RFI Estimation from Non-GSO Satellites Based on Two Line Element Assisted Equivalent Power Flux Density Calculations

Tom Hartman*, Niek Moonen*, Frank Leferink*†
*University of Twente, Enschede, Netherlands, t.h.f.hartman@student.utwente.nl
†Thales Nederland B.V., Hengelo, Netherlands

Abstract—**As a result of out-of-band emissions and spurious emissions from air-borne or space-borne transmitters simulations and analysis methods have been developed. These methods were developed to determine the amount of interference such a transmitter could cause. The method discussed and applied in this paper is the equivalent power flux density (EPFD) method, which is formally described in international telecommunication union (ITU) recommendations. Up until this point this method has been implemented in MATLAB, which can be used for the determination of the amount of data loss in radio telescopes due to the unwanted emissions of the Iridium satellite constellation. In this paper, it is proposed to use the two line elements (TLEs) instead of the licensed Satellite ToolKit in the EPFD calculation. A theoretical case has been implemented, which is adaptable to the user preferences. The flexibility of the algorithm has been improved, while computational time remains in the same order of magnitude.**

I. INTRODUCTION

In radio astronomy, extremely sensitive radio antennas are used to detect very faint radio signals of cosmic origin. The strength of these signals are several orders of magnitude smaller than the radio signals used in for instance telecommunication services. Due to the radio frequency interference (RFI) susceptibility of the antennas, mitigation techniques need to be developed, for which RFI investigations are required [1]. The international telecommunication union (ITU) regulates the division of the radio spectrum and allocates frequency bands to different radio services. Some are dedicated solely to radio astronomy. This allocation means that radio astronomy can claim protection within these bands, however out-of-band emissions and spurious emissions may end up in these allocated bands and could then cause harmful interference [2]. To determine the RFI quantitatively, e.g. from the Iridium satellites, a simulation, measurements, and analysis methods have been developed [3]–[5]. For satellites in non-stationary orbits [6], such a method is the so-called equivalent power flux density (EPFD) method. This method calculates the detrimental influence of the several satellites in a certain constellation on a telescope. The power of this method lies in the fact that it takes the time invariant state of the system in consideration and creates an equivalent power flux density (PFD) evaluated at boresight.

In ECC Report 247 [5], the EPFD method and its implementation in MATLAB are described. In this paper, the downlink of the interference from non-geostationary orbiting (GSO) satellites onto radio telescopes is studied.

The challenges in the RFI determination are related to the non-stationary orbits. Gain and free space path loss (FSPL) are dependent on the relative position of an individual satellite, which is time varying. Info about satellite orbits are thus required for accurate estimation. Various solutions can be found for calculating or estimating orbits for individual satellites. The MATLAB implementation described in ECC Report 247 uses the Satellite ToolKit and is dependent on measurement data [5]. Keeping in mind that one requires a simplified perturbation model to estimate the state (i.e. position and velocity) of a satellite for any given time, the two line element (TLE) of the Iridium constellation was implemented. However, it is not limited to this particular constellation. Theoretically, it is required to consider the RFI contribution from all possible satellite constellations, taking into account their emitted electromagnetic interference (EMI) to create one overall EPFD evaluation.

Even with the use of the TLEs, the Python implementation still needed measurement data, e.g. from the Leeheim station, to make an analysis, which still kept the code restricted. For this reason, a theoretical case was implemented. This way, a quick theoretical calculation can be done to get a feeling of what impact a certain, possibly theoretical, constellation can have. This theoretical PFD can then make use of the positional data from the TLEs and makes it so that no stochastic approach is needed. This means that in this paper it is proposed to use the open-source format TLE to introduce flexibility in evaluating different satellite constellations and different radio telescopes, while using a deterministic PFD.

This paper starts with a theoretical background of the EPFD method, which is then followed up by a theoretical explanation of the stochastic process involved in the EPFD calculations. Then, the implementation is explained, where the focus lies on adding the TLEs. After this, the results are given followed up by a conclusion.

II. THEORY: EPFD CALCULATION

In case of GSO satellites, a simplified method for calculating the received power based on PFD is used, as the relative position of the satellite compared to the observer's position is constant. In case of non-GSO satellites, the relative position is varying, thus EPFD is proposed to incorporate the time invariant state of the system. This section derives an equation for calculating the EPFD. It originates from the overall power transfer function of a radio channel.

