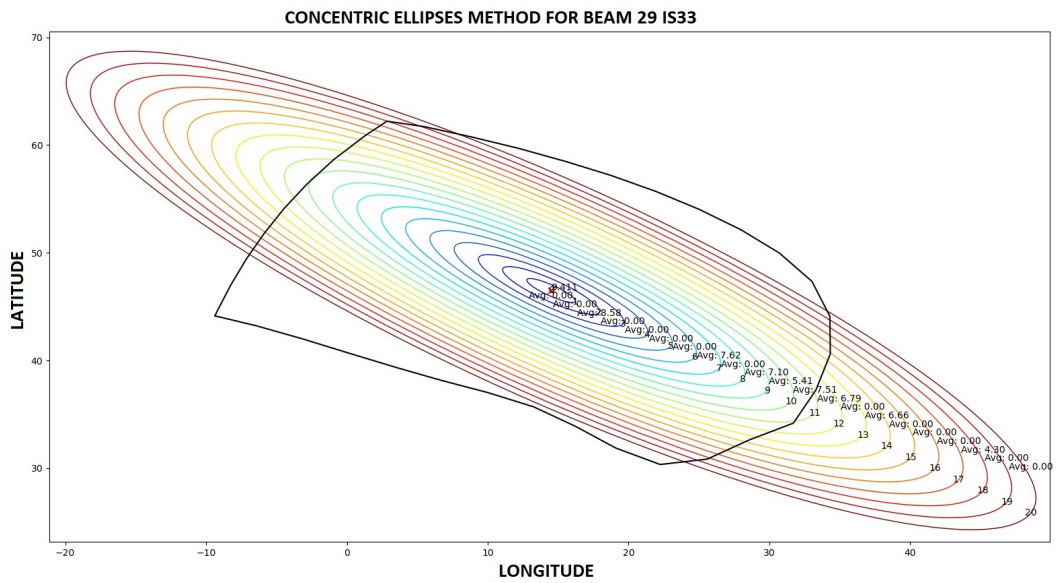


Figure 4.35: Concentric Elliptical Expansion of the beam 29 of IS33 averaged user data for each ellipse.

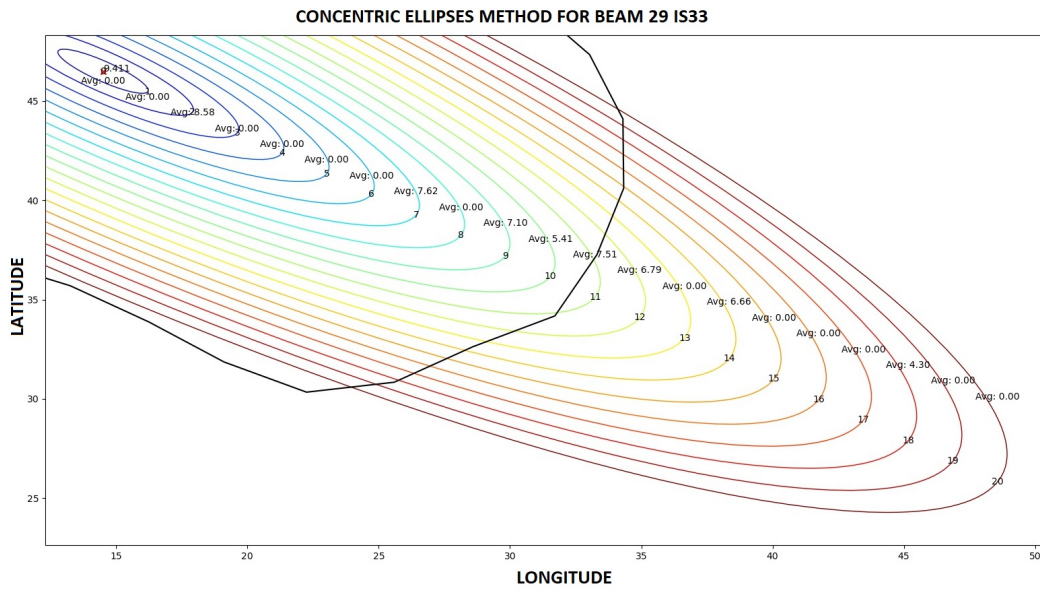


Figure 4.36: Concentric Elliptical Expansion of the beam 29 of IS33 averaged user data for each ellipse.

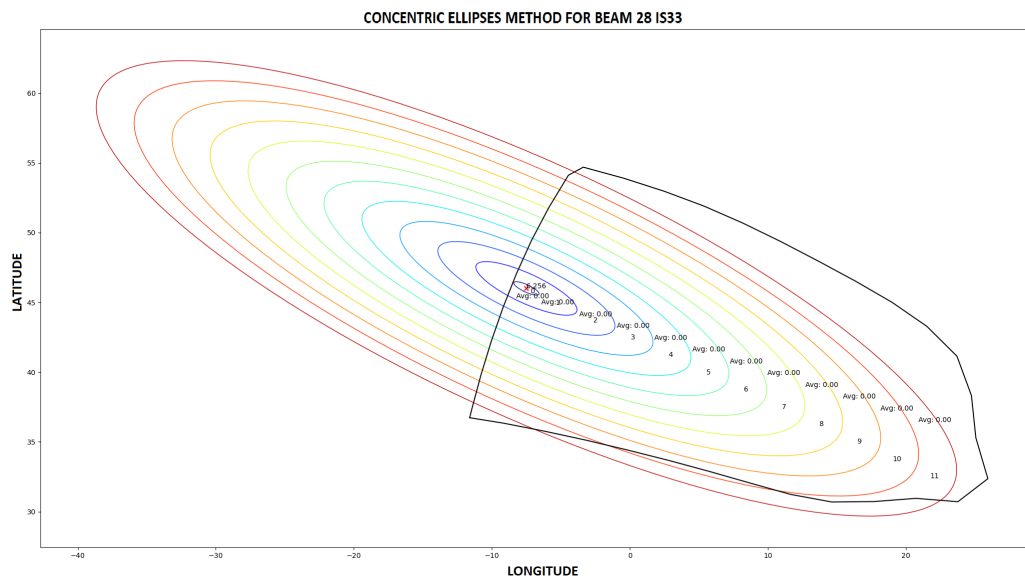


Figure 4.37: Concentric Elliptical Expansion of the beam 28 of IS33 with no user data.

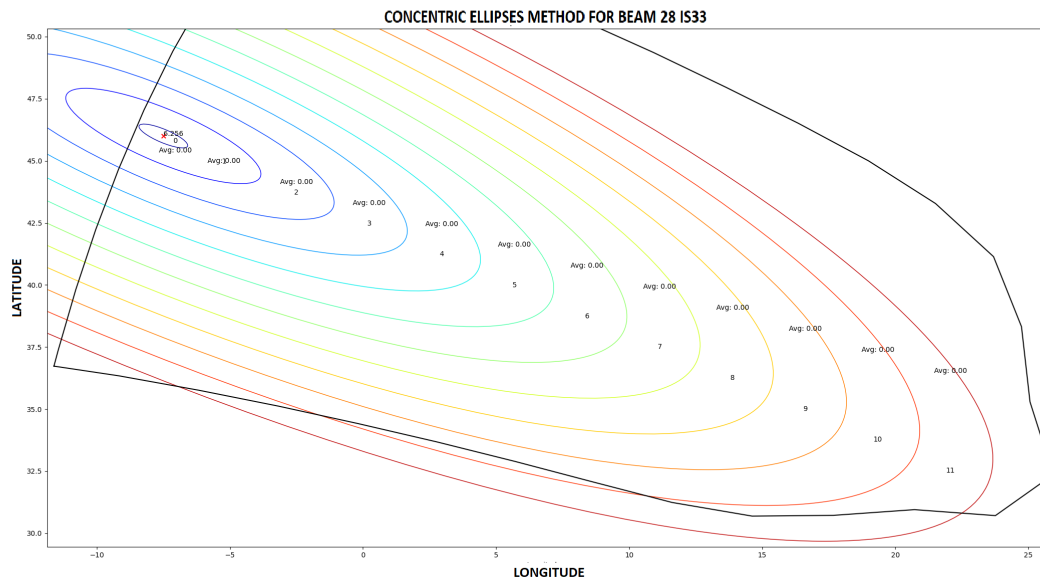


Figure 4.38: Concentric Elliptical Expansion of the beam 28 of IS33 with no user data, a closer look.