A. Power flux density

Friis transmission formula is seen in many different styles of writing. It simplifies radio link calculations to its basic components:

$$\frac{P_r}{P_e} = \frac{G_e G_r}{L_{fs}} = D_e D_r \left(\frac{\lambda}{4\pi d}\right)^2 = \frac{A_r A_e}{d^2 \lambda^2}$$

Where:

- $P_{e,r}$ - Power;
- $G_{e,r}$ - Gain;
- L_{fs} - Path loss;
- $D_{e,r}$ - Directivity;
- $A_{e,r}$ - Effective aperture area;

For RFI calculations, the same principle holds. In this case, the satellites are the emitting antennas, while the radio-telescope is considered to be the receiving antenna. Based on the assumption of ideal propagation, FSPL can be calculated from the distance between transmitter and receiver. The PFD is then defined as the amount of power per surface area that hits an observer that originated from an emitter at a certain distance, d:

$$\text{PFD} = \frac{P_e G_e}{4\pi d^2}$$

Note that the gain of the emitter includes directivity of the antenna and compensates for the assumption that transmitted power is distributed over a sphere with radius d, as if it were a isotropic radiator. The more commonly used form for PFD, in the analysis of EM radiation, is the flux of the Poynting vector. Combining the PFD and the assumed known effective aperture surface area (A_r) results in the received power. When considering a constellation of GSO satellites, the total PFD is a superposition of all contributing satellites:

$$\text{PFD} = \sum_{i=1}^{\#sat} \frac{P_{e,i} G_{e,i}}{4\pi d_i^2}$$

As mentioned, in case of GSO satellites this superposition of PFD method holds, as it can be assumed that P_{e_i},$G_{e,i}$ and d_i are constant and time invariant. In case of non-GSO satellites the relative positions are varying and requires a slightly different approach.

B. Equivalent Power Flux Density

EPFD calculation takes into account all emissions from non-GSO satellites at any given moment in time. At each unique time instance, the (directive) gain of the receiving antenna should be considered, i.e. the pointing of the receiver towards every source of interference. The methodology of EPFD considers the interference to be varying in time due to the varying relative positioning in space. Formally, EPFD is defined as [7] [8]:

$$\text{EPFD} = \sum_{i=1}^{\#sat} \frac{P_e G_{e,i}(\theta_i)}{4\pi d_i^2} \cdot \frac{G_{r,i}(\phi_i)}{G_{r_{max}}}$$

Time variant challenges that arise during the complex calculations:

1) P_e, time-variant and system dependent. However, in worst case scenario the maximum emitted power can be used.
2) θ_i, the off-axis angle between boresight of the emitter and the direction of the receiver.
3) $G_{e,i}(\theta_i)$, due to the rotation of the satellite this gain can vary over time.
4) ϕ_i, the off-axis angle between the pointing direction of the receiver and the direction of the emitter.
5) $G_{r,i}(\phi_i)$, the radiation pattern of the receiver is assumed to be known or estimated, therefore simplifying the calculation to only be dependent on the relative position of the interference source.
6) d_i, dependent on time, however can directly be calculated from the relative position of the transmitter.

Combining the power and gain of the noise source into an effective isotropic radiated power (EIRP) will result in the following equation:

$$\text{EPFD} = \sum_{i=1}^{\#sat} \frac{\text{EIRP}_i}{4\pi d_i^2} \cdot \frac{G_{r,i}(\phi_i)}{G_{r_{max}}} \tag{1}$$

Now that the calculation of EPFD is explained, its evaluation needs to be addressed. The goal of the RFI estimation is to determine the impact on astronomical measurements. In the next section it is explained how the ITU has made it possible to asses its impact.

III. THEORY: STOCHASTIC EPFD EVALUATION

To determine the detrimental effects the emitted noise has on the astronomical data, a standard has been developed to display the results from the calculation shown in section II. It is based on the probability the constellation of satellites has on data-loss of astronomical data per part of the sky. A division is made, which is explained in the following subsection. After this, the computation stress of the stochastic analysis is discussed.

A. Sky-division Cells

To make a statistically valid EPFD, every possible point in the sky should be considered. For this reason, the sky is divided into rings and cells, defined in the ITU recommendation [7], [8]. The sky is described as 30 rings stacked on top of each other creating a dome, where each ring has a certain number of cells. The lower rings have the maximum number of cells, and the top rings have the least number of cells. This is because the top of the dome is smaller than the lower part. In a later subsection, it is explained how the code goes through all these rings and cells and how it chooses a random position within such a cell. Fig. 1 provides an example of the sky division found in Recommendation ITU-R M.1583-1 [7].

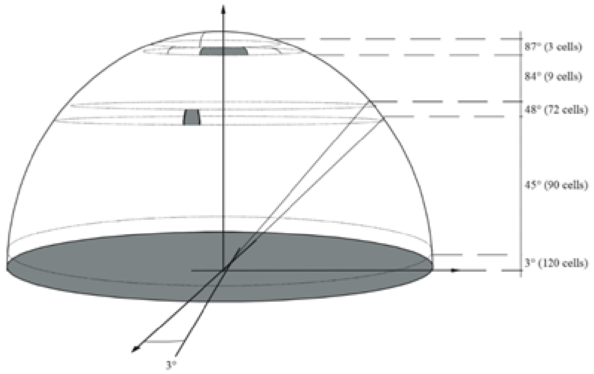


Fig. 1: Example of the Sky Division acoording to ITU-R M.1583-1

B. Computational Stress

In ITU-R M.1583-1, it is stated: "The methodology involves a number of trials, each of which calculates the averaged EPFD level over a 2000 s integration interval. A sufficient number of trials is needed to achieve the required confidence level in the results. In particular, the number of trials multiplied by the 2000 s integration time should be significantly higher than the period of the constellation" This results in parameter sweeps that need to be performed:

- Pointing of the Telescope, the direction in which the telescope is pointing within a certain cell.
- Moment in Orbit-Period, a time moment chosen within one orbit-period.
- Sky-division Cell, a cell chosen within the sky-division.
- Number of Satellites

With the Pointing and Moment parameters being random. The idea is to calculate the EPFD for different constellation moments, while considering a random pointing of the telescope within each cell. The goal is to obtain a statistical distribution of the EPFD in each considered cell. This is comparable to a Monte-Carlo type of simulation.

C. Data Loss Threshold

For every cell in the entire dome, the EPFD value is compared to the threshold found in [8]. The total number of times the threshold is exceeded is then added up and divided by the total number of cells. This is then defined as the data loss for one specific trial. This calculation is done over several trials, chosen to be five following the recommendations, from which a standard deviation and an average percentage are calculated. At this point, the power reduction needed to meet the 2% data loss threshold given by the ITU Recommendations [8] can be found. This is done by systematically lowering the total EPFD with steps of 1 dB, according to the recommendations. For every step, within the given range, a new average percentage is calculated. All these new average percentages are then plotted against their corresponding power increments. The amount, in dB, by which the EPFD needs to be reduced to get below the threshold, is then printed. Such a figure of the data losses can be seen in Fig. 2, where the actual reduction is found by taking the attenuation value found when the 2% value is passed.

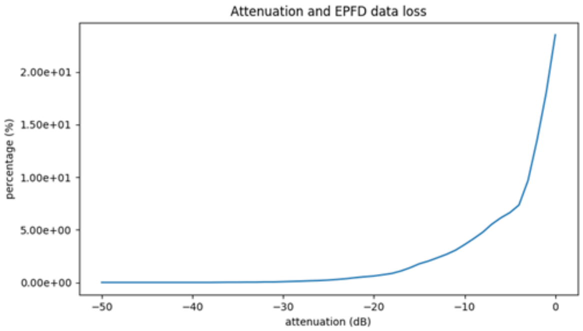


Fig. 2: The EPFD data loss for different EPFD value shifts

In the following section, the implementation of EPFD calculation is explained as it was recommended by the ITU. It focuses on the novelty that was added using TLEs and how it can be used as a solution to the time variant challenges as described previously.