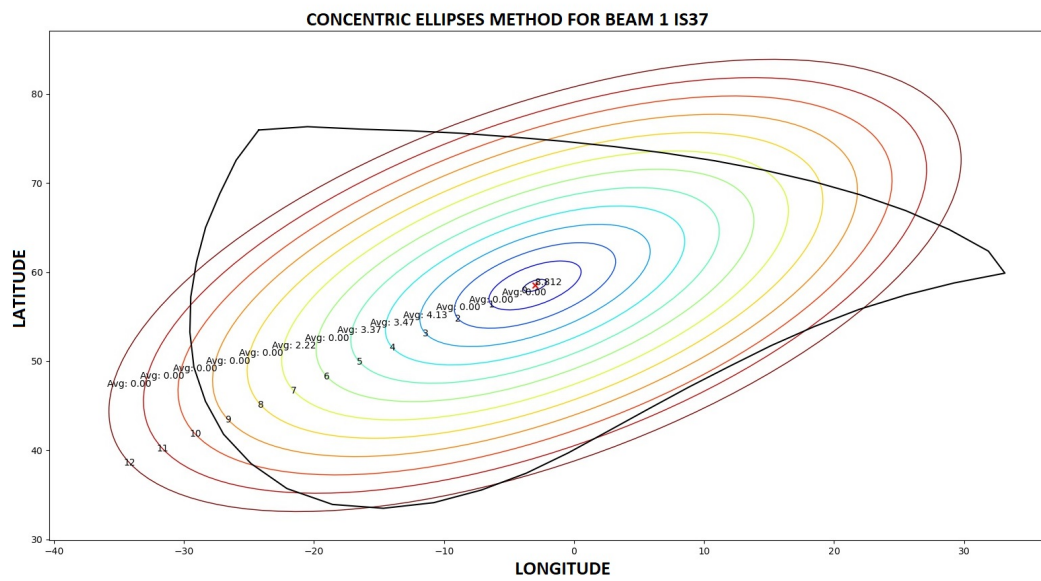


Figure 4.39: Concentric Elliptical Expansion of the beam 1 of IS37 averaged user data for each ellipse.

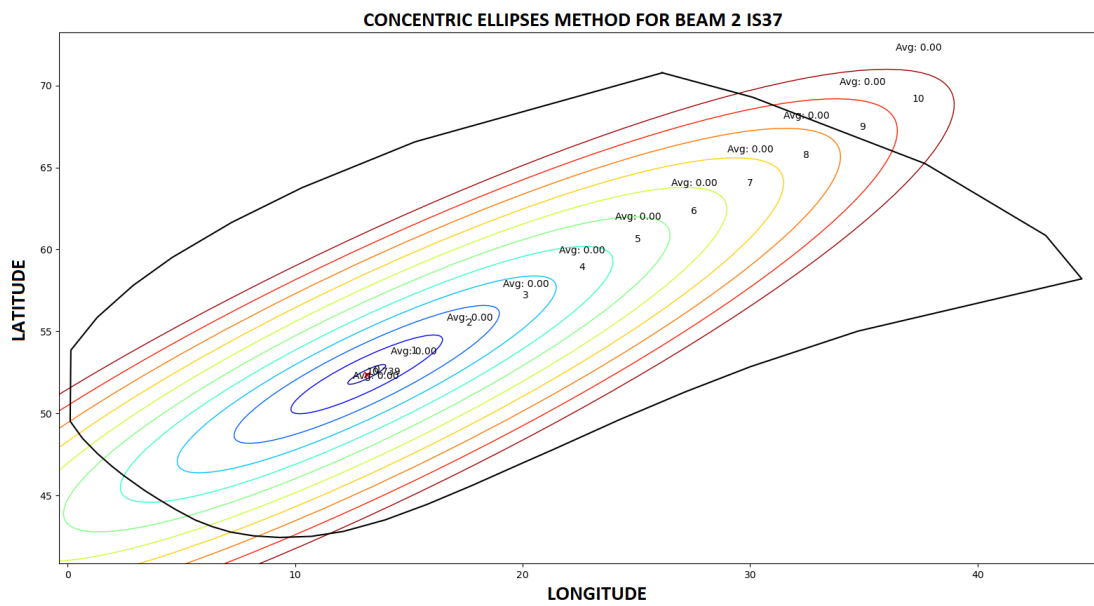


Figure 4.40: Concentric Elliptical Expansion of the beam 2 of IS37.

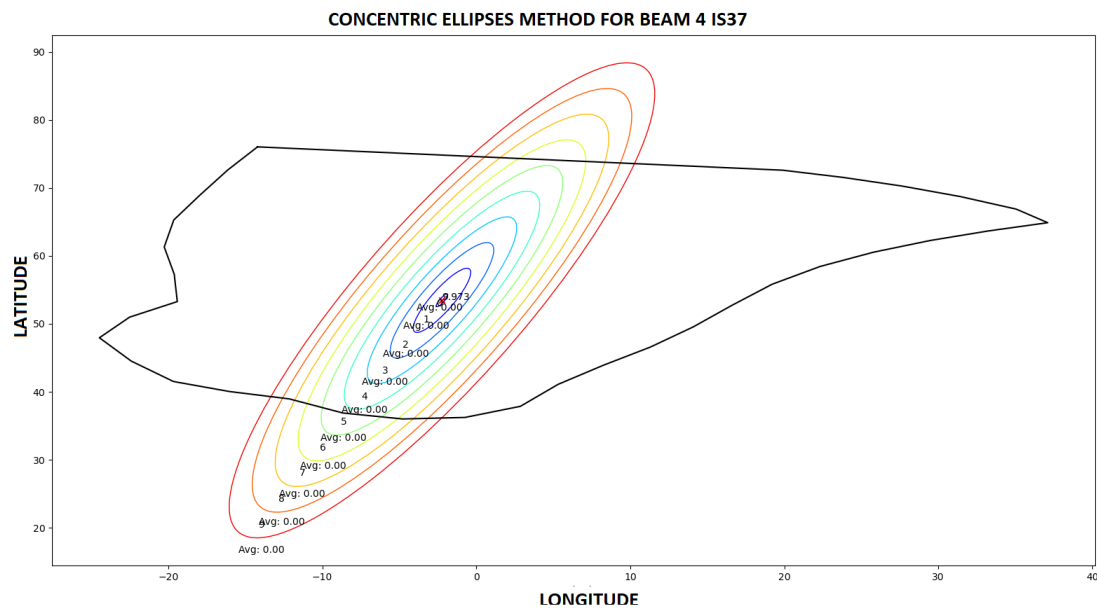


Figure 4.41: Concentric Elliptical Expansion of the beam 4 of IS37.

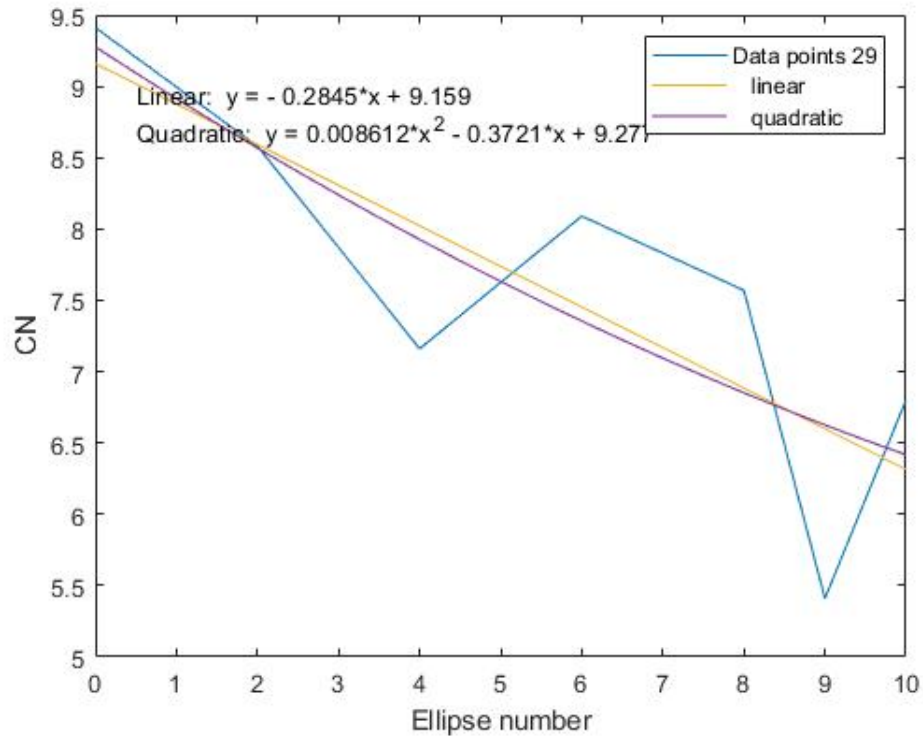


Figure 4.42: Basic fit for the averaged values in beam 29 of IS33.

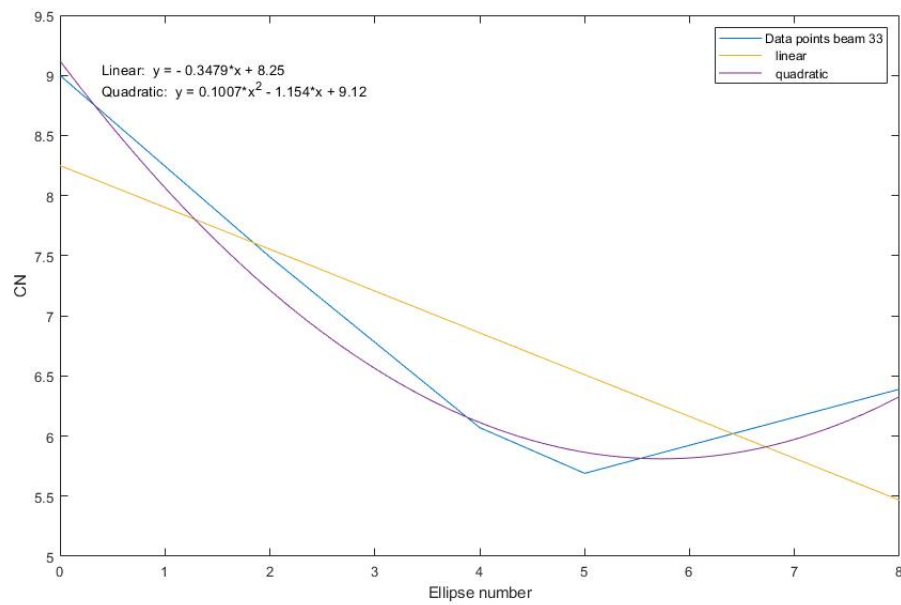


Figure 4.43: Basic fit for the averaged values in beam 33 of IS33.

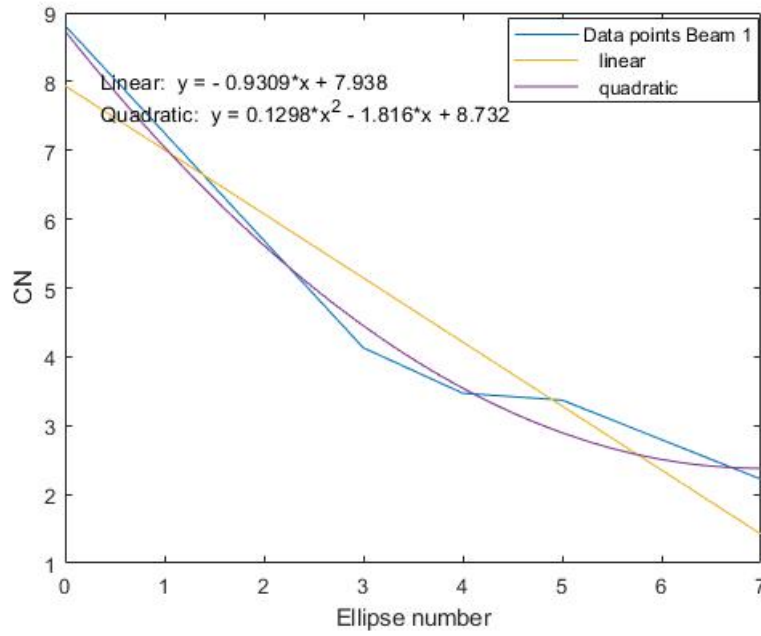


Figure 4.44: Basic fit for the averaged values in beam 1 of IS37.

$$CN_{Prediction} = CN_{Center} - Ellipse\ count * Reduction\ factor \quad (4.2)$$

By observing figures 4.42, 4.43 and 4.44 there is an assumption that each satellite has an individual reduction factor for each ellipse and that the ones chosen for IS33 is 0.3470 and for IS37 is 0.9309.

4.2.4.3 Method 3 : Least Square Fit or LSF method

In this method there is an attempt to try and directly solve for the loss in dB from the beamcenter to the user position with the least square fit method. The idea was that the reduction in CNR due to the position could be divided into two components, x-component and y-component, which impact the loss differently.

$$CN_{Measured} = CN_{Relative\ max} - (A \cdot \sin(\theta + \phi) - B \cdot \cos(\theta + \phi)) * d \quad (4.3)$$

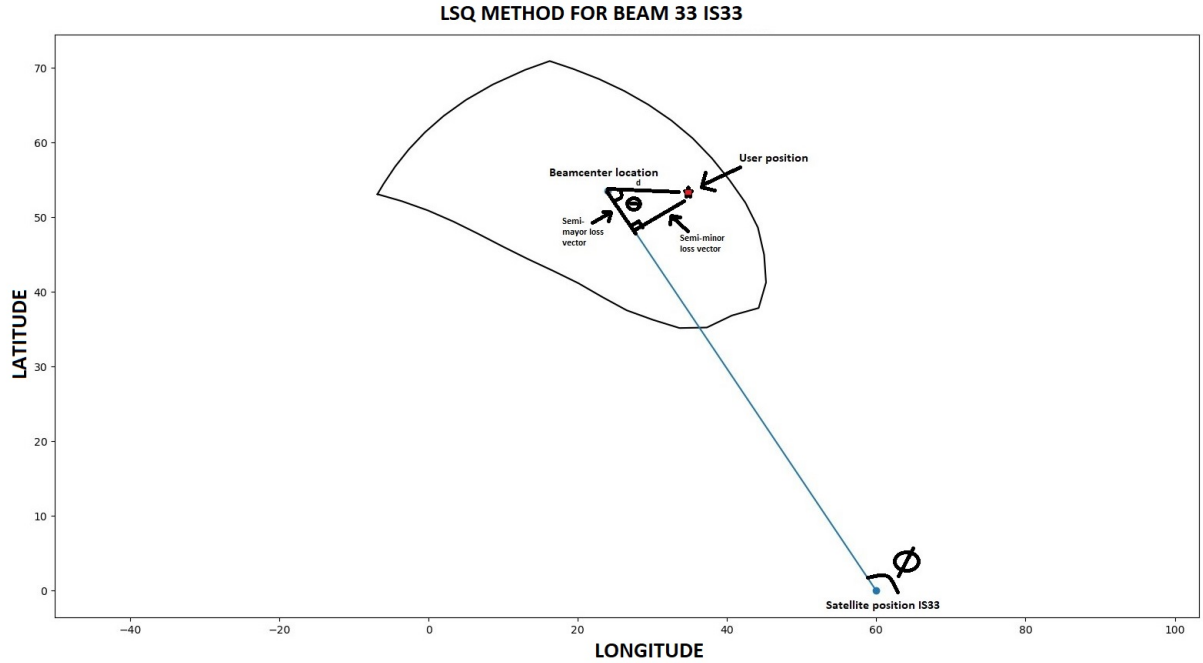


Figure 4.45: Illustration of the idea behind method 3 using Least Square Fit.

The distance d between the points can be found using the Haversine formula and for the case of simplicity we assume the distance between the user position and beam center location is a straight line. The angle θ determines 2 vectors along 2 axis in the contour. One which would go along the ϕ -angle which travels along the semi-major axis inside the contour assuming there existed a ellipse and one axis orthogonal to that axis. By using the LSF method and solving for A and B the idea is to find one set of coefficients that fits the data and can render a solution for any arbitrary user position. There was not much success developing such a function due to the fact that the contour sizes differ greatly and the user data is riddled with error sources that make it hard to use it to solve for A and B . The problem regarding the contour sizes could possibly be solved if the vectors governing the semi-major and semi-minor could be normalized with respect to the total distance to the contour edges. In that way the the Least square fit would use all available user data and create a solution that could be translatable over all beams depending on only the user position, beamcenter position and contour shape.

4.2.5 Step 5: CNR sorted list and comparison with the current algorithm

Moving forward with the CEE-method a test was done to compare the current algorithm in the SatCube Ku with the new algorithm. This is the result showcasing all the steps in the algorithm described above. For this example we only consider IS33 and IS37 satellites.

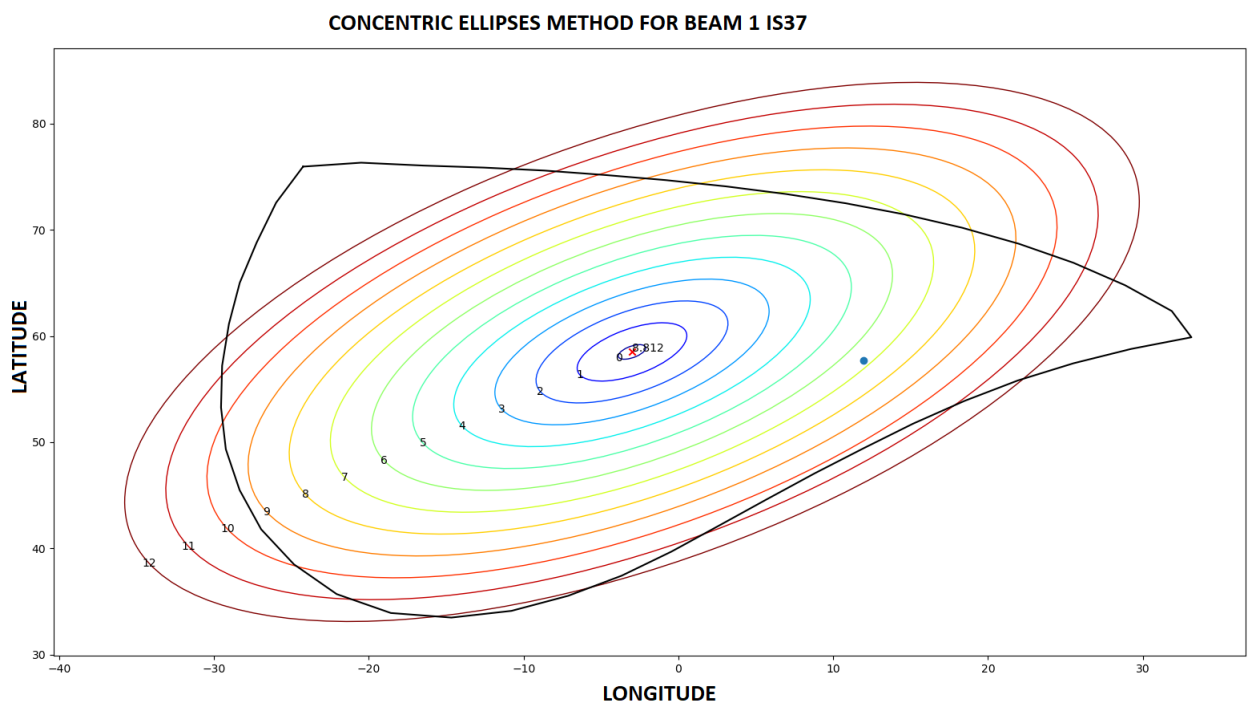


Figure 4.46: Gothenburg test position in beam 1 of IS37 with CEE, 7 ellipses from the center.

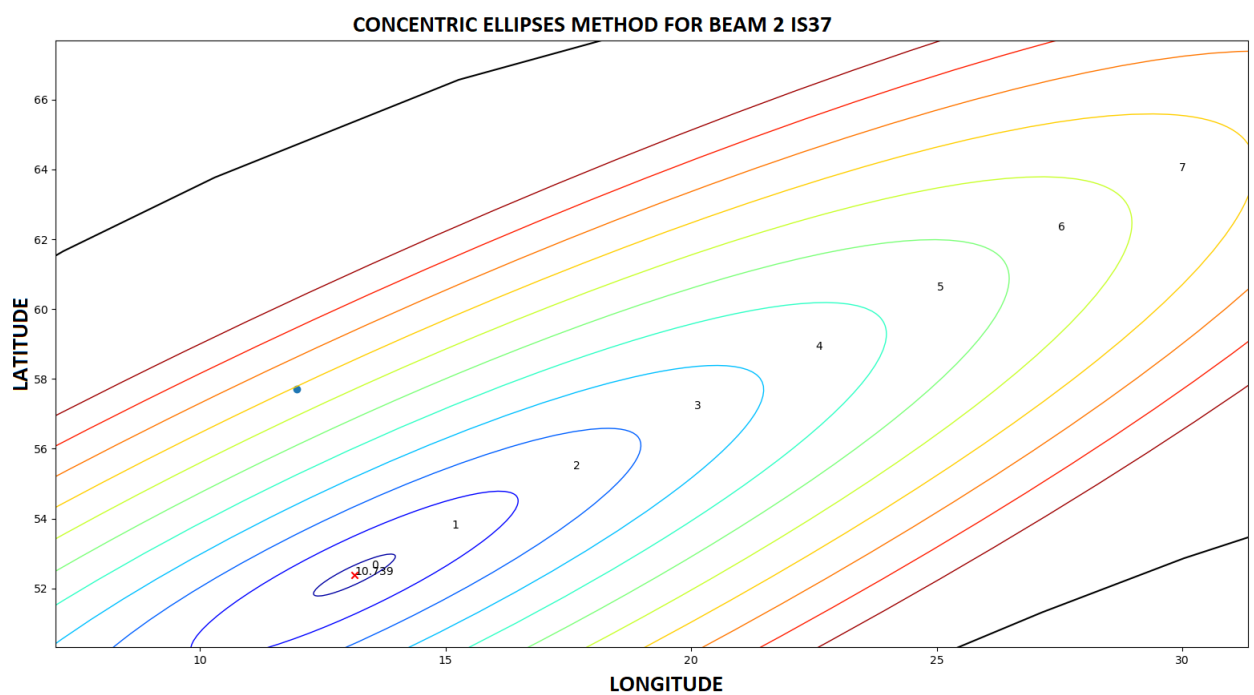


Figure 4.47: Gothenburg position in beam 2 of IS37 with CEE, just about 8 ellipses out from the center.

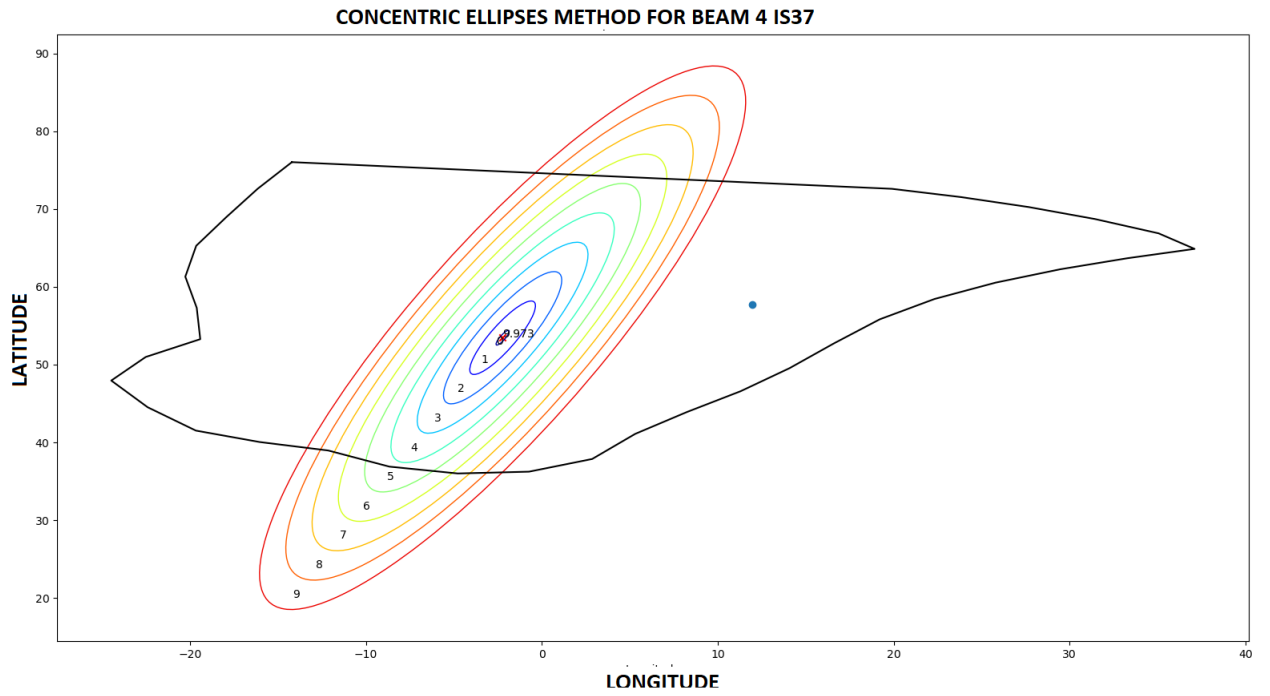


Figure 4.48: Gothenburg position in beam 4 of IS37 with CEE, with more than 10 ellipses out.

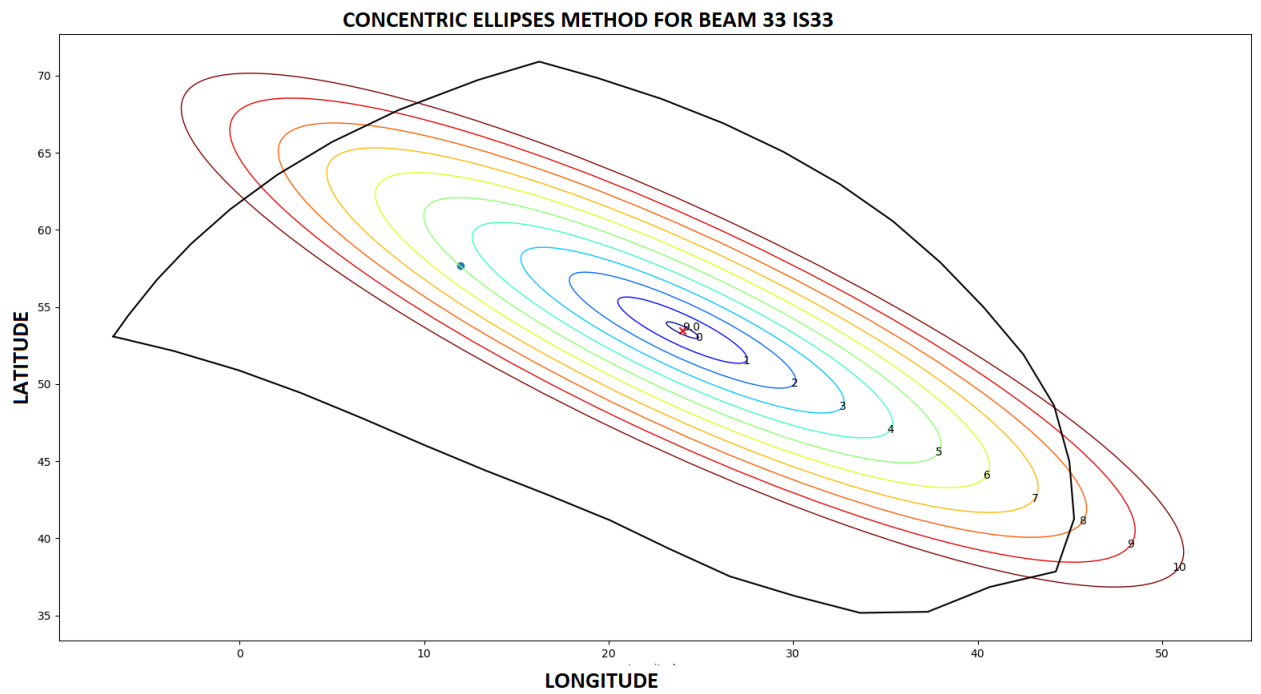


Figure 4.49: Gothenburg position in beam 33 of IS33 with CEE, 5 ellipses from the center.

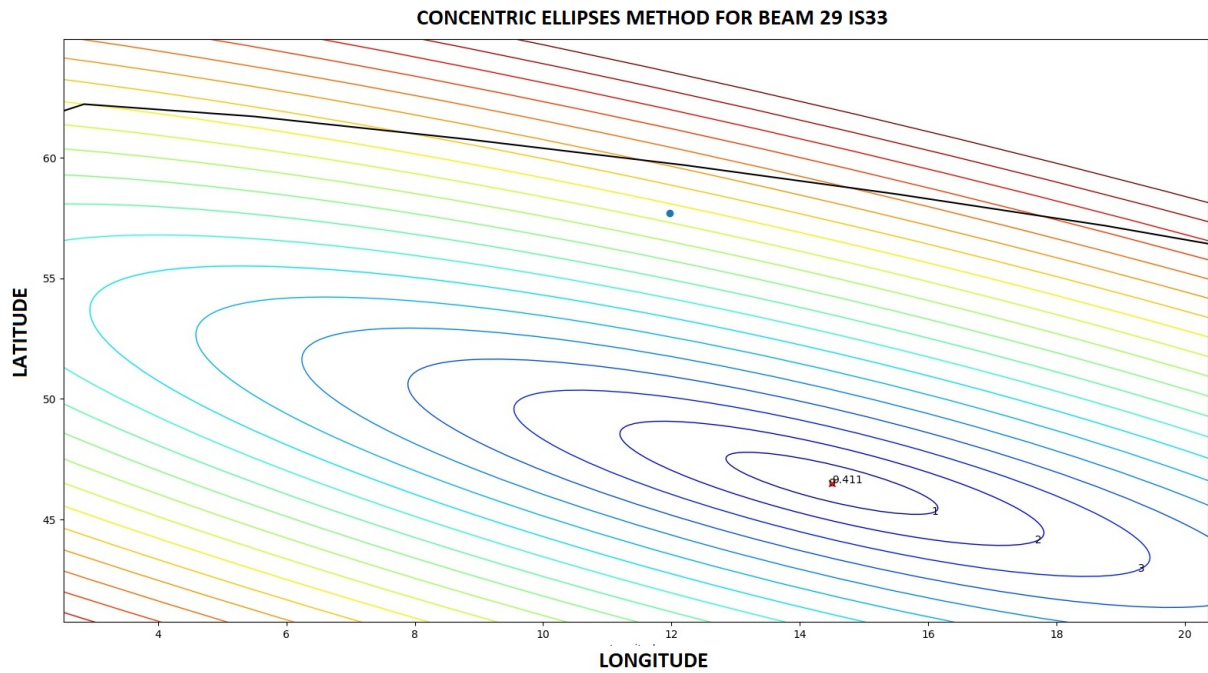


Figure 4.50: Gothenburg position in beam 29 of IS33 with CEE, 12 ellipses from the center.

By using equation 4.3 the predicted values for the beams covering the Gothenburg area are presented in table 4.2.

Table 4.2: The predicted carrier to noise based on the CEE-method for a test position in Gothenburg, a sorted list can be made by changing the Beam 1 and Beam 2 positions based on the predicted values.

List	Predicted CN	Measured value
Beam 33 IS33	7.265	7.1
Beam 29 IS33	5.247	4.5
Beam 1 IS37	2.2957	3.2
Beam 2 IS37	3.2918	3
Beam 4 IS37	less than 0.664	0

Table 4.3: The table of beams available to the user in the test position in Gothenburg in the current beam algorithm.

List	Measured value
IS-37 (18 West) Beam 1	3.2
IS-33 (60 East) Beam 33	7.1
IS-37 (18 West) Beam 2	3
IS-33 (60 East) Beam 29	4.5
IS-37 (18 West) Beam 4	0

Table 4.4: The predicted carrier to noise based on the CEE-method some user positions inside some random beams to thest the algorithm outside of the Gothenburg area.

Other Locations	Predicted CN	Measured value
Beam 31	7.265	6.6
Beam 60	4.013	3.08

5

Conclusion

In this chapter we will discuss the results of this project, if the goals were reached and list possible improvements to specific results.

5.1 User data analysis

The user data file can be a great asset if used correctly, the accuracy of the CEE method is dependent on the user data inside the beams. The user data file consists of both genuine users and engineers doing tests and experiments. These rogue data points cause a havoc when any analysis is attempted to root out performance related inquiries. There are also satellite related sources of error which are hard to quantify. Examples such as if the EIRP have been different during 2019 as it was 2020 or if the user data points have been recorded when the contours covered a different area. It is difficult to know and estimate every source of error that could have a strong influence on the user data. In this project two terminal related sources of error were investigated, the influence in the terminals and pointing error.

5.1.1 Terminal difference

In Figure 4.1 to 4.4 the terminals are connected to 3 different satellite, the 0.8 west and 1 west are the same satellites but saved under different "satellite pointing longitude" direction in the user data file. To mitigate other influencing parameters and subsequently only see the effects of the different terminals, strong constraints were instituted on the user data entries. The figures only show CN values when the data points were recorded to be connected to the same satellite, when there is no rain, cloud density less than 20%, 100 by 100 meters around the Almedal office area and a error in the elevation and polarization pointing less than 0.2 degrees from the target angles. There were also measurements done by us which can be seen in table 4.1, Unfortunately the terminal ids was not recorded. The measurements in table 1 is also set in the peak azimuth angle for each beam from around the Satcube Almedal office area.

By looking at figure 4.6 and 4.11 the terminals 1 and 24 connected to the same satellite and frequency in their peak azimuth pointing angle showcase a difference of roughly 0.6 dB. In figure 4.8 and 4.9 the peak azimuth angle values collected there by terminal 52 and 230 show a difference of roughly 0.4 dB. It is also crucial to point out 2 points on a plot is not much to go on usually but it is all we have to work with here. The results of our own measurements shown in table 4.1 show that for

the same terminal the differences in CN between different measurements is less than 0.3 dB. For the difference inbetween terminals disregarding the cloudyness of the day the measurement is between 0.1 to 0.8 dB. The datapoints recorded after the "New terminal 1" change are confusingly high and that terminal is most probably somewhat special to achieve such better than usual results.

5.1.2 Azimuth difference

The terminal can not establish a duplex link if it is not within 1 degree of the pointing target angle in polarization and elevation. For the azimuth the requirements are more unrestricted, due to local interference the user can be as much as 20 degrees off target and must sweep left and right to find the signal. There are no ways to know if the recorded data points in the user data file are of users who take the time to achieve the most optimum received CNR before establishing the link or not. Because of this the azimuth pointing angle was investigated in the same manner as the terminal difference case. The same constraints apply here aswell with an addition of only looking at the data points which were recorded in quick session of each other. Depending on the position of each set of data points a difference of 50 meters in location can equal a difference of 20 degrees. By looking at figures 4.6 and 4.10 it seems as if the azimuth pointing angle has a peak and a quadratic polynomial shape around that peak. Assuming that the peak in figure 4.8 is located at 212.8 degrees and the border is at 210.6, there is a span of 2.2 degrees in which the beam loses 30% in dB. The error factor due to the azimuth pointing error is an unknown variable. A way to deal with this and still make the user data something that can potentially be useful is to only consider the maximum values recorded during measurements done by the same terminal, in the same area, connected to the same frequency and during a small time interval. Unfortunately the users do not always record several points with different azimuth orientation angles, many user data points are of simple point measurements and there is no way to determine how many degrees they are off target. Here, for the case of how much every user entry in the user data file is off target in relation to what could have been the true angle we hypothesize that they, for the majority of the case, should find the beam within the same region as is required in polarization and azimuth. Using that hypothesis as a foundation for the uncertainty introduced by the user data entries, it should be roughly 0.5 dB, as can be seen in figure 4.6. Figure 4.9 shows a interesting difference between 2 collected entries. Considering all the constraints applied to the user data entries, when the azimuth angle is the same there is still a difference of 0.15 dB. A 0.15 dB difference of the unknown gives a sense of a resolution when these constraints are applied to the data points.

Considering improvements that could potentially help the user locate the azimuth more quickly and more accurately could be the idea of a floating azimuth target angle. Considering the behaviour of the users to connect several times to the same beam, from the same location in a short span of time with varying results, the future measurements could rely on the previous pointing angles and shift the target angle closer to where the beam actually is in respect to the user. This could potentially decrease the time it takes to come online if the user perhaps chooses to change

beams and then change back from the same position. This way there is a history on the terminal of the previous link and therefor a much more accurate azimuth target angle. For this endeavor the GNSS positioning resolution becomes key to decide if the user is stationary or not. Every beam the user connects to while in that position can be used in this manner and utilize the efforts of the user to help future azimuth pointing.

5.2 Beam algorithm

In this section we will discuss the steps in the beam algorithm.

5.2.1 Step 1: Position and all available beams

The user position is the only input that is available for the terminal. The polygon method of creating polygons of the contours given by the OTA-file in the terminal is the best way to translate the user position into a list of available beams in that area. This is used in the current beam algorithm in the Satcube terminal and what is suggested in this new beam algorithm.