IV. IMPLEMENTATION: ADDING TLE

As the previous section has shown, calculating RFI from non-GSO satellites and its impact on radio astronomical measurements is a complex and elaborate endeavour. Therefore, in this section, equation (1) is used to simplify the RFI estimation.

In equation (1), the three main components of EPFD calculation can be distinguished:

- EIRP
- FSPL
- Weight

with:

$$\mathrm{PFD}_i = \frac{\mathrm{EIRP}_i}{4\pi d_i^2} = \frac{\mathrm{EIRP}_i}{\mathrm{FSPL}_i} \tag{2}$$

$$W_i = \frac{G_{r,i}(\phi_i)}{G_{r_{max}}} \tag{3}$$

To verify the correct implementation of equation (1), the MATLAB and Python implementation are compared through a qualitative analysis of the results. The results are obtained by using the same input data in both implementations. This is emphasized, since the novelty of using TLEs lies not in the implementation in Python, but in the more general way of determining the FSPL, as seen in equation (2), and the weight, defined as equation (3).

By using the proposed TLE method: EIRP_i is assumed (isotropic radiator) for simplicity, while W_i and d_i are derived from perturbation models that predict orbits of satellites. In the following subsections it is first explained how TLE support is added, followed by a solution for determining the elevation of a satellite.

A general overview of the changes implemented by using TLE's and the effects can be seen in Fig. 3.

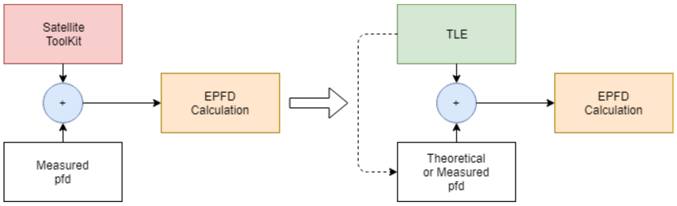


Fig. 3: New inputs for EPFD Calculations

A. Two line elements

Instead of making use of the Satellite ToolKit, TLEs were chosen to give the satellite constellations. TLEs are free to use and easily accessible on the internet. The info about which TLEs file to read is hardcoded into the program because only Iridium satellites were considered up until this point. This can be easily changed into an input for the user, but for now it is kept automatic. TLEs have a specific format for all satellites. It consists of three lines per satellite starting with a line which specifies the name and possibly the number of the satellite. This is then followed up by two lines including all the satellite information in a cryptic way. Within these two lines, the information about the satellite number, epoch date, and derivatives of the mean motion can be found, which can be easily read out. Frequently updated TLEs can be downloaded from sites like Celestrak [9], where it is just downloaded as a text file.

Using the sgp4 packet in Python, these two lines from the TLE can be read out. When doing so, several satellite objects are created. From these objects, characteristics such as position and velocity can be found. The position of a certain satellite can be called in a single line which needs a date and time as input. The created python code calls upon this position in a loop going through 1440 steps and increments the minute value of the input every time. The 1440 minutes were chosen to represent one day. All the positions of all the satellites for a specific day can be found, where the specific date is specified by the day at which the TLE was uploaded. This can be changed easily to be given as user input, where a specific day is given. Once all these positions are gathered, only the data points within sight of the telescope should be considered.

B. Changing the reference frame

Since the positions given in the TLEs files are in a Cartesian coordinate system with the centre of earth as the origin, a change of the reference frame is needed. The satellite positions relative to the radio telescope allow for simple determination of their elevation and distance. As can be seen in Fig. 4, the used coordinate system transformation consists of two rotational operations and a translation.

By first applying the rotational matrix multiplication as shown in equation (4), the translational operation becomes an additive constant along the rotated Z-axis that is equal to the Earth's radius (approx. 6371 km).

$$\mathbf{p_{sat}}' = \mathbf{M} \cdot \mathbf{p_{sat}} \tag{4}$$

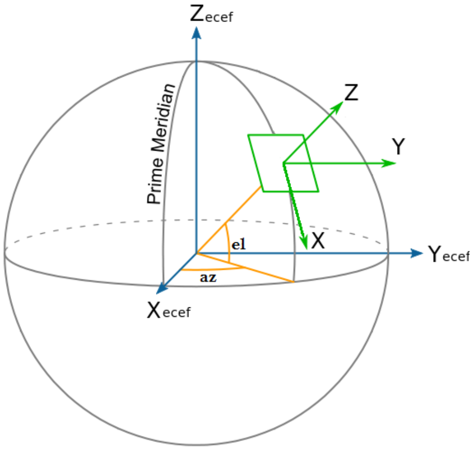


Fig. 4: Visualization of a change of the reference frame

In which the matrix $\mathbf{M}$ is defined as:

$$\mathbf{M} = \begin{bmatrix} \cos(az) & -\sin(az) & 0 \\ \sin(az) & \cos(az) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \sin(el) & 0 & \cos(el) \\ 0 & 1 & 0 \\ -\cos(el) & 0 & \sin(el) \end{bmatrix}$$

Where az and el are determined by the position of the arbitrary evaluated radio telescope and P_{sat} is the coordinate vector of a satellite, where the apostrophe denotes the new reference frame. In this specific case, it is assumed that the longitude and latitude coordinates are respectively the azimuth and elevation coordinates/angles.

By mere Cartesian to spherical coordinate transformation, the angle (ϕ_i) and distance (d_i) between the radio telescope and the satellite can be determined. If the elevation angle is positive, the satellite is in sight of the radio telescope. If the elevation is negative, the data point can be neglected, since it has no influence on the radio telescope.

To estimate the RFI and finally determine the impact a constellation of non-GSO satellites has on the astronomical data-loss, the EIRP should be determined.

C. Theoretical EIRP

Info about the power and the gain of the transmitter are required for calculating the EIRP per satellite. This data could be provided by measurements. However, due to lack of this data, a theoretical rule of thumb is used. The assumption made is a constant power multiplied by a gain of 1 W per 25 kHz. After this, an interference to carrier factor of -40 dB is used. Both of these values are based on related observations. Note that in this paper, it is irrelevant to use accurate values for the EIRP, as it a focuses on RFI estimations based on TLE assisted EPFD calculations. The novelty lies in incorporating the positional data of non-GSO satellites.

V. RESULTS

A. Verification Results

Along the way of writing the new implementation, a lot of results were constantly checked with the existing MATLAB code. This was done because the MATLAB code was already approved and checked by the appropriate organizations. This

TABLE I: Verification Results

	MATLAB	Non-TLE Python
Frequency (MHz):	1610.6267	1610.6267
Integration time (s):	2000	2000
Data loss (%):	41.21	40.93
Statistical error (%):	0.89	0.35
Required reduction (dB):	18.37	18

check was done one on one until the implementation of TLEs, where the results would deviate as expected due to the different input data and the newly implemented statistical approach. To check whether the actual EPFD method was implemented correctly, the data from the Satellite ToolKit used in the MATLAB code was used to do the calculations in Python once. If the result coincided within an expected statistical deviation, it could be concluded that the EPFD calculation was implemented correctly in Python according to the standards.

The results of such a comparison of MATLAB and Python using the Satellite ToolKit data can be seen in Table I. As can be seen in the table, only a small deviation between both data losses are found, which can be expected as a result of the statistical approach used for the EPFD calculation. The difference in the required reduction is due to the approach of how MATLAB goes through different value shifts.

B. Analyzing the evaluated EPFD

For each trial that the code is ran, the average data loss percentage over the entire sky for a specific frequency is calculated. The entire average data loss over the trials over the sky is plotted according to the way the sky is defined. Such a graph can be seen in Fig. 5, where the representation style is based on the figures created in the MATLAB code.

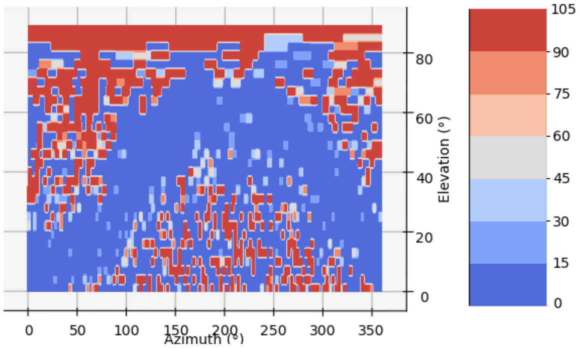
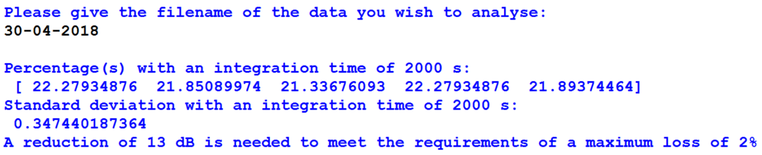