5.2.1.1 Problems

5.2.1.1.1 Contours The final process of the beam worked as intended. However, it is very reliant of a continuous beam contour. It was noticed that occasionally beam contours provided by the OTA-files would be distorted.

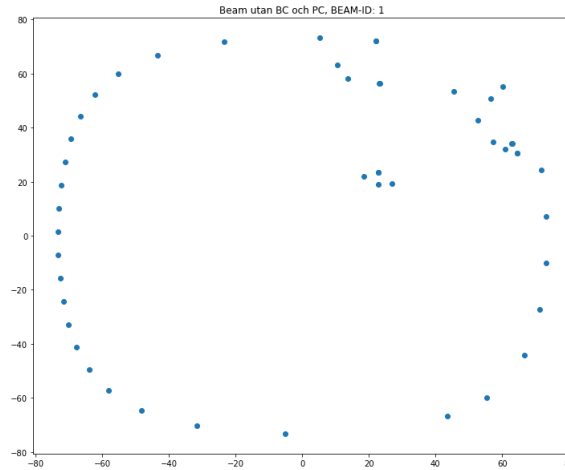


Figure 5.1: Example of an unreliable beam contour

Figure 5.1 is an example of a distorted beam contour and would be considered unreliable. Passing a contour of that nature to the polygon function could result in an inaccurate polygon being created and thus an inaccurate polygon center.

5.2.1.1.2 Meridian crossing A major issue when handling the beam contours was when a beam contour crossed meridians on Earth. Primarily, issues were found

when either the anti-meridian or the prime-meridian was crossed. The anti-meridian is the line where longitude is 180° and the prime-meridian is where the longitude is 0° . The background behind this issue is in how the contour data is handled. Longitude coordinates provided are defined in the range $0 - 360^\circ$. This caused an issue with beams around the prime meridian where contours would get split up and effectively creating two separate contours on opposite sides of the longitude axis. A separated beam can not be fed into the polygon function as it will lead to a polygon being created from one edge of the meridian, across the world to the other edge. Creating the polygon is the foundation of geometrically handle the beams, a faulty polygon will therefore ruin the algorithm. This issue was rather difficult and required effort. The first alternative that was looked into was to handle the two sides of the split beam each. This means to patch up the separated beam ultimately creating two new beams.

To achieve this, new data was interpolated between the minimum and maximum latitude values on each separated side. The interpolated data points is seen as the patch for the separated contours. Approaching the problem with this method problematic however. As the goal is to find a beam center of one beam, splitting the initial beam will cause issues. Firstly, the polygon function will not be able to handle the incoherent contours. If a polygon is to be created, the contours has to be further handled presumably by dividing them into new variables and eventually conjoined. This is an elaborate process and luckily an alternate method was found instead.

By further analyzing the contours it was noticed that the meridian issues was caused by the value range of the longitude values. Longitudes in the value $-180 - 180^\circ$ will cause issues around the anti-meridian and the value range of $0 - 360^\circ$ will cause issued around the prime meridian. It was ultimately decided to define longitude in the range of $-180 - 180^\circ$. This was accomplished by subtracting 360 from all longitude values over 180. As mentioned, the main issue arose when feeding a split beam into the polygon function. The solution was then to feed a beam with a continuous contour to the function. A split beam can be temporarily transformed into a complete beam by shifting the data points to the prime meridian, 0° longitude axis. To do this the contour data points was divided into two sections, positive longitude ($\lambda > 0$) and negative longitude ($\lambda < 0$). All positive points were subtracted by 180 and 180 was added to all the negative points. When this is done a continuous beam contour is created around the prime meridian, this contour can then be passed to the polygon function.

The typical procedure is then applied, first is a polygon created and then the polygon center is calculated. After that all of the data points are transferred back to their original position. This time 180 is added to the negative longitude values and 180 is subtracted from the positive. The polygon center is also shifted depending on the same condition. Shifting back to the original position is crucial for the beam center correction where the position of the polygon center is one key parameter.

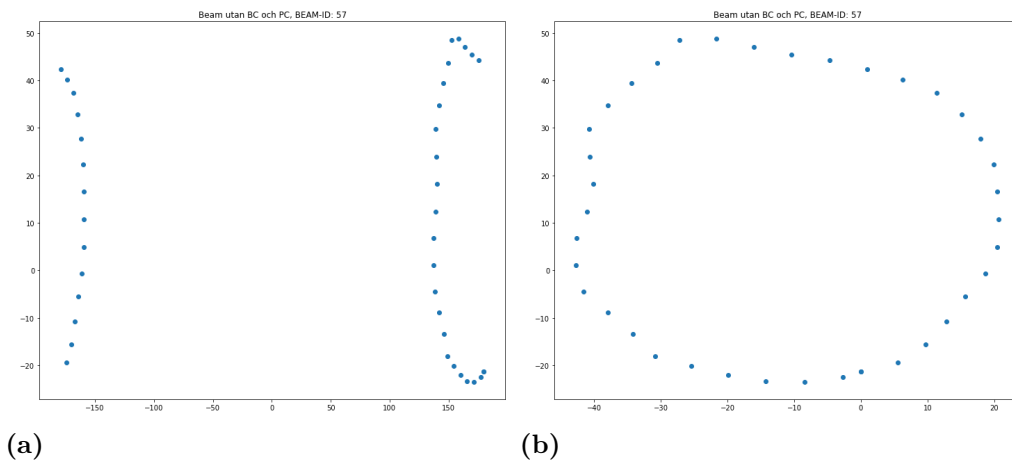


Figure 5.2: a: An example of a split HTS-beam. b: The same beam shifted around the prime meridian .

5.2.2 Step 2: Beamcenter of beams

It was clear from the start that a polygon center is not an accurate estimation of the beams center. By analyzing the projection of the beams on Earth with help from Intelsat and Satbeams it was clear where to expect a beam center given its contour. HTS-beams was of relevance when analyzing beams. FSS-beams and large beams were a limitation to the analysis as some of these beams have obscure contours and large FSS-beams were of lower priority. As mentioned in the results chapter, the polygon center is only accurate for circular beams. However, the new beam center correction algorithm does not correct the center point much for circular beams as they are close to the satellite. The correction function works as expected as the function is built upon data that can be considered correct. This function alone does not solve the issue but is built to get a better understanding of the distance to the beam center. In a few of the analyzed scenarios the difference in distance between the corrected beam center and polygon center was measured up towards 700km. This is dependant on the distance from the satellite as mentioned.

A further development was also discussed to the beam center estimation. Instead of estimating a beam center point, a beam center area could be a more accurate representation. This is due to the shape of the transmitted beam from the satellite. It is discussable if this approach would be beneficial to the algorithm. This solution was considered at a later period in the project, it was therefore not implemented and tested as it would require significant more investment in time and energy. The reason for not implementing this solution is primarily due to the focus being on other areas.

5.2.3 Step 3: CNR value at beamcenter

The CNR at the beamcenter proved to be very difficult to connect to the beam characteristics given in the OTA-file, such as contour or satellite position. Intelsat

and other 3rd party satellite beam footprint providers also came to be of no use as the uncertainty given exceeded 10 dB when the difference between measured beams was approximately 6 dB. This is where the beam algorithm becomes limited to only deal with the IS33 and IS37 satellites by using the Intelsat BC-relative file. This file gives the position of the beamcenter and the relative CNR across all beams for each satellite individually. If Intelsat decides to move the contours to a new position the 2 previous steps in this algorithm will adjust accordingly. But if Intelsat decides to change the output power of certain beams that change must be updated to the BC-relative file used in this step for this algorithm to function accordingly. This is another area where the user data can be implemented to help observe changes in EIRP of the satellite or even expand the BC-relative file to new beams and new satellites that are introduced in the future.

5.2.4 Step 4: CNR reduction to the user position

There are three methods examined in this part and each method has its own flaws and potential.

5.2.4.1 Method 1 : Concentric Contour contraction or CCC method

The CCC method is useful for circular HTS-beams. It does not expand from the beamcenter but contracts from the contours inwards. This is a problem as the beam center does not necessarily require to be in the center of the beam. Furthermore it also incorrectly assumes the first contour section to be the same, as is known the edges of the most far out stretched HTS-beams are not the borders of the contour but a cutoff due to the edge of the planet. As can be seen in figure 4.37. This method may not be useful for modelling the shape of the HTS-beams but it could be used as a basis for non-HTS beams in the Intelsat satellite fleet. Those beams are of irregular shapes and never cutoff by the edges of the planet. The EIRP inside the non-HTS beams are more irregular and therefore needs special handling, but as a basis the CCC-method could be useful.

5.2.4.2 Method 2 : Concentric Elliptical Expansion or CEE method

This method is what was chosen to be the most appropriate method for the beam algorithm. The user data integrated into the contours are averaged for each ellipse and then used to fit mathematical models to the user data. Considering beam 33 has more and well distributed points across the contour that curve-fit was used for all beams for the IS33 satellite. Beam 1 for IS37 was also curve fitted and used to derive a reduction factor for each ellipse. The method itself is very complex and relies on the combination of 3 different angles to create the ellipses shown in figures 4.44 to 4.52. There is currently a bug which sometimes rotates the ellipses 90 degrees off center and the reason is unknown. The method also assumes spherical Earth which is inconsistent with the contours shape and sizes, this is why the beams appear to be angled slightly off from the contours they are expanded within. There is much room for improvement, but for now this is good enough to show the idea of an elliptical solution. An interesting find was that the reduction value calculated in figure 4.43

and 4.42 are much more similar than for 4.44. Here we had made the assumption that the beams transmitted by the same satellite will have the same reduction factor as could be the result of antenna design. There is the need for more testing and curve fitting more beams to come to a correct conclusion.

5.2.4.3 Method 3 : Least Square Fit or LSF method

The LSF method is similar to the CEE-method in the way that instead of ellipses that are averaged this method instead directly uses the user data entries to find the losses caused due to the displacement in the ϕ -plane and orthogonal to that plane as can be seen in figure 4.56. The idea is that the received CNR at the user position should be different whether you are located along the ϕ -plane as supposed to cross the ϕ -plane. The position of the user in respect to the beam center can be expressed in an angle θ and the distances can be calculated. The user data can be used to find the A- and B-coefficients to fit the data. There were issues with this method due to the fact that the contours are not equal in size, there needs to be an addition to equation 4.3 to include the difference in contour sizes. The way this method differs from the CCC and CEE-method is that the user data can be combined cross many beams to solve for how much the CNR must be reduced to the position of the user. Obviously such a method requires a more filtered user data points and a comprehensive non-linear data analysis methods approach which is out of scope for this project.

5.2.5 Step 5: CNR sorted list and comparison with the current algorithm

The results presented in table 4.2 is satisfactory. What is noteworthy is that the difference between the measured value and predicted CN in beam 29 is 0.747 which is approximately 2 ellipses further from being perfectly correct. For the Beam 1 case the difference is 0.9043 dB which is 1 ellipse short of being perfectly aligned. This could be the result of the elliptical expansion on a spherical Earth, further improvements on this matter could possibly yield an even better result.

The list which is presented in table 4.2 is based on the user-data of actual users with different terminals, different weather conditions, different pointing errors and also other sources of error which have not been investigated in this project due to time limits. There is a minimum uncertainty of 0.3 to 0.8 dB which is caused by the difference of terminals, another 1 to 2 dB which could be caused by rain and clouds, a roughly estimated azimuth uncertainty contribution of another 0.5 dB and other unknown source of errors that are not dealt with in this project. Considering how the beams 1,29 and 33 are the most populated beams of all the ones in the user data many of the more common sources of errors are averaged out in the process of CEE. But because of the points used in the analysis and the curve fit in the CEE-method the accuracy of the algorithm could be between 0.3 to 3 dB depending on the weather conditions. The points effected by heavy rain which cause a CNR loss of up to a 2 dB are far less than the other points recorded in a area during a long period of time. When averaged over a large sample size, these high loss values and

uncertainty in total becomes less than 2 dB to the total uncertainty of the model. It would perhaps be more suitable to attribute losses due to weather to the curve fits made in beam 29, 33 and 1 to be less than 1 dB as there are many points recorded in these beams which the curve fits are designed of. The total uncertainty would then be between 0.3 and roughly 1.5 which agrees with the tables 4.4 and 4.2.

5.2.6 For future projects

Considering the user data is a tool used to determine the accuracy of the algorithm and methods there should be an effort made to save more than one entry for each link. For example By collecting the CN value of a user over a period of time the effects of weather or pointing error can be averaged and estimate more easily. The algorithm is limited by the BC-relative file and a study into how the maximum carrier to noise could be derived could help the algorithm to expand to more satellites and beams. A suggestion in this case could be if there are new beams introduced in satellites where other beams contain BC-relative values, then by using the CEE method you could estimate the beamcenter by moving from the user position to the beamcenter instead, and adding the reduction factor to the ellipses.

5.3 Conclusions and areas of improvement

As this project contains a lot of uncertainties and unknown parameters it has been difficult to know if any optimizations were possible from the start. The current algorithm works given its very limited input parameters. However, the limitations to the algorithm are connected to the limited amount of information the algorithm works with. The goal of this project was therefore to investigate the limitations and the uncertainties of the current algorithm in order to find ways to improve the beam selection. To narrow down the scope it was decided early on that primarily user data and geographical data was to be utilized.

Evaluations of all information within the projects scope was performed regularly. The planning has therefore not been followed thoroughly as new ideas have been discovered and tested along the way. This was somewhat expected as analysis was a large part of this project. With so many parameters affecting the beam selection algorithm it was never clear what to prioritize and the path to improve the algorithm was changed frequently.

Bibliography

- [1] Intelsat fleetmaps footprints, Intelsat 33e at 60° E: Key parameters
Intelsat coverage maps,2023-06-12
https://www.intelsat.com/fleetmaps/satellites/satellite_39/
- [2] Intelsat fleetmaps footprints, Intelsat 37e at 342° E: Key parameters
Intelsat coverage maps,2023-06-12
https://www.intelsat.com/fleetmaps/satellites/satellite_32/
- [3] Satellite footprints, Satbeams footprints
Satellite footprint maps,2023-06-12
<https://www.satbeams.com/footprints>
- [4] Electromagnetic wave polarization, Polarization
Polarization,2023-06-12
https://www.researchgate.net/figure/1-Polarization-types-of-the-electromagnetic-wave_fig13_322908774
- [5] Calculation of elevation calculation, Elevation
Elevation calculation,2023-06-12
<https://space.stackexchange.com/questions/58041/how-to-calculate-azimuth-elevation-of-a-satellite>
- [6] How to calculate the azimuth, Azimuth
Azimuth calculation,2023-06-13
<https://www.omnicalculator.com/other/azimuth>
- [7] What is the Impact of Rain on RF Signal Propagation?, Attenuation
Attenuation due to rain,2023-06-27
<https://www.everythingrf.com/community/what-is-the-impact-of-rain-on-rf-signal->
- [8] Putting small antennas on the spot,Telenorsat
Telenorsat,2023-07-04
<https://www.telenorsat.com/putting-small-antennas-on-the-spot/>

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se



CHALMERS
UNIVERSITY OF TECHNOLOGY