Fig. 5: Data loss over the sky

At this point, the reduction needed to get below the recommended threshold is calculated. It is plotted in Fig. 2 and all the results are then printed as can be seen in Fig. 6.

One could notice that different results are achieved when compared to the results shown in Table I. This difference arises from different input type files that have been used for the position of the satellites and a different position of the telescope. For the comparison, the Satellite ToolKit data was used, where the relative position to the telescope was already included. With the use of the TLEs the position of the telescope has to be given as user input. For this run, an arbitrary Longitude an Latitude have been chosen resulting in a different data loss, which is not unexpected.



```
Please give the filename of the data you wish to analyse:
30-04-2018

Percentage(s) with an integration time of 2000 s:
 [ 22.27934876  21.85089974  21.33676093  22.27934876  21.89374464]
Standard deviation with an integration time of 2000 s:
 0.347440187364
A reduction of 13 dB is needed to meet the requirements of a maximum loss of 2%
```

Fig. 6: General output of the Python code

VI. CONCLUSION

It has been shown that the Python implementation gives the same results as the MATLAB implementation, considering the same input data. This was done to check whether the EPFD calculations were implemented correctly. Instead of the Satellite ToolKit the Python implementation makes use of TLEs, which are a lot more accessible, free to use, and are updated almost daily. Apart from the TLEs to make the code more accessible, a theoretical case has also been included. This theoretical addition is one specific case, but can be changed easily according to the circumstances the user expects. Via this, one can make an analysis of a certain, even entirely theoretical, constellation without having to have actual measurement data. These new additions make the Python code more accessible and usable for more separate cases.

It turned out that both the MATLAB and the Python implementation have similar computation times. Python, however, achieves this by using less data points. More data points are easily accessible with the Python code but this would hugely increase the computation time while only slightly increasing the accuracy.

ACKNOWLEDGMENT

This paper is based on an internship at Astron under supervision of dr. ir. Hans van der Marel.

REFERENCES

[1] P. G. Wiid, H. C. Reader, and R. H. Geschke, "Radio frequency interference and lightning studies of a square Kilometre Array demonstrator structure," *IEEE Transactions on Electromagnetic Compatibility*, vol. 53, no. 2, pp. 543–547, 2011.

[2] S. Van De Beek, R. Vogt-Ardatjew, and F. Leferink, "Intentional electromagnetic interference through saturation of the RF front end," *2015 Asia-Pacific International Symposium on Electromagnetic Compatibility, APEMC 2015*, pp. 132–135, 2015.

[3] ECC, "ECC REPORT 171 IMPACT OF UNWANTED EMISSIONS OF IRIDIUM SATELLITES ON RADIOASTRONOMY OPERATIONS IN THE BAND 1610.6-1613.8 MHZ 0 EXECUTIVE SUMMARY," 2011.

[4] ——, "ECC REPORT 226 -Page 2 0 EXECUTIVE SUMMARY."

[5] ——, "Description of the software tool for processing of measurements data of IRIDIUM satellites at the Leeheim station," 2016.

[6] B. Levit and J. Lesh, "Radio Frequency Interference From Near-Earth Satellites," 1977.

[7] ITU, "Policy on Intellectual Property Right (IPR) Multichannel sound technology in home and broadcasting applications," pp. 2159–4, 2009.

[8] ——, "RECOMMENDATION ITU-R S . 1586 Calculation of unwanted emission levels produced by a non-geostationary fixed-satellite service system at radio astronomy sites ANNEX 1 Calculation of unwanted emission levels produced by a non-GSO FSS system at radio astrono," pp. 1586–1, 2002.

[9] NORAD, "CelesTrak: Current NORAD Two-Line Element Sets." [Online]. Available: https://www.celestrak.com/NORAD/elements